Water Science and Technology Library

PRACTICAL HYDROINFORMATICS

Computational Intelligence and Technological Developments in Water Applications

edited by
Robert J. Abrahart, Linda M. See,
Dimitri P. Solomatine

PRACTICAL HYDROINFORMATICS

Water Science and Technology Library

VOLUME 68

For other titles published in this series, go to www.springer.com/series/6689

Robert J. Abrahart · Linda M. See ·
Dimitri P. Solomatine (Eds.)

Practical Hydroinformatics

Computational Intelligence and Technological Developments in Water Applications

Springer

Editors

Robert J. Abrahart
University Nottingham
Dept. Geography
University Park
Nottingham
United Kingdom NT7 2QW
bob.abrahart@nottingham.ac.uk

Linda M. See
University of Leeds
School of Geography
Fac. Earth and Environment
Woodhouse Lane Leeds
United Kingdom LS2 9JT
l.m.see@leeds.ac.uk

Dimitri P. Solomatine
UNESCO - IHE
Institute for Water Education
2601 DA Delft
The Netherlands
and
Water Resources Section
Delft University of Technology
The Netherlands
d.solomatine@unesco-ihe.org

ISBN: 978-3-540-79880-4 e-ISBN: 978-3-540-79881-1

Library of Congress Control Number: 2008928293

Cover design: Boekhorst Design b.v.

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Contents

Part VI Model Integration

List of Contributors

M.B. Abbott
Knowledge Engineering BVBA, Avenue Francois Folie 28, Box 28, 1180 Brussels, Belgium, and European Institute for Industrial Leadership, Château Latour de Freins, 1180, Brussels, Belgium, www.eiil.net,
e-mail: knowledge.engineering@skynet.be

R.J. Abrahart
School of Geography, University of Nottingham, University Park, Nottingham, NG7 2RD, UK

A. Bardossy
Institut für Wasserbau, Universität Stuttgart Pfaffenwaldring 61, D-70569 Stuttgart, Germany, e-mail: Andras.Bardossy@iws.uni-stuttgart.de

H. Betts
SAGRIC International Pty Ltd., 16 Carlisle Court, Hallett Cove, S.A. 5158 Australia

M. Brilly
Faculty of Civil and Geodetic Engineering, University of Ljubljana, Jamova 2, SI-1000 Ljubljana, Slovenia, e-mail: mbrilly@fgg.uni-lj.si

I-F. Chang
Department of Hydraulic and Ocean Engineering, National Cheng Kung University, Taiwan

Q. Chen
RCEES Chinese Academy of Sciences, Shuangqinglu 18, Haidian District, Beijing 100085; WL | Delft Hydraulics, Rotterdamsweg 185, 2600 MH Delft,
The Netherlands

S.-T. Chen
Department of Hydraulic and Ocean Engineering, National Cheng Kung University, Taiwan

C.D. Doan
Tropical Marine Science Institute, National University of Singapore, 12 Kent Ridge Road, Singapore 119223

S. Clark
Water Technology Pty Ltd, Unit 15 Business Park Drive, Notting Hill, VIC, 3168 Australia

I.D. Cluckie
Water and Environmental Management Research Centre (WEMRC), Department of Civil Engineering, University of Bristol, Bristol, BS8 1UP, UK,
e-mail: I.D.Cluckie@bristol.ac.uk

P. Coulibaly
Department of Civil Engineering/School of Geography and Earth Sciences, McMaster University, Hamilton, Ontario, L8S 4L7 Canada,
e-mail: couliba@mcmaster.ca

C.K. Baldwin
Utah Water Research Laboratory, Utah State University, Logan, Utah 84322-8200 USA

D. Butler
Centre for Water Systems, School of Engineering, University of Exeter, UK

C.W. Dawson
Department of Computer Science, Loughborough University, Loughborough, Leicestershire, LE11 3TU, UK, e-mail: C.W.Dawson1@lboro.ac.uk

N.J. de Vos
Water Resources Section, Faculty of Civil Engineering and Applied Geosciences, Delft University of Technology, P.O. Box 5048, 2600 GA, Delft, The Netherlands

A. Efstratiadis
Department of Water Resources, School of Civil Engineering, National Technical University of Athens, Heroon Polytechneiou 5, 157 80 Zographou, Greece,
e-mail: andreas@itia.ntua.gr

A. Elshorbagy
Centre for Advanced Numerical Simulation (CANSIM), Department of Civil & Geological Engineering, University of Saskatchewan, Saskatoon, SK, Canada S7N 5A9

E.S. Chan
Tropical Marine Science Institute, National University of Singapore, 12 Kent Ridge Road, Singapore 119223

D. Fortune
Wallingford Software Ltd, Howbery Park, Wallingford, Oxfordshire, OX10 8BA UK, e-mail: david.fortune@wallingfordsoftware.com

L. Garrote
Universidad Politécnica de Madrid, Dpto. Ingeniería Civil, Hidráulica y Energética, E.T.S. Ingenieros de Caminos, Ciudad Universitaria, 28040 Madrid, Spain, e-mail: garrote@caminos.upm.es

S. Ghosh
Department of Civil Engineering, Indian Institute of Science, Bangalore, India, e-mail: pradeep@civil.iisc.ernet.in

P. Gijsbers
WL | Delft Hydraulics, NL, e-mail: peter.gijsbers@wldelft.nl

J. Gregersen
DHI Water and Environment, Denmark, e-mail: jbg@dhigroup.com

D. Han
Water and Environmental Management Research Centre (WEMRC), Department of Civil Engineering, University of Bristol, Bristol, BS8 1UP, UK

A.J. Heppenstall
School of Geography, University of Leeds, Woodhouse Lane, Leeds, LS2 9JT, UK

A.P. Jacquin
Departamento de Obras Civiles, Universidad Técnica Federico Santa María, Casilla 110-V, Valparaísco, Chile, e-mail: alejacquin@yahoo.com

A. Jain
Department of Civil Engineering, Indian Institute of Technology Kanpur, Kanpur – 208 016, UP, India, e-mail: ashujain@iitk.ac.in

Z. Kapelan
Centre for Water Systems, School of Engineering, Computer Science and Mathematics, University of Exeter, UK EX4 4QF

S.-T. Khu
Centre for Water Systems, School of Engineering, Computer Science and Mathematics, University of Exeter, UK EX4 4QF, e-mail: s.t.khu@exeter.ac.uk

D. Koutsoyiannis
Department of Water Resources, School of Civil Engineering, National Technical University of Athens, Heroon Polytechneiou 5, 157 80 Zographou, Greece, e-mail: andreas@itia.ntua.gr

X. Li
Department of Civil and Environmental Engineering, Louisiana State University, 3418G Patrick F. Taylor Hall, Baton Rouge, LA 70803-6405, e-mail: xli11@lsu.edu

C. Maksimovic
Deptartment of Civil and Environmental Engineering, Imperial College London, UK

C.K. Makropoulos
Department of Water Resources & Environmental Engineering, School of Civil Engineering, National Technical University of Athens, Greece, e-mail: cmakro@mail.ntua.gr

S. Markar
WRM Water and Environment Pty Ltd, Level 5, Paddington Central, Brisbane, Qld., 4064 Australia

L. Mediero
Universidad Politécnica de Madrid, Dpto. Ingeniería Civil, Hidráulica y Energética, E.T.S. Ingenieros de Caminos, Ciudad Universitaria, 28040 Madrid, Spain, e-mail: garrote@caminos.upm.es

A. Moghaddamnia
Formely of Water and Environmental Management Research Centre (WEMRC), Department of Civil Engineering, University of Bristol, Bristol, BS8 1UP, UK (currently at University of Zabol, Iran)

K. Mohammadi
Associate Professor, Tarbiat Modares University, P.O. Box 14115-336, Tehran, Iran, e-mail: kouroshm@modares.ac.ir

M. Molina
Universidad Politécnica de Madrid, Dpto. Ingeniería Civil, Hidráulica y Energética, E.T.S. Ingenieros de Caminos, Ciudad Universitaria, 28040 Madrid, Spain, e-mail: garrote@caminos.upm.es

R. Moore
CEH Wallingford, UK, e-mail: rvm@ceh.ac.uk

P.P. Mujumdar
Department of Civil Engineering, Indian Institute of Science, Bangalore, India, e-mail: pradeep@civil.iisc.ernet.in

A. Mynett
RCEES China Academy of Sciences (Shuangqinglu 18, Haidian District, Beijing 100085); WL | Delft Hydraulics (Rotterdamsweg 185, 2600 MH Delft, The Netherlands)

J.P. O'Kane
Department of Civil and Environmental Engineering, University College Cork, Cork, Ireland, e-mail: jpokane@ucc.ie

K. Parasuraman
Centre for Advanced Numerical Simulation (CANSIM), Department of Civil & Geological Engineering, University of Saskatchewan, Saskatoon, SK, Canada S7N 5A9

T.H.M. Rientjes
Department of Water Resources, Institute for Geo-Information Science and Earth Observation (ITC), P.O. Box 6, 7500 AA, Enschede, The Netherlands

D. Savic
Centre for Water Systems, School of Engineering, Computer Science and Mathematics University of Exeter, North Park Road, Exeter, EX4 4QF, UK, e-mail: d.savic@exeter.ac.ukf

L.M. See
School of Geography, University of Leeds, Woodhouse Lane, Leeds, LS2 9JT, UK, e-mail: l.m.see@leeds.ac.uk

A.Y. Shamseldin
Department of Civil and Environmental Engineering, The University of Auckland, Private Bag 92019, Auckland, New Zealand, e-mail: a.shamseldin@auckland.ac.nz

S.Y. Liong
Tropical Marine Science Institute, National University of Singapore, 12 Kent Ridge Road, Singapore 119223, e-mail: tmslsy@nus.edu.sg

D. Solomatine
UNESCO-IHE Institute for Water Education, P.O. Box 3015, 2601 DA Delft, The Netherlands

M. Sraj
Faculty of Civil and Geodetic Engineering, University of Ljubljana, Jamova 2, SI-1000 Ljubljana, Slovenia

L. Stravs
Faculty of Civil and Geodetic Engineering, University of Ljubljana, Jamova 2, SI-1000 Ljubljana, Slovenia, e-mail: lstravs@fgg.uni-lj.si

F.T-C. Tsai
Department of Civil and Environmental Engineering, Louisiana State University, 3418G Patrick F. Taylor Hall, Baton Rouge, LA 70803-6405, e-mail: ftsai@lsu.edu

E. Toth
DISTART – Faculty of Engineering, University of Bologna, Viale Risorgimento, 2 I-40136 Bologna, Italy, e-mail: elena.toth@unibo.it

Z. Vojinovic
UNESCO-IHE Institute for Water Education, P.O. Box 3015, 2601 DA Delft, The Netherlands

M. Werner
WL | Delft Hydraulics, P.O. Box 177, 2600 MH, Delft, The Netherlands

S.M. White
Integrated Environmental Systems Institute, Cranfield University, College Road, Cranfield, Bedfordshire, MK43 OAL, UK

P.-S. Yu
Department of Hydraulic and Ocean Engineering, National Cheng Kung University, Taiwan, e-mail: yups@mail.ncku.edu.tw

Y. Yuan
Formely of Water and Environmental Management Research Centre (WEMRC), Department of Civil Engineering, University of Bristol, Bristol, BS8 1UP, UK (currently at UNESCO-IHE)

Preface

Hydroinformatics has emerged over the last decade to become a recognised and established field of independent research activities within the hydrological and environmental science communities. Hydroinformatics is not just an application of Information and Communications Technologies (ICT) to water resources, hydraulics, hydrology or environment. It strives to provide an amalgamation of water science with modern technologies for the purposes of satisfying social requirements. The European Geosciences Union (EGU) held its first dedicated session on Hydroinformatics in 2005 at the Vienna Meeting; that same meeting voted to establish the Hydroinformatics Sub-Division and Technical Committee (part of the Hydrological Sciences Division). The aim of that original session was to provide an active forum in which to demonstrate and discuss the integration and appropriate application of emergent computational technologies in a water modelling context. The initial proposal for this book arose at that meeting out of a desire to collect together a range of different contributions from academics and practitioners working in various sectors across the field; there were no other published compendiums at that point which attempted to span the latest set of methods or topics of hydrological interest that were presented at our meeting. The starting point for the selection of authors was the session itself. Further contributors were invited to submit papers in order to bolster particular sections and provide a representative selection of research across the main thematic areas: neural networks, fuzzy logic, global and evolutionary optimisation, emerging technologies and model integration.

This book is aimed at hydrologists, scientists, students and practitioners interested in a set of techniques derived largely from artificial and computational intelligence to solve a range of problems in hydrology. We hope that this book will promote the field of Hydroinformatics and bridge the gap between theory and practice.

We would like to thank the chapter authors for their interesting contributions and the many reviewers who have helped to make this a useful and high-quality publication. We would also like to thank the publication team at Springer for their efficient services.

Nottingham, UK — *Robert J.Abrahart*
Leeds, UK — *Linda M. See*
Delft, The Netherlands — *Dimitri P. Solomatine*

Part I
Hydroinformatics: Integrating Data and Models

Chapter 1
Some Future Prospects in Hydroinformatics

M.B. Abbott

Abstract This chapter reviews more recent developments in hydroinformatics, contrasting the developments in engineering practice and those currently predominant in academia. The possibility is considered of whether a major part of current academic research corresponds to a degenerating research programme in the sense established by Lakatos. It is then indicated how the industrial and the academic developments can be reintegrated, as exemplified by a new approach, called the *Waterknowledge Initiative*. The modus operandi of the production and operation of the deliverables of the Initiative is explained, including the new institutional and business developments through which they may most appropriately come to presence. The notion of a *global knowledge provider* is introduced correspondingly.

Keywords Hydroinformatics · industry · academia · fifth-generation modelling · Waterknowledge Initiative · agriculture

1.1 Introduction

The present situation in hydroinformatics is increasingly marked by a discrepancy between the activities proceeding in industrial practice and those occurring in academia. The roots of this discrepancy may be traced to the social–economic pressures exerted upon practice on the one hand, and the often very different pressures exerted upon academic teaching and research on the other. The possibility must then be considered of whether a major part of current academic research corresponds to a degenerating research programme in the sense established by Lakatos (e.g. Lakatos, 1976/1979). It is then necessary to consider how the industrial and the academic developments can be reintegrated, as will be exemplified here by a new approach, called the *Waterknowledge Initiative*. This is de-

M.B. Abbott
Knowledge Engineering BVBA, Avenue Francois Folie 28, Box 28, 1180 Brussels, Belgium, and European Institute for Industrial Leadership, Château Latour de Freins, 1180, Brussels, Belgium, www.eiil.net., e-mail: knowledge.engineering@skynet.be

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

voted to elaborating and applying a fifth generation in modelling practice whereby web-based access to and instantiation and operation of numerical modelling systems are woven together with web-facilitated access to the most supportive and immediately relevant, human-expert knowledge and understanding. It will be explained why the production of this Initiative must be conceived as *essentially sociotechnical constructs*. The modus operandi of the production and operation of the deliverables of the Initiative must also be explicated, including the new institutional and business developments through which they may most appropriately come to presence. The notion of *a global knowledge provider* can then be introduced correspondingly. This contribution thus modifies as well as updates an earlier overview of Abbott (2004).

1.2 Hydroinformatics in Engineering Practice

Hydroinformatics was born when numerical modelling and data collection and processing came into a synergic relation at the end of the 1980s (Abbott, 1991). By that time, the field of numerical modelling had expanded its range from one that was restricted to the modelling of flows of water exclusively to a much wider ranging field that combined flows and all that these flows transported with them or otherwise influenced, which increasingly included living creatures that had, in turn, their own means of motion (Abbott and Warren, 1974). Data collection had expanded its range of activities similarly, passing from recordings of water levels and velocity distributions to recordings and samplings of distributions of sediments, chemical substances and aquatic vegetation and other forms of aquatic life, including marine, – and even to movements of human populations over daily and other cycles (Abbott et al., 1977). This development was driven by engineering practice applied to many construction and management projects, of which the largest and most prominent were associated with the flood protection of the city of Venice, the rail and motorway link across the Great Belt and the similar link between Denmark and Sweden across the Sound (Øresund, in Danish). The levels of investment involved in these last constructions alone, of some eight billion euros, justified large investments in modelling and measuring equipment with its surveillance, control and data acquisition (SCADA) systems and radio-linked, online and so real-time operating systems. The advantages of using a hydroinformatics approach became increasingly evident as these projects progressed, while these advantages attained to a new dimension in the last of these great engineering works due to the admission of the general public, whether through their interest groups or individually, into the chains of decision-making processes throughout the project. The extension of hydroinformatics onto this sociotechnical dimension as projected already in Abbott (1996) was realised for the first time in this project in such a way as to make the project publicly acceptable in the first place and a success in every other place. In the concluding words of Thorkilsen and Dynesen (2001), representing the owners of the project:

> With regard to the subject of the present paper, the role of hydroinformatics in the completion of the Øresund Fixed Link, hydroinformatics certainly took an important place, right

from the beginning of the feasibility phase right through to its completion. It was fully integrated into the planning and design, as well as the execution and control phases of the project. Its significance, which may prove relevant for similar or other kinds of future construction projects having water and environmental sides to them, may be summarised as follows:

1) The environmental monitoring and modelling programme that emerged from the concerns of many organisations and the public generally was accordingly directed to the purpose of dissemination of environmental information, as well as to the needs of the construction process itself, and institutional arrangements were made correspondingly. This emergence of a complex of aims and methods, of public aspirations and government legislation, of citizen groups and institutional arrangements, of criteria and controls, of instrument networks and modelling tools, and, most generally, of people and technical equipment, was prepared by and remains the ongoing subject of hydroinformatics.
2) It seems reasonably established that without the promises held out by the development of hydroinformatics already in the 1990s, the transport link between Denmark and Sweden could only have been realised with great difficulties, despite its being backed by powerful political interests. Of course, without the political support it would never have succeeded either, regardless of any amount of development in hydroinformatics. The success of the project was the result of a synergy between these two streams of developments. It was a consequence of a synergy between a new kind of political will and new kinds of sociotechnical means.
3) From the point of view of hydroinformatics, the Øresund project marked the transition from an emphasis on information technology to an emphasis on communication technology. Although exceedingly rudimentary and limited compared with the potentialities of currently ongoing developments based on new telecommunications technologies, the [internetted information and knowledge serving] EAGLE system can be seen as a precursor of presently planned and future decision support systems that can be distributed across the internet (see, for example, Abbott and Jonoski, 2001).
4) These developments not only served the interests of the general public but also provided substantial advantages to the contractors and others who were concerned with the construction. For example, the dredging works were able to proceed with a high degree of efficiency, while the widespread use of hydroinformatics and GPS systems substantially reduced the costs of the placing of rock armour, the tunnel elements, bridge piers and other structural elements. There were thus important business advantages, much less negotiation and very little litigation as compared with similar large infrastructural projects carried out elsewhere in Europe.
5) An important consequence of this situation was that the link was built to specification, its construction and operation satisfied all environmental conditions, it was constructed within its original budget and it was completed six months ahead of schedule. This was a compliment to all parties concerned, but also to the achievements of hydroinformatics as a new discipline in this field.

It is a long quotation, but it sums up very well the development in hydroinformatics as currently applied in engineering practice.

1.3 Hydroinformatics in Academia

Hydroinformatics in academia has for the most part taken another track from that followed in engineering practice, having been much more occupied with a range

of techniques that have been directed towards transforming existing data sets into new actionable knowledge. Starting with artificial neural networks in the 1980s and extending through such technologies as those of genetic programming and support-vector machines, hydroinformatics has taken up what is still sometimes called a *sub-symbolic paradigm*, and it has attempted to extend this paradigm to cover a wide range of problems arising in the aquatic environment. This nomenclature arose from an earlier period that was associated with a succession of paradigms that were concerned with what was called *artificial intelligence* (AI) and then one particular development in that field that acquired the sobriquet of *Good Old-Fashioned AI* (GOFAI) which was based on representing intelligent behaviour by sequences of symbols, construed as rules and facts, so that it constituted a *symbolic paradigm*. Although showing that such devices could not by themselves imitate human behaviour constructively, Dreyfus pointed out (*loc. cit*. p. 5) that some rule-based devices that preceded the sub-symbolic devices, *but necessarily worked together with their human-intelligent users*, as exemplified already in 1970 by M.I.T.'s MATHLAB and Stanford's DENDRAL, were already at that time (and their successors still are) successful, and this was and remains possible because these devices themselves 'achieve success precisely because they are restricted to a narrow domain of facts, and thus exemplify what Edward Feigenbaum, the head of the DENDRAL project has called *knowledge engineering*'. In effect, the workings of these systems cannot be separated in practice from the knowledge of the persons who are using them, whereby the much more flexible knowledges of the users actively complement the knowledge encapsulated as rules into the systems themselves.

By the time that hydroinformatics took up sub-symbolic methods, Hubert Dreyfus (1992) could write with confidence that

> After fifty years of effort…however, it is now clear to all but a few diehards that this attempt to produce general intelligence has failed. This failure does not mean that this sort of AI is impossible; no one has been able to come up with such a negative proof. Rather, it has turned out that, for the time being at least, the research programme based upon the assumption that human beings produce intelligence by using facts and rules has reached a dead end, and there is no reason to think that it could ever succeed. Indeed…GOFAI is a paradigm case of what philosophers call a degenerating research programme.
>
> A degenerating research programme, as defined by Imre Lakatos [1976/1978], is a scientific enterprise that starts out with great promise, offering a new approach that leads to impressive results in a limited domain. Almost inevitably researchers will want to try to apply the approach more broadly, starting with problems that are in some way similar to the original one. As long as it succeeds, the research programme expands and attracts followers. If, however, researchers start encountering unexpected but important phenomena that consistently resist the new techniques, the programme will stagnate, and researchers will abandon it as soon as a progressive alternative approach becomes available.

Hydroinformatics came on the scene when the GOFAI paradigm was already largely discredited, but it then came to take up as the successor of GOFAI in a big way, and the viability of the programme of this successor has in turn been called into question. In the words of Dreyfus, 'The triumphal arrival of the neural-net revolutionaries, also called connectionists, completed the degeneration of the GOFAI programme [in that] computers running simulations of such nets do not count as

physical symbol systems'. Dreyfus in turn quotes Paul Smolensky when he says, 'In connectionist systems, knowledge is encoded not in symbolic structures, but rather in the pattern of numerical strengths of the connections between processors'. Thence the expression: *'sub-symbolic* paradigms'.

Now of course the range of sub-symbolic paradigms has extended way beyond the already wide class of artificial neural networks (ANNs) and now covers all manner of devices, many of which are sure to be described in this volume. In the case of ANNs, Dreyfus explained why these must also be very limited indeed in their capacity to represent, not to speak of reproducing, intelligent human behaviour. At the same time, however, none of these devices so far necessitates the presence of human intelligence in its actual workings, although of course some intelligence may be required to instantiate them and to make sense of their productions. They are thus *essentially technocratic systems*, and as Dreyfus has shown in the case of ANNs, this alone must limit their range of applications severely. As Dreyfus explained, the basic problems had been adumbrated already by Pascal in the seventeenth century, and we may add that they were already explicated in Chinese mathematics some 1000 years earlier (Chemla and Guo Shuchun, 2004).

The great achievements on the one hand of what we might, for want of a better name, call *engineering hydroinformatics* seem nowadays to stand in sharp contrast with the relative paucity of commercially viable applications of sub-symbolic paradigms in engineering and management practice. The question then naturally arises of whether the sub-symbolic paradigm constitutes in its turn a degenerating research programme of the kind for which GOFAI provides a paradigm case. As soon as we pose this question, however, we are bound to recall many similar situations that have occurred in our own field over the last 50 or so years.

The most memorable of these for those whose memory goes back that far was the *dimensional analysis paradigm* – one is tempted to write 'syndrome' – that swept through hydraulics during the period that computational hydraulics was still in its infancy and which was largely inspired by similar movements in other disciplines. The journals of hydraulic research of the time had a regular quota of papers on this, that or the other dimensional analysis of whatever hydraulic phenomenon the author happened to be investigating, and doctoral theses were largely given over to great discoveries of new dimensionless numbers correspondingly, and several of these doctors became professors on this basis. Nowadays, of course, this era is all but forgotten and the results, such as they were, are largely ignored as irrelevant. The dimensionless numbers that we still use and still find useful originated for much the greater part in an earlier era.

But of course hydraulics was not alone in such endeavours. Indeed in this respect it is still apposite to repeat the observations of Klemeš (1986; see also Abbott, 1992) on this unfortunate habit of hydrologists also of misappropriating techniques borrowed from other disciplines:

> Hydrology, having no solid foundation of its own and moving clumsily along on an assortment of crutches borrowed from different disciplines, has always been an easy victim of this practice. Every mathematical tool has left behind a legacy of misconceptions invariably heralded as scientific breakthroughs. The Fourier analysis, as was pointed out by Yevjevich

> (in 1968), had seduced the older generation of hydrologists into decomposing hydrologic records into innumerable harmonics in the vain hope that their reconstruction would facilitate prediction of future hydrologic fluctuations (fortunately few computers were available at the time, so that the Fourier fever did not become an epidemic); various statistical methods developed for evaluation of differences in repeatable experiments have been misused to create a scientific analysis of unrepeatable hydrologic events; linear algebra has been used to transform the idea of a unit hydrograph from a crude but useful approximation of a soundly based concept into a pretentious masquerade of spurious rigour now exercised in the modelling of flood events; time series analysis has been used to remake inadequate 20-year stream flow records into 'adequate' 1,000 year records, or even more adequate 10,000 year records; and the illusion of pattern recognition is now being courted in the vain hope that it will lend legitimacy to the unscientific concept of mindless fitting that dominates contemporary hydrologic modelling. In all these cases, mathematics has been used to redefine a hydrologic problem rather than to solve it.

From the point of view of practitioners in physics-based hydrologic modelling using all the available resources of supporting modern-scientific disciplines, it seems that many hydrologists continue to go to any lengths to avoid the hard thinking and mathematical difficulties necessitated by a physics-based approach in this subject. One of the escapes from the shear difficulty, and often algebraic drudgery (*Matlab* notwithstanding!), of the physics-based approach is that of the so-called 'conceptual modelling', and there are still many persons who try to represent the behaviour of a complete catchment over all possible successions of meteorological events by a few parameters in such a model. The physics-based modellers often refer to these productions derogatively as 'pots-and-pipes models'.

The now-classical analysis of such movements in the general case is that of Heidegger (1927/1962, pp. 165 and 168–179). He explained how such a paradigm can be formed, or rather forms itself, within a community, and how it takes over the thinking processes of those who are held within its spell within the community concerned. It is important however to understand how and why hydroinformatics proceeds in this way within most of academia and in another way again in the world of engineering practice, and this can be done by considering the most flagrant discrepancies, such as occur in the area of hydrology.

In 1996, Abbott and Refsgaard considered the marked discrepancies between developments in hydrology which could in principle be catalysed by developments in hydroinformatics and developments in those fields, like river engineering and management and coastal engineering, where already at that time hydroinformatics had triggered major advances. It had by then become clear that only distributed physically based models could make use of the rapid advances in instrumentation, SCADA systems with real-time data transmission, data assimilation modules, remotely sensed data and its interpretation systems, seamless GIS interfacing, advances in geodetic surveying incorporating GPS-based position-fixing equipment, cartographic transformation packages, intranetted and extranetted communications systems and the many other such developments that hydroinformatics had already woven together to obtain such massive synergies in other areas. By way of an explanation of this situation, four areas of difficulty were identified in the applications of such physics-based systems in hydrology, which are given as follows (*ibid*, pp. 12 and 13):

1. Data availability: A prerequisite for making full use of the distributed physically based models is the existence and easy accessibility of large amounts of data, including detailed spatial information on natural parameters such as geology, soil and vegetation and man-made impacts such as water abstractions, agricultural practices and discharge of pollutants.

In most cases, all such relevant data do not exist and even the existing data are most often not easily accessible due to lack of suitably computerised databases. A further complication in this regard is the administrative problem created by the fact that these models, in addition to the traditional hydrometeorological data, require and can make use of many other data sources, such as those arising from agricultural, soil science and geological investigations.

Another important development gradually improving the availability of data is the application of GIS technology, which is particularly suitable for couplings with distributed hydrological models. In general, even engineering hydrology does not attract the large investments in instrumentation, data networks and other facilities that are already accepted as justified in hydraulics and coastal engineering applications.

2. Lack of scientific-hydrological understanding: With the introduction of a new modelling paradigm and concurrent research in process descriptions, new shortcomings in the scientific-hydrological understanding have emerged, especially with regard to flow, transport and water-quality processes at small scales and their upscaling to describe larger areas. [Some of the key scientific problems were highlighted in several chapters of Abbott and Refsgaard (1996).]

These scientific shortcomings have, on the one hand, constrained the practical applications of distributed hydrological models and, on the other hand, the existence and application of such models have put a new focus on some of these problems, thus contributing to advances in the scientific-hydrological understanding.

3. Traditions in hydrology and water resources understanding: The distributed physically based model codes, as represented by the European Hydrologic System/Système Européen Hyrologique (SHE), constituted a ‘quantum jump’ in complexity as compared with any other code so far known in hydrology. Moreover, it used technologies, such as had been developed in computational hydraulics, with which few hydrologists were familiar. Although the numerical algorithmic problems could be largely overcome through the development of the codes into user-friendly fourth-generation modelling systems with well-proven algorithms, such as the MIKE SHE, the problem was then only shifted back to one of comprehending the fully integrated complexity of the physical system that was being modelled together with the constraints that were inherent in the modelling procedures. Very few professional engineers and managers were, and still are, educated and trained with the necessary integrated view of hydrological processes in anything like their real-world physical complexity.

This difficulty is exacerbated by the very nature of hydrology itself, whereby most professionals possess only a limited view of the physical processes involved. Soil physicists, plant physiologists, hydrogeologists and others usually have only

a very partial view on the whole system, while there are few organisations that have available both the full range of such specialists and the more broader-ranging professionals that are needed in many situations to exploit the potential of distributed physically based codes to such a degree that this exploitation is economically justified.

4. Technological constraints: In order to achieve a wide dissemination of modelling technology to a considerable part of the professional community (and not only to experienced hydrological modellers) experience from hydraulic engineering shows that fourth-generation systems (i.e. user-friendly software products) are required. Furthermore, it is believed that fifth-generation systems are required to realise their full potential in terms of practical applications. The fifth generation systems [will be] hydroinformatics based…'.

Thus, although the further and more intensive development of physically realistic computer-based hydrological modelling systems was seen to devolve upon developments in hydroinformatics, formidable institutional problems were seen already at that time to be blocking these developments in many areas. These institutional problems have become increasingly serious since that time and have now, in their turn, led to serious problems within the hydrologic community itself. These problems appear as a divergence within the community concerning the very nature of hydrologic modelling and practice, and with this the role that hydrology has to play, or indeed can play, in society as a whole. Such divisions strike at the heart of the scientific foundations of hydrology, raising questions concerning its very nature as a discipline based on the results of modern science.

1.4 Bringing Practice and Academia Back Together: The Waterknowledge Initiative

This Initiative, which is now operational, is concerned in the most immediate and pragmatic sense only with the construction, within an open-source environment, of an Internet-enabled vehicle for accessing modelling systems, instantiating models remotely using facilities provided by these systems in conjunction with proprietary GIS and running these models by using other such facilities again, while doing all of this with the Internet-enabled support of human experts. A further development associated more with the analysis, design and construction of a disaggregated modelling system using an agent-orientated architecture (see Abbott, 2004) was initiated during 2006. This is being developed initially as a teaching and academic research platform, but it obviously has important commercial potentialities. The work is proceeding around doctoral and other studies in Brussels, Cork, Delft and Exeter with the author as the 'owner', in the established open-source sense, of the project (Raymond, 1999, 2001). This provides the so-called 'ABCDE core group'.

The view of the project only in terms of a delivery vehicle for such sociotechnical modelling services needs however to be extended in order to take account of changes occurring in the purposes for which models will have to be employed in the

quite near future, and the present contribution is directed to explicating something of this new world of applications of modelling services. This is then further to say that the systems – or more precisely 'constructs' – have in this case to be conceived within many different operating environments, and these must come to differ in several important respects from our present, more uniform operating environments. The new operating environments with which this Initiative is concerned are then conditioned by several other factors than those occurring within the modelling community itself. There is no longer any question of an internal mutation occurring spontaneously within the hydroinformatics community itself, but one that is determined, and even forced, by circumstances that are far outside the control of this community. This development, forced for the most part by external social and technological advancements, is called *the fifth generation of modelling* (see Abbott, Tumwesigye et al., 2006 and Tumwesigye, 2005).

Since the notion of a *forcing process* may at first appear somewhat far removed from our present purposes, it may be apposite to introduce it through its most immediate consequence in our own field of modelling in hydraulics. This may be exemplified very simply by the displacement – although not necessarily replacement – of physical modelling by numerical modelling. It may then be recalled that the flows of water occurring in the waters of the physical model were made more visually apparent by placing illuminated objects upon their surfaces and photographing these light sources cinematographically as they moved, so as to provide traces of the water movements. In this way, the physical models provided what are called in semiotics *metonymic structures*. The eyes and mind of the experienced modeller then transformed these metonymic structures into *metaphors* that conveyed such meanings as eddy formations, vortex streets and tidal races. In the first decades of numerical modelling, before the development of graphical user interfaces, numerical models produced only sequences of numbers, which were ill-suited to the model-interpretation capabilities of all but a few experts. In this situation, the market for numerical models was quite severely restricted and physical models remained viable. However, as soon as graphical user interfaces were introduced that could provide similar devices to those provided by the physical models, such as patterns of moving blobs and their traces and patterns of arrows, and so metonymic structures and thus devices that enabled almost anyone to generate the corresponding metaphors in their own minds, the days of the predominance of the physical model were for the most part themselves 'numbered'. Thus, what is important to understand here is that *the physical model was not displaced from its dominant position by the numerical model as such, but it was displaced by the graphical user interface of the numerical model* – which is really quite another thing again.

As the results of model simulations become directed to new classes of end users, such as through projections from 3G telephones of expected crop developments under different water consumption and distribution scenarios on the walls of the homes of individuated (and often illiterate) farmers using the predictions of mass-customised advice-serving systems, so the nature of the modelling exercise – the way in which it functions – must change correspondingly (see Abbott et al., 2006; Gesso, 2005). The move from two-dimensional colour representations of results

to pseudo-three-dimensional colour representations using the devices of projection, with their realism heightened by the use of highlights and shadows and by placing the observer in an orbit around the phenomenon of interest – a development that has been developed extensively over 10 years now – indicates the continuation of this tendency as the range of end users for such *simulacra* has continued to expand. Once again, these must be woven into the activities of the human agents within whose minds the decisive metaphors must come most immediately to presence. In mass-customised advice-serving systems used in agriculture, aquacultural and public health applications in rural areas, these are called the *rural communicators* (Abbott 2000; Abbott and Jonoski, 2001; Gesso, 2005). These are the communicators and interpreters of the proffered advice and the elicitors of local, indigenous social and experiential knowledge. The present contribution is intended also to direct attention to these means of accessing the great new markets of users of the productions of models so as to provide some orientation on the development of the knowledge-promoting and knowledge-developing services involved.

Thus, the *Waterknowledge Initiative* is directed to laying the foundations for constructs that may be directed to becoming the most advanced and therewith the most cost-effective operations of their kinds. It proposes to achieve this goal by supporting the web-based access and exploitation of electronically encapsulated knowledge with parallel but interacting means to draw upon an adaptive network of genuine international expertise and other sources of knowledge and understanding in a variety of efficient and correspondingly economic ways. It is thus directed to the longer-term development of the initial delivery vehicle and supporting services even as other contributions have been and continue to be prepared to cover the shorter-term objectives. As an immediate consequence, the Initiative is directed simultaneously to the development of new business models, understanding that without commercially sustainable developments its innovations will be stillborn. Inseparable from this is the latent ambition of the larger business organisations to become *global knowledge providers*, providing web-based software services and brokering expert knowledge support also over the web on a worldwide basis.

This more general development is directed in the longer term to supporting and indeed enabling the analysis, design, installation and development of online and real-time construction, operating and management systems that will be highly adaptive to changing conditions, such as may occur slowly over years in some cases and over a few hours, or in extreme cases even over some minutes, in other cases. In many and probably in most cases, such systems will be under constant development in order to accommodate changes occurring in the area of interest as at the same time they will in many cases be updating themselves constantly and automatically in order to accommodate changes that they are themselves registering and analysing. Such systems will thus be highly dynamic and strongly self-referencing.

These systems not only will be applicable to the real-time operation of water environments of course, but will be invaluable adjuncts to the planning and the implementing of investments in flood prevention and flood reduction, land-improvement and amelioration projects. For example, the move away from ‘hard’ structural protection, as afforded by dykes, and more towards flood amelioration by the intentional

flooding of upstream areas (with of course adequate warnings, safety precautions and compensations, as well as the installation of safe areas, such as mounds, raised structures and suitably prepared housing) will make it possible to reduce downstream flooding but will require careful synchronisations of individual flooding events. We shall refer to this process as one of *sacrificial flooding* because it introduces in this area the (often ritualistic) orderings in time of sacrificial acts in general. Similarly, and by way of another example, the ongoing and intensified introduction of online water-quality measuring and monitoring systems with their SCADA systems providing permanent online communications to decision-enhancing facilities of all kinds, as directed in turn to all manner of 'decision-makers' including the general population, will provide greatly enhanced warning facilities and associated customised advice. Such facilities will also of course provide an increasing support to future industrial, residential and infrastructural investments by providing scenarios of future water-quality situations under, for example, normal, flood and drought situations as a basis for the issuance of construction and operating licences and permits. Thus, these online and real-time installations will provide the essential prerequisites for all future socio-economic developments in the water sectors of our societies. They correspond to the more general drift of societies away from knowledge communities composed primarily of 'knowers' to knowledge communities that are composed predominantly of 'consumer of knowledge', commonly described as a mutation from a *modern condition* of society to a *post-modern condition* of society (e.g. Lyotard, 1979/1986).

It follows from these and other examples that the designs of integrated systems of this kind cannot be made on the basis of technical data and technical considerations alone: they also necessitate a thorough appraisal of the social, including the socio-economic and political, environments within which the particular systems will have to operate as these change in time and space as human activities in river basins, for example, change and develop correspondingly. Since the social and political environments will be themselves in turn increasingly influenced by the changes wrought in the water sector, we come into the presence of often closely interacting, and indeed interwoven, social and technical developments – developments that are said to be *implected* in the language of object theory – so that we are again in the presence of the sociotechnical.

By the same token and as another example again, although one model of the consequences of this changing environment may be appropriate at one time, such as during the night in a residential area, it will often not be appropriate at another time, such as during the day when most persons in the area are working elsewhere. Thus, it will often be necessary to identify the ways in which activities in a basin, for example, may develop on the basis of specific habitual, infrastructural and managerial interventions while also identifying the consequences of these interventions for the populations and the lands within the basin. These consequences will commonly include the actions and reactions of the populations concerned, whether at the level of individuals or of social groups, as well as the reactions of and with physical and biological processes occurring within the areas concerned. These developments are in turn related to developments in web-based and satellite-enabled

mobile telephone technologies and – as the most challenging issues of all – their most appropriate modes of application and corresponding business arrangements.

It follows also that the Waterknowledge-generated projects will not normally be directed to providing the means to implement design, implementation and management procedures directed to one complete and final situation, but will instead be directed primarily to the installation of systems that will monitor the changing interacting social and technical environments occurring within the basin under whatever is the current operating strategy, while constructing extrapolations of the changing socio-economic and more general social conditions in order to evolve a new strategy that adapts to the changing one while still leading development in an overall desirable direction. The approach that is envisioned here is then incremental, sequential, reflexive and evolutionary rather than being first discontinuous and afterwards static. Such a system has of course to be 'set-up', or rather *instantiated*, for the existing situation, but then in such a way that it can monitor the parameters that define its further operation and estimate the best future courses of action at that and future times as indicated by these changing parameters.

This incremental, sequential and evolutionary approach is being applied also to the productions of the Waterknowledge Initiative itself, in that its constructs may be instantiated so as to comprehend only some parts and some aspects of water-related activities, being extended as required by changing circumstances as these become increasingly relevant and significant.

A system of this kind is commonly divided into its human components and its technical components, and this habit persists even when it is understood that these two components are so connected and intertwined, or again 'implected' that they cannot be properly separated in their operations. Such a system is then commonly described as an *essentially sociotechnical system*. However, to the extent that this system is extended to include the actions and reactions of the populations involved, it will usually not remain internally consistent, since there will commonly be several and even many different interests and intentions within the social groups that are involved and the 'system' that is proposed here has to accommodate itself to this situation. As already adumbrated, in strictly scientific terms this means that the operation and management problem cannot be posed as one of 'a system' at all in many applications, in that its parts may sometimes not be mutually consistent due to human attitudes and activities that are not consistent the one with the other, and in this case, as introduced earlier and as already employed here, we speak instead of an *essentially sociotechnical construct*. Obviously any so-called 'master plan' must take account of inconsistencies in human behaviour if it is to be at all realistic, so that one really ought to keep to this last terminology. Since however the word 'system' is so widely used (even if, in its strict scientific sense, it is here being misused!) we shall for the most part keep to it in this contribution.

In a similar vein, the term 'water resources' is still employed even while understanding, following the seminal work of Heidegger (1963/1977), that this is commonly used with strongly performative intentions, whereby the whole world of nature is regarded as something existing only as a 'standing reserve' for human exploitation and not as something existing in its own right. This was emphasised

from the beginning in the *Hydroinformatics* of Abbott (1991), but hydroinformatics as a discipline seems largely to have forgotten the admonitions of that work. Thus, we shall continue here to speak of 'water resources' even as we understand the dangers inherent in speaking in this way of what truly pertains to nature most generally and not to mankind exclusively.

At the same time, it is well understood that many and possibly most of the problems experienced downstream in river basins generally have their origins in inappropriate agricultural practices. Consequently, an important part of the applications of systems of the kind considered here will be given over to the operation and management of water resources for agricultural purposes and for determining the proper relation of these agricultural processes to those operational strategies and management practices that are directed more towards urban areas. This is envisaged as the greatest of all markets for modelling services, involving millions of end users, and it must also be analysed appropriately, and so in a *socio*technical way. It then appears that this development, and probably the development of hydroinformatics generally, will migrate to Asia, where it will be closer and better integrated with its main markets. Correspondingly, the first commercially viable demonstrators provided by the Waterknowledge Initiative was presented at the *Yangtze Forum* in April 2007.

1.5 Concluding Remarks

Although the programme proposed here must appear ambitious, it can be carried out for a large part with relatively little extra expense if the expertise that is available in the fields concerned is properly identified and engaged. Although the accent at the moment is on the use of models to analyse and design structural methods for providing flood prevention and to reduce water pollution, this priority is expected to change with increased rapidity over the next decade, with a greater emphasis on 'softer' and more natural methods covering much wider fields of application. This tendency is expected to spread further in education also. The new role of modelling that proceeds through this endeavour must then be self-evident.

References

Abbott MB (1991) Hydroinformatics – Information Technology and the Aquatic Environment. Avebury Technical, Aldershot

Abbott MB (1992) The theory of the hydrologic model, or: the struggle for the soul of hydrology, in O'Kane JP (ed) Advances in Theoretical Hydrology, A Tribute to James Dooge, Elsevier, Amsterdam, pp. 237–254

Abbott MB (1996) Introduction: The sociotechnical dimension of hydroinformatics, Ed. Müller, A. Hydroinformatics 96, Zurich

Abbott MB (2000) The gender issue in hydroinformatics, or Orpheus in the Underworld, Journal of Hydroinformatics 2(2): 87–104

Abbott MB (2004) Ten years of hydroinformatics conferences: A Partial Assessment, Parts 1 and 2, in Liong S-Y, Phoon K-K, Babovic V (eds) Proc. 6th Int. Conf. Hydroinformatics, World Scientific, Singapore, pp. 1679–1694

Abbott MB, Jonoski A (2001) The democratisation of decision making in the water sector, Journal of Hydroinformatics 3(1): 22–48

Abbott MB, Refsgaard JC (eds) (1996) Distributed Hydrological Modelling, Kluwer, Dordrecht

Abbott, MB, Warren R (1974) A dynamic population model, Journal of Environmental Management 2: 281–297

Abbott MB, Hodgins DO, Dinsmore A.P, Donovan M (1977) A transit model for the City of Miami, Journal of Environmental Management, 5: 229–242

Abbott MB, Thein KN, Huang Y (2006) Mass customised advice-serving systems in hydroinformatics, Proc. 7th Int. Conf. Hydroinformatics, Nice

Abbott MB, Tumwesigye BE, Vojinovic Z (2006) The fifth generation of modelling in hydroinformatics, Proc 7th Int. Conf. Hydroinformatics, Nice

Chemla K, Guo Shuchun (2004) Les Neuf Chapitres: Le Classique mathématique de la Chine ancienne et ses commentaires. Dunod, Paris

Dreyfus HL (1992) What Computers Still Can't Do: A Critique of Artificial Reason. MIT Press, Cambridge, Mass

Gesso KM (2005) A Demonstrator of a Mass-Customised Advice Serving System. MSc Thesis, UNESCO- IHE, Delft

Heidegger M (1927/1962) Sein und Zeit, Niemeyer, Tübingen/Being and Time. Trans. Macquarrie J, Robinson E, Blackwell, London

Heidegger M (1963) Die Technik und Die Kehre, Niemyer, Tübingen (Tenth Edition: Cotta, 2002)/ (1977), The Question Concerning Technology and Other Essays. Trans. Lovitt W, Harper, New York

Klemeš V (1986) Dilettantism in hydrology: transition or destiny? Water Resources Research 22(9): 177–188

Lakatos I (1976/1979) Proofs and Refutations, in Worral J, Zahar E (eds), Cambridge University Press, Cambridge

Lyotard J-F (1979/1984) La Condition postmoderne: raport sur le savoir, Minuit, Paris/The Postmodern Condition: A Report on Knowledge, University of Minnesota

Pascal B (1670) Les Pensées / (1962) (ed) Lafuma L, Seuil, Paris, pp. 244–246 / (1977) (ed) Le Guern M, Gallimard, Paris, pp. 328–330

Raymond ES (1999, 2001) The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary. O'Reilly, Sebastopol, CA

Thorkilsen M, Dynesen C (2001) An owner's view of hydroinformatics: its role in realising the bridge and tunnel connection between Denmark and Sweden. Journal of Hydroinformatics 3(2): 105–136

Tumwesigye E (2005) Towards a New Business Model in Hydroinformatics: the Case of Urban Drainage Models. MSc Thesis, UNESCO-IHE

Yevjevich V (1968) Misconceptions in hydrology and their consequences, Water Resources Research 4(2): 225–232

Chapter 2
Data-Driven Modelling: Concepts, Approaches and Experiences

D. Solomatine, L.M. See and R.J. Abrahart

Abstract Data-driven modelling is the area of hydroinformatics undergoing fast development. This chapter reviews the main concepts and approaches of data-driven modelling, which is based on computational intelligence and machine-learning methods. A brief overview of the main methods – neural networks, fuzzy rule-based systems and genetic algorithms, and their combination via committee approaches – is provided along with hydrological examples and references to the rest of the book.

Keywords Data-driven modelling · data mining · computational intelligence · fuzzy rule-based systems · genetic algorithms · committee approaches · hydrology

2.1 Introduction

Hydrological models can be characterised as physical, mathematical (including lumped conceptual and distributed physically based models) and empirical. The latter class of models, in contrast to the first two, involves mathematical equations that are not derived from physical processes in the catchment but from analysis of time series data. Examples include the unit hydrograph method, linear regression and ARIMA models. Recent developments in computational intelligence, in the area of machine learning in particular, have greatly expanded the capabilities of empirical modelling. The field which encompasses these new approaches is called data-driven modelling (DDM). As the name suggests, DDM is based on analysing the data about a system, in particular finding connections between the system state variables (input, internal and output variables) without explicit knowledge of the physical behaviour

D. Solomatine
UNESCO-IHE Institute for Water Education, P.O. Box 3015, 2601 DA Delft, The Netherlands

L.M. See
School of Geography, University of Leeds, Woodhouse Lane, Leeds, LS2 9JT, UK

R.J. Abrahart
School of Geography, University of Nottingham, Nottingham, NG7 2RD, UK

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

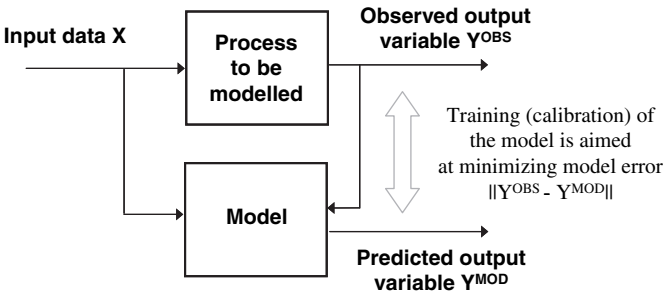


Fig. 2.1 General approach to modelling

of the system. These methods represent large advances on conventional empirical modelling and include contributions from the following overlapping fields:

- artificial intelligence (AI), which is the overarching study of how human intelligence can be incorporated into computers.
- computational intelligence (CI), which includes neural networks, fuzzy systems and evolutionary computing as well as other areas within AI and machine learning.
- soft computing (SC), which is close to CI, but with special emphasis on fuzzy rule-based systems induced from data.
- machine learning (ML), which was once a sub-area of AI that concentrates on the theoretical foundations used by CI and SC.
- data mining (DM) and knowledge discovery in databases (KDD) are focused often at very large databases and are associated with applications in banking, financial services and customer resources management. DM is seen as a part of a wider KDD. Methods used are mainly from statistics and ML.
- intelligent data analysis (IDA), which tends to focus on data analysis in medicine and research and incorporates methods from statistics and ML.

Data-driven modelling is therefore focused on CI and ML methods that can be used to build models for complementing or replacing physically based models. A machine-learning algorithm is used to determine the relationship between a system's inputs and outputs using a training data set that is representative of all the behaviour found in the system (Fig. 2.1).

Once the model is trained, it can be tested using an independent data set to determine how well it can generalise to unseen data. In the next section, the main DDM techniques are discussed.

2.2 An Overview of Data-Driven Modelling Techniques

This section describes the most popular computational intelligence techniques used in hydrological modelling, including neural networks, fuzzy rule-based systems, genetic algorithms, as well as approaches to model integration.

2.2.1 Neural Networks

Neural networks are a biologically inspired computational model, which is based on the way in which the human brain functions. There has been a great deal written on this subject; see, e.g. Beale and Jackson (1990) and Bishop (1995). Neural network models are developed by training the network to represent the relationships and processes that are inherent within the data. Being essentially non-linear regression models, they perform an input–output mapping using a set of interconnected simple processing nodes or neurons. Each neuron takes in inputs either externally or from other neurons and passes it through an activation or transfer function such as a logistic or sigmoid curve. Data enter the network through the input units arranged in what is called an input layer. These data are then fed forward through successive layers including the hidden layer in the middle to emerge from the output layer on the right. The inputs can be any combination of variables that are thought to be important for predicting the output; therefore, some knowledge of the hydrological system is important.

The hidden layer is the essential component that allows the neural network to learn the relationships in the data as shown originally in Rumelhart et al. (1986); these authors popularised also the backpropagation algorithm for training a feed-forward neural network but the principle was first developed by Werbos in 1974 (see Werbos, 1994). This configuration is also referred to as a multilayer perceptron (MLP) and it represents one of the most commonly used neural networks (Kasabov, 1996). The backpropagation algorithm is a variation of a gradient descent optimisation algorithm that minimises the error between the predicted and actual output values. The weighted connections between neurons are adjusted after each training cycle until the error in the validation data set begins to rise. The validation data set is a second data set that is given to the network to evaluate during training. If this approach is not used, the network will represent the training data set too well and will then be unable to generalise to an unseen data set or a testing data set. Once the networks are trained to satisfaction, it can be put to operation when the new input data are passed through the trained network in its non-training mode to produce the desired model outputs. In order to validate the performance of the trained network before it is put into real operation, however, the operation mode is usually imitated by using the test data set. An important way to help promote generalisation to unseen data is to ensure that the training data set contains a representative sample of all the behaviour in the data. This could be achieved by ensuring that all three data sets – training, validation and test – have similar statistical properties. Note that the test set cannot be used to change the properties of the trained model.

The use of ANNs has many successful applications in hydrology, in modelling rainfall-runoff processes: Hsu et al. (1995); Minns and Hall (1996); Dawson and Wilby (1998); Dibike et al. (1999); Abrahart and See (2000); Govindaraju and Ramachandra Rao (2001); replicating the behaviour of hydrodynamic/hydrological models of a river basin where ANNs are used to provide optimal control of a reservoir (Solomatine and Torres, 1996); building an ANN-based intelligent controller

for real-time control of water levels in a polder (Lobbrecht and Solomatine, 1999); and modelling stage-discharge relationships (Sudheer and Jain, 2003; Bhattacharya and Solomatine, 2005).

Section two of the book specifically deals with neural network applications in different areas of hydrology. In Chap. 3, Abrahart and See provide a guide to neural network modelling in hydrology. The next six chapters are applications of neural networks to rainfall-runoff modelling. Dawson demonstrates how neural networks can be used to estimate floods at ungauged catchments (Chap. 4). Jain decomposes the flood hydrograph into subsets and trains a neural network on each individual subset (Chap. 5). Coulibaly considers the behaviour of neural networks on nonstationary time series (Chap. 6). See et al. examine the behaviour of the hidden neurons of a neural network as a way of providing physical interpretation of a neural network rainfall-runoff model (Chap. 7). De Vos and Rientjes look at the effects of modifying the objective function used by neural network rainfall-runoff models (Chap. 8), while Toth considers the influence on flow forecasts when changing the temporal resolution of the input and output variables (Chap. 9). The final two chapters are applications of neural networks in groundwater (Mohammadi, Chap. 10) and sediment modelling (White et al., Chap. 11).

2.2.2 Fuzzy Rule-Based Systems (FRBS)

Fuzzy rule-based systems use fuzzy logic for inference. Fuzzy logic is based on fuzzy set theory in which binary set membership has been extended to include partial membership ranging between 0 and 1 (Zadeh, 1965). Fuzzy sets, in contrast to their crisp counterparts, have gradual transitions between defined sets, which allow for the uncertainty associated with these concepts to be modelled directly. After defining each model variable with a series of overlapping fuzzy sets, the mapping of inputs to outputs can be expressed as a set of IF-THEN rules, which can be entirely specified from expert knowledge, or from data. However, unlike neural networks, fuzzy models are prone to a rule explosion, i.e. as the number of variables or fuzzy sets per variable increases, there is an exponential increase in the number of rules, which makes it difficult to specify the entire model from expert knowledge alone (Kosko, 1997). Different automated methods for optimising fuzzy models are now available (Wang, 1994), including neural networks and genetic algorithms.

The fuzzy sets and rules are referred to as the fuzzy model knowledgebase. Crisp inputs to the model are first fuzzified via this knowledgebase, and a fuzzy inference engine is then used to process the rules in parallel via a fuzzy inference procedure such as max-min or max-product operations (Jang et al., 1997). The fuzzy solution surface resulting from the execution of the rulebase is defuzzified to produce the system output(s). Fuzzy IF-THEN rules can also be comprised of functional consequents, usually of a linear or polynomial form, in a formulation referred to as a TSK model (Takagi and Sugeno, 1985; Sugeno and Kang, 1988). The crisp inputs are fuzzified according to the fuzzy set definitions, combined via the inference engine,

and the functional consequents are weighted by the memberships that result from the execution of the rules. The overall result is a weighted average of the equations as more than one rule can fire positively during a single pass of the rulebase.

Fuzzy logic has found multiple successful applications, mainly in control theory (see, e.g. Kosko, 1997). As mentioned previously, fuzzy rule-based systems can be built by interviewing human experts, or by processing historical data and thus forming a data-driven model. The basics of the latter approach and its use in a number of water-related applications can be found in Bárdossy and Duckstein (1995). FRBS were effectively used for drought assessment (Pesti et al., 1996), prediction of precipitation events (Abebe et al., 2000), analysis of groundwater model uncertainty (Abebe et al., 2000), control of water levels in polder areas (Lobbrecht and Solomatine, 1999) and modelling rainfall-discharge dynamics (Vernieuwe et al., 2005).

Part III of the book deals specifically with fuzzy systems applications in hydrology. Mujumdar provides an overview of fuzzy logic-based approaches in water resource systems (Chap. 12). Examples of fuzzy rule-based flood forecasting models are then presented by Bardossy (Chap. 13) and Jacquin and Shamseldin (Chap. 14), while Cluckie et al. (Chap. 15) consider the use of an adaptive neuro-fuzzy inference system in the development of a real-time flood forecasting expert system. Finally, the section ends with Chap. 16 by Makropoulos et al. who examine the use of fuzzy inference for building hydrological decision support systems.

2.2.3 Genetic Algorithms (GAs) in Model Optimisation

Genetic algorithms (GAs) (or, more widely, evolutionary algorithms) are non-linear search and optimisation methods inspired by the biological processes of natural selection and survival of the fittest (Goldberg, 1989). They do not belong to the class of data-driven models, but since they are widely used in optimising models, we consider them here as well. Genetic (evolutionary) algorithms are typically attributed to the area of computational intelligence.

Unlike other methods such as hillclimbing and simulated annealing, a GA, like other randomised search algorithms such as Adaptive Cluster Covering (Solomatine, 1999), exhibits implicit parallelism, considering many points at once during the search process and thereby reduces the chance of converging to a local optimum. GAs also use probabilistic rules in the search process, and they can generally outperform conventional optimisation techniques on difficult, discontinuous and multimodal functions. Despite their unique and adaptive search capabilities, there is no guarantee that GAs will find the global solution; however, they can often find an acceptable one quite quickly. A detailed introductory survey can be found in Reeves and Rowe (2003).

The basic unit of a GA is the gene, which in biological terms represents a given characteristic of an individual, such as eye colour. In a GA, a gene represents a parameter that is being optimised. An individual or chromosome is simply the combined set of all the genes, i.e. all the parameters needed to generate the solution. To

start the search, a population of these individuals or strings is randomly generated. Each string is then evaluated by a fitness or objective function according to some measure of performance. This represents the success of the solution and is analogous to the survival ability of an individual within the population. In order to evolve better performing solutions, the fittest members of the population are selected and exposed to a series of genetic operators, which produce offspring for the next generation. The least fit solutions, on the other hand, will die out through natural selection as they are replaced by new, recombined individuals.

The main genetic operator is crossover in which a position along the bit string is randomly chosen that cuts two parent chromosomes into two segments, which are then swapped. The new offspring are comprised of a different segment from each parent and thereby inherit bits from both. The occurrence of crossover is determined probabilistically; when crossover is not applied, offspring are simply duplicates of the parents, thereby giving each individual a chance of passing on a pure copy of its genes into the gene pool. The second main genetic operator is mutation, which is applied to each of the offspring individually after crossover. Mutation can alter the bits in a string, but with an extremely low probability. Crossover allows the genetic algorithm to explore new areas in the search space and gives the GA the majority of its searching power while mutation exploits existing areas to find a near optimal solution and essentially provides a small amount of random search to ensure that no point in the search space has a zero probability of being examined. The newly generated offspring are then placed back into the population, and the exercise is repeated for many generations until a set of user-specified termination criteria are satisfied, such as exceeding a preset number of generations or if no improved solution is found after a given period of time. Over many generations, a whole new population of possible solutions, which possess a higher proportion of the characteristics found in the fitter members of the previous generation, is produced.

GAs are a very useful tool for handling difficult problems where conventional techniques cannot cope, or alternatively, they can be used to improve existing methods through hybridisation. For example, fuzzy logic rule-based models can be entirely optimised by a GA in a completely inductive approach, or expert knowledge can be used to specify the rules or membership functions, leaving the GA to optimise only the unknown parts of the model. See Cordón and Herrara (1995) and Karr (1991) for more details of fuzzy logic model optimisation using a GA. GAs can be also used to optimise other data-driven models like neural networks (Yao and Liu, 1997); this approach was also used by Parasuraman and Elshorbagy (Chap. 28 in this volume).

Part IV of the book is devoted to examples of hydrological optimisation by genetic algorithms. Savic, in Chap. 17, provides an overview of global and evolutionary optimisation in hydrology and water management problems. Tsai considers the use of a GA in a groundwater problem (Chap. 18). Efstratiadis and Koutsoyiannis (Chap. 19) and Khu et al. (Chap. 20) look at two different multi-objective versions of evolutionary optimisation algorithms for model calibration (in the latter chapter, in order to reduce the number of model runs, a meta-model for the error surface approximation is used). Jain in Chap. 21 considers the calibration of a hydrological

model using real-coded GAs. In Chap. 22, Solomatine and Vojinovic compare a series of different global optimisation algorithms for model calibration. In the final chapter of this section by Heppenstall et al., a cooperative co-evolutionary approach is demonstrated that evolves neural network rainfall-runoff models.

2.2.4 Other Approaches

In addition to neural networks and fuzzy rule-based systems, there are other data-driven methods that have been used successfully to solve hydrological problems. These methods are still less known if compared to ANN and FRBS, and the chapters covering these are in Part V entitled "Emerging Technologies". Methods that are currently finding a lot of researchers' attention are listed below:

- Genetic Programming and evolutionary regression
- Chaos theory and non-linear dynamics
- Support vector machines

Additionally, machine-learning methods for clustering and classification are often used to support regression methods considered above, as well as methods of instance-based learning (IBL), used for both classification and regression.

Genetic programming (GP) is a method for evolving equations by taking various mathematical building blocks such as functions, constants and arithmetic operations and combining them into a single expression and was originally developed by Koza (1992). Evolutionary regression is similar to GP but the goal is to find a regression equation, typically a polynomial regression, where the coefficients are determined through an evolutionary approach such as a GA. Examples of hydrological applications include the work by Khu et al. (2001), who applied GP to real-time runoff forecasting for a catchment in France, and Giustolisi and Savic (2006) who used evolutionary regression for ground water and river temperature modelling.

Classification is a method for partitioning data into classes and then attributing data vectors to these classes. The output of a classification model is a class label, rather than a real number like in regression models. The classes are typically created such that they are far from one another in attribute space but the points within a class are as tightly clustered around the centre point as possible. Examples of classification techniques include *k*-nearest neighbour, Bayesian classification, decision tree classification (Witten and Frank, 2000) and support vector machines (SVM) (Vapnik, 1998).

There are many examples of applying classification methods in hydrology. For example, Frapporti et al. (1993) used fuzzy c-means clustering to classify shallow Dutch groundwater sites, Hall and Minns (1999) used a SOFM to classify catchments into subsets on which ANNs were then applied to model regional flood frequency. Hannah et al. (2000) used clustering for finding groups of hydrographs on the basis of their shape and magnitude; clusters are then used for classification by experts. Harris et al. (2000) applied clustering to identify the classes of river regimes.

Velickov et al. (2000) used self-organising feature maps (Kohonen networks) as clustering methods, and SVM as a classification method in aerial photograph interpretation with the purpose of subsequent construction of flood severity maps. Solomatine et al. (2007) used decision trees and *k*-NN in classification of river flow levels according to their severity in a flood forecasting problem in Nepal. Zhang and Song (2006) used a combination of SOF and ART networks for special pattern identification of soil moisture. In this volume, Parasuraman and Elshorbagy (Chap. 28) used clustering before applying ANNs to forecasting streamflow.

In *instance-based learning* (IBL), classification or prediction is made by combining observations from the training data set that are close to the new vector of inputs (Mitchell, 1997). This is a local approximation and works well in the immediate neighbourhood of the current prediction instance. The nearest neighbour classifier approach classifies a given unknown pattern by choosing the class of the nearest example in the training set as measured by some distance metric, typically Euclidean. Generalisation of this method is the *k*-nearest neighbour (*k*-NN) method. For a discrete valued target function, the estimate will just be the most common value among k training examples nearest to $\mathbf{x}_q$. For real-valued target functions, the estimate is the mean value of the *k*-nearest neighbouring examples. Locally weighted regression (LWR) is a further extension in which a regression model is built on *k*-nearest instances.

Applications of IBL in water-related problems mainly refer to the simplest method, viz *k*-NN. Karlsson and Yakowitz (1987) showed the use of this method in hydrology, focusing however only on (single-variate) time series forecasts. Galeati (1990) demonstrated the applicability of the *k*-NN method (with the vectors composed of the lagged rainfall and flow values) for daily discharge forecasting and favourably compared it to the statistical ARX model. Shamseldin and O'Connor (1996) used the *k*-NN method for adjusting the parameters of the linear perturbation model for river flow forecasting. Toth et al. (2000) compared the *k*-NN approach to other time series prediction methods in a problem of short-term rainfall forecasting. Solomatine et al. (2007) considered IBL in a wider context of machine learning and tested their applicability in short-term hydrologic forecasting.

Chaos theory and non-linear dynamics can be used for time series prediction when the time series data are of sufficient length and carry enough information about the behaviour of the system (Abarbanel 1996). The main idea is to represent the state of the system at time t by a vector in m-dimensional state space. If the original time series exhibits chaotic properties, then its equivalent trajectory in phase space has properties allowing for accurate prediction of future values of the independent variable. Hydrological examples include the work by Solomatine et al. (2000) and Velickov et al. (2003), who used chaos theory to predict the surge water level in the North Sea close to Hook of Holland. For two-hourly predictions, the error was as low as 10 cm and was at least on par with the accuracy of hydrodynamic models. Babovic et al. (2000) used a chaos theory-based approach for predicting water levels at the Venice lagoon, and Phoon et al. (2002) for forecasting hydrologic time series.

Support vector machines (SVM) is a relatively new important method based on the extension of the idea of identifying a line (or a plane or some surface) that separates two classes in classification. It is based on statistical learning theory initiated by V. Vapnik in the 1970s (Vapnik, 1998). This classification method has also been extended to solving prediction problems, and in this capacity was used in hydrology-related tasks. Dibike et al. (2001) and Liong and Sivapragasam (2002) reported using SVMs for flood management and in prediction of water flows and stages. Chapter 26 by Yu et al. provides a recent example of flood stage forecasting using SVM.

2.3 Combination and Integration of Models

2.3.1 Modular Models

Since natural processes are complex, it is sometimes not possible to build a single global model that adequately captures the system behaviour. Instead the training data can be split into a number of subsets, and separate specialised models can be built on each subset. These models are called local or expert models, and this type of modular model is sometimes called a committee machine (CM) (Haykin, 1999). Two key decisions must be made when building a CM. The first is how to split the data and the second is how to combine the individual models to produce a final output.

The group of statistically driven approaches with "soft" splits of input space is represented by *mixtures of experts* (Jordan and Jacobs, 1995), *bagging* (Breiman, 1996) and *boosting* (Freund and Schapire, 1997).

Another quite popular approach is to build an *ensemble* of models and to combine the model results by some averaging scheme; this approach is widely used in meteorology.

Yet another group of methods does not combine the outputs of different models but explicitly uses only one of them, i.e. the most appropriate one (a particular case when the weights of other expert models are zero). Such methods use "hard" splits of input space into regions. Each individual local model is trained individually on subsets of instances contained in these regions, and finally the output of only one specialised expert is taken into consideration. This can be done manually by experts on the basis of domain knowledge. Another way is to use information theory to perform such splits and to perform splitting progressively; examples are decision trees, regression trees, MARS (Breiman et al., 1984) and M5 model trees (Quinlan, 1992).

These machine-learning techniques use the following idea: split the parameter space into areas (subspaces) and build a separate regression model in each of them. Tree-based models are constructed by a divide-and-conquer method. The set T is either associated with a leaf, or some test is chosen that splits T into subsets corresponding to the test outcomes and the same process is applied recursively to the

subsets. If models in the leaves are of zero order (numeric constants) then this model is called a regression tree (Breiman et al., 1984); if the models are of first order (linear regression models) then the model is referred to as an M5 model tree (Quinlan 1992; "M5" stands for "Model trees, version 5"). The splitting criterion in both algorithms is based on treating the standard deviation of the output values that reach a node as a measure of the error at that node, and calculating the expected reduction in this error as a result of testing each attribute at that node. Solomatine and Dulal (2003) used M5 model trees in rainfall-runoff modelling of a catchment in Italy.

Note that to denote a combination of models (or modular models), various authors use different terms: in machine learning these are typically mixtures of experts and committee machines; when other models are combined the term "data fusion" is often used – see, for example, an earlier chapter by Abrahart and See (2002) where six alternative methods to combine data-driven and physically based hydrologic models were compared.

Two chapters in Part V of the book deal with such modular approaches that lately are becoming more and more popular. Solomatine starts off the part in Chap. 24 with an overview of modular models. Stravs et al. then provide an example of precipitation interception modelling using M5 model trees (Chap. 25).

It is also possible to use a combination of models in a given solution. If these models work together to create a single solution they are referred to as *hybrid models*. If, on the other hand, this combination of models is not used to model the same process but instead they work with each other, then this combination is referred to as a *complementary model*. Examples of hybrid models include a study by See and Openshaw (2000) where several types of models were combined using an averaging scheme, a Bayesian approach and two fuzzy logic models; the combination of physically based models using a fuzzy model (Xiong et al., 2001); and the combination of data-driven models of various types trained on subsets of the original data set (Solomatine and Xue, 2004). Examples of complementary models include updating a physically based model using a neural network (Shamseldin and O'Connor, 2001; Lekkas et al., 2001; Abebe and Price, 2004). Solomatine et al. (2007) built an ANN-based rainfall-runoff model where its outputs were corrected by an instance-based model.

2.3.2 Integration of Models

The final section of the book deals specifically with model integration and different hydrological examples. The focus here is on the technological developments in the area of model integration, i.e. the integration of models and a variety of data sources. The chapters by Fortune and Gijsbers and by Werner describe the architectures of modern model integration frameworks (OpenMI and Delft-FEWS, respectively). The chapter by Xuan and Cluckie addresses the issue of uncertainty propagation in the integrated model including numerical weather prediction and

hydrologic components. Betts et al. describe an integrated modelling framework implemented for the Yangtze River basin in China. Finally, the chapter by O'Kane addresses the issue of incorporation of data into models and of "social calibration" of models – involving stakeholders with the best knowledge of the aquatic system in question, rather than purely numerical calibration without an insight – which is especially important when models are to be used in education and in real-life-decision-making frameworks. An extensive study of flooding in the polder landscape of the Lower Feale catchment in Ireland is used as illustration of the principle.

2.4 Conclusions

Data-driven modelling and computational intelligence in general have proven their applicability to various water-related problems: modelling, short-term forecasting, data classification, reservoir optimisation, building flood severity maps based on aerial or satellite photos, etc. Data-driven models would be useful in solving a practical problem or modelling a particular system or process if (1) a considerable amount of data describing this problem is available; (2) there are no considerable changes to the modelled system during the period covered by the model. Such models are especially effective if it is difficult to build knowledge-driven simulation models (e.g. due to lack of understanding of the underlying processes), or the available models are not adequate enough. It is of course always useful to have modelling alternatives and to validate the simulation results of physically based models with data-driven ones, or vice versa.

The developers and users of data-driven models should realise that such models typically do not really represent the physics of a modelled process; they are just devices used to capture relationships between the relevant input and output variables. However, such devices could be more accurate than process models since they are based on objective information (i.e. the data), and the latter may often suffer from incompleteness in representing the modelled process.

A contemporary trend is to combine data-driven models, i.e. combining models of different types and which follow different modelling paradigms (thus constituting *hybrid* models), including the combination with physically based models in an optimal way. One of the challenges for hydroinformatitians in this respect is to ensure that data-driven models are properly incorporated into the existing modelling and decision support frameworks.

References

Abarbanel HDI (1996) Analysis of Observed Chaotic Data. Springer-Verlag: New York.

Abebe AJ, Solomatine DP, Venneker R (2000) Application of adaptive fuzzy rule-based models for reconstruction of missing precipitation events. Hydrological Sciences Journal 45(3): 425–436.

Abebe AJ, Guinot V, Solomatine DP (2000) Fuzzy alpha-cut vs. Monte Carlo techniques in assessing uncertainty in model parameters. Proc. 4th Int. Conference on Hydroinformatics, Cedar Rapids.

Abebe AJ, Price RK (2004) Information theory and neural networks for managing uncertainty in flood routing. ASCE Journal of Computing in Civil Engineering 18(4): 373–380.

Abrahart RJ, See L (2000) Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecast in two contrasting catchments. Hydrological Processes 14: 2157–2172.

Abrahart RJ, See L (2002) Multi-model data fusion for river flow forecasting: an evaluation of six alternative methods based on two contrasting catchments. Hydrology and Earth System Sciences 6(4): 655–670.

Babovic V, Keijzer M, Stefansson M (2000) Optimal embedding using evolutionary algorithms. Proc. 4th Int. Conference on Hydroinformatics, Cedar Rapids.

Bárdossy A, Duckstein L (1995) Fuzzy Rule-Based Modeling with Applications to Geophysical, Biological and Engineering Systems. CRC press Inc: Boca Raton, Florida, USA.

Beale R, Jackson T (1990) Neural Computing: An Introduction, Adam Hilger: Bristol.

Bhattacharya B, Solomatine DP (2005) Neural networks and M5 model trees in modelling water level – discharge relationship. Neurocomputing 63: 381–396.

Bishop CM (1995) Neural Networks for Pattern Recognition. Clarendon Press: Oxford.

Breiman L, Friedman JH, Olshen RA, Stone CJ (1984) Classification and regression trees. Wadsworth International: Belmont.

Breiman L (1996) Stacked regressor. Machine Learning 24(1): 49–64.

Cordón O, Herrara F (1995) A general study on genetic fuzzy systems. In: Winter G, Périaux J, Gálan M, Cuesta P (eds) Genetic Algorithms in Engineering and Computer Science. John Wiley & Sons, Chichester, pp. 33–57.

Dawson CW, Wilby R (1998) An artificial neural network approach to rainfall-runoff modelling. Hydrological Sciences Journal 43(1): 47–66.

Dibike Y, Solomatine DP, Abbott MB (1999) On the encapsulation of numerical-hydraulic models in artificial neural network. Journal of Hydraulic Research 37(2): 147–161.

Dibike YB, Velickov S, Solomatine DP, Abbott MB (2001) Model induction with support vector machines: introduction and applications. ASCE Journal of Computing in Civil Engineering 15(3): 208–216.

Frapporti G, Vriend SP, Van Gaans PFM (1993) Hydrogeochemistry of the shallow Dutch groundwater: classification of the national groundwater quality monitoring network. Water Resources Research, 29(9): 2993–3004.

Freund Y, Schapire R (1997) A decision-theoretic generalisation of on-line learning and an application of boosting. Journal of Computer and System Science 55(1): 119–139.

Galeati G (1990) A comparison of parametric and non-parametric methods for runoff forecasting. Hydrology Sciences Journal 35(1): 79–94.

Giustolisi O, Savic DA (2006) A symbolic data-driven technique based on evolutionary polynomial regression. Journal of Hydroinformatics 8(3): 202–207.

Goldberg DE (1989) Genetic Algorithms in Search Optimisation and Machine Learning. Addison-Wesley: USA.

Govindaraju RS, Ramachandra Rao A (eds) (2001) Artificial Neural Networks in Hydrology. Kluwer: Dordrecht.

Hall MJ, Minns AW (1999) The classification of hydrologically homogeneous regions. Hydrological Sciences Journal 44: 693–704.

Hannah DM, Smith BPG, Gurnell AM, McGregor GR (2000) An approach to hydrograph classification. Hydrological Processes 14: 317–338.

Harris NM, Gurnell AM, Hannah DM, Petts GE (2000) Classification of river regimes: a context for hydrogeology. Hydrological Processes 14: 2831–2848.

Haykin S (1999) Neural Networks: A Comprehensive Foundation. McMillan: New York.

Hsu KL, Gupta HV, Sorooshian S (1995) Artificial neural network modelling of the rainfall-runoff process. Water Resources Research 31(10): 2517–2530.

Jang J-S, Sun C-T, Mizutani E (1997) Neuro-Fuzzy and Soft Computing. Prentice Hall.
Jordan MI, Jacobs RA (1995) Modular and hierarchical learning systems. In: Arbib M (ed) The Handbook of Brain Theory and Neural Networks. MIT Press, Cambridge.
Karr CL (1991) Genetic algorithms for fuzzy logic controllers. AI Expert 6: 26–33.
Karlsson M, Yakowitz S (1987) Nearest neighbour methods for non-parametric rainfall runoff forecasting. Water Resources Research 23(7): 1300–1308.
Kasabov K (1996) Foundations of Neural Networks, Fuzzy Systems and Knowledge Engineering. MIT Press: Cambridge.
Khu S-T, Liong S-Y, Babovic V, Madsen H, Muttil N (2001) Genetic programming and its application in real-time runoff forecasting, Journal of the American Water Resources Association 37(2): 439–451.
Kosko B (1997) Fuzzy engineering. Prentice-Hall: Upper Saddle River.
Koza JR (1992) Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press: Cambridge, MA
Lekkas DF, Imrie CE, Lees MJ (2001) Improved non-linear transfer function and neural network methods of flow routing for real-time forecasting. Journal of Hydroinformatics 3(3): 153–164.
Liong SY, Sivapragasam C (2002) Flood stage forecasting with SVM. Journal of American Water Resources Association 38(1): 173–186.
Lobbrecht AH, Solomatine DP (1999) Control of water levels in polder areas using neural networks and fuzzy adaptive systems. In: Savic D, Walters G (eds) Water Industry Systems: Modelling and Optimization Applications. Research Studies Press Ltd., Baldock, pp. 509–518.
Minns AW, Hall MJ (1996) Artificial neural network as rainfall-runoff model. Hydrological Sciences Journal 41(3): 399–417.
Mitchell TM (1997) Machine Learning. McGraw-Hill: New York.
Pesti G, Shrestha BP, Duckstein L, Bogárdi I (1996) A fuzzy rule-based approach to drought assessment. Water Resources Research 32(6): 1741–1747.
Phoon KK, Islam MN, Liaw CY, Liong SY (2002) A practical inverse approach for forecasting nonlinear hydrological time series. ASCE Journal of Hydrologic Engineering, 7(2): 116–128.
Quinlan JR (1992) Learning with continuous classes. In: Adams A, Sterling L (eds) Proc. AI'92, 5th Australian Joint Conference on Artificial Intelligence. World Scientific, Singapore, pp. 343–348.
Reeves CR, Rowe JE (2003) Genetic Algorithms – Principles and Perspectives. A Guide to GA Theory, Kluwer Academic Publishers Group.
Rumelhart D, Hinton G, Williams R (1986) Learning internal representations by error propagation. In: Rumelhart D, McClelland J, (eds) Parallel Distributed Processing: Explorations in the microstructure of cognition. Volume 1: Foundations. MIT Press, Cambridge, MA, pp. 318–363.
See LM, Openshaw S (2000) A hybrid multi-model approach to river level forecasting. Hydrological Sciences Journal 45: 523–536.
Shamseldin AY, O'Connor KM (1996) A nearest neighbour linear perturbation model for river flow forecasting. Journal of Hydrology 179: 353–375.
Shamseldin AY, O'Connor KM (2001) A non-linear neural network technique for updating of river flow forecasts. Hydrology and Earth System Sciences 5 (4): 557–597.
Solomatine DP, Torres LA (1996) Neural network approximation of a hydrodynamic model in optimizing reservoir operation. Proc. 2nd Int. Conference on Hydroinformatics, Balkema: Rotterdam, 201–206.
Solomatine DP (1999) Two strategies of adaptive cluster covering with descent and their comparison to other algorithms. Journal of Global Optimization 14(1): 55–78.
Solomatine DP, Rojas C, Velickov S, Wust H (2000) Chaos theory in predicting surge water levels in the North Sea. Proc. 4th Int. Conference on Hydroinformatics, Cedar-Rapids.
Solomatine DP, Dulal KN (2003) Model tree as an alternative to neural network in rainfall-runoff modelling. Hydrological Sciences J. 48(3): 399–411.
Solomatine DP, Xue Y (2004) M5 model trees and neural networks: application to flood forecasting in the upper reach of the Huai River in China. ASCE J. Hydrologic Engineering 9(6): 491–501.

Solomatine DP, Maskey M, Shrestha DL (2007) Instance-based learning compared to other data-driven methods in hydrologic forecasting. Hydrological Processes, 21 (DOI: 10.1002/hyp.6592).

Sudheer KP, Jain SK (2003) Radial basis function neural network for modeling rating curves. ASCE Journal of Hydrologic Engineering 8(3): 161–164.

Sugeno M, Kang GT (1988) Structure identification of fuzzy model, Fuzzy Sets and Systems 28(1): 15–33.

Takagi T, Sugeno M (1985) Fuzzy identification of systems and its applications to modeling and control. IEEE Transactions on Systems, Man, and Cybernetic SMC-15: 116–132.

Toth E, Brath A, Montanari A (2000) Comparison of short-term rainfall prediction models for real-time flood forecasting. Journal of Hydrology 239: 132–147.

Vapnik VN (1998) Statistical Learning Theory. Wiley & Sons: New York.

Velickov S, Solomatine DP, Yu X, Price RK (2000) Application of data mining techniques for remote sensing image analysis. Proc. 4th Int. Conference on Hydroinformatics, USA.

Velickov S, Solomatine DP, Price RK (2003) Prediction of nonlinear dynamical systems based on time series analysis: issues of entropy, complexity and predictability. Proc. of the XXX IAHR Congress, Thessaloniki, Greece.

Vernieuwe H, Georgieva O, De Baets B, Pauwels VRN, Verhoest NEC, De Troch FP (2005) Comparison of data-driven Takagi–Sugeno models of rainfall–discharge dynamics. Journal of Hydrology 302(1–4): 173–186.

Wang LX (1994) Adaptive Fuzzy Systems and Control: Design and Stability Analysis. PTR Prentice Hall Inc.: Englewood Cliffs, NJ.

Werbos PJ (1994) The Roots of Backpropagation. NewYork: John Wiley & Sons (includes Werbos's 1974 Ph.D. thesis, Beyond Regression).

Witten IH, Frank E (2000) Data Mining. Morgan Kaufmann: San Francisco.

Xiong LH, Shamseldin AY, O'Connor KM (2001) A non-linear combination of the forecasts of rainfall–runoff models by the first-order Takagi-Sugeno fuzzy system. Journal of Hydrology 245(1–4): 196–217.

Yao X, Liu Y (1997) A new evolutionary system for evolving artificial neural networks. IEEE Transactions on Neural Networks 8(3): 694–713.

Zadeh LA (1965) Fuzzy sets. Information and Control 8: 338–353.

Zhang X, Song X (2006) Spatial pattern identification of soil moisture based on self-organizing neural networks. Proc. 7th Intern. Conf on Hydroinformatics, Nice, September.

Part II
Artificial Neural Network Models

Chapter 3
Neural Network Hydroinformatics: Maintaining Scientific Rigour

R.J. Abrahart, L.M. See and C.W. Dawson

Abstract This chapter describes the current status of neural network hydrological modelling. Neural network modelling is now a regular feature in most peer-reviewed hydrological and water resource publications. The number of reported operational models is, nevertheless, restricted to a small handful of diverse implementations located in different parts of the world. The social and institutional reasons for this fundamental mismatch are discussed, and a requirement for stronger scientific rigour in modelling and reporting is highlighted. Eight potential guidelines for the development of a stronger scientific foundation are provided.

Keywords Neural network · scientific method · hydrological modelling

3.1 Black Art or Hard Science?

Flooding is a major worldwide issue; effective warning systems can save lives and prevent unnecessary damage. Such systems are also the cheapest safeguarding option since major flood prevention works are very expensive and not always successful. The forecasting and modelling of river regimes to deliver effective management and sustainable use of water resources is also of particular concern in regions that experience water shortages. It is important to harness modern technologies that can be used to improve the capture, storage and release of hydrological assets. The artificial neural network (NN) offers untapped opportunities that could benefit hydrological modellers, both now and in the future. The number of reported operational implementations, however, remains limited to a handful of specialist situations and

R.J. Abrahart
School of Geography, University of Nottingham, University Park, Nottingham, NG7 2RD, UK

L.M. See
School of Geography, University of Leeds, Woodhouse Lane, Leeds, LS2 9JT, UK

C.W. Dawson
Department of Computer Science, Loughborough University, Leicestershire, LE11 3TU, UK

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

the socio-theoretical reasons for this knowledge transfer gap must be appreciated; for a water treatment example of the practical difficulties involved in implementing an operational model, see Zhang et al. (2004a,b).

Hydrological modelling is a procedure wherein one or more phases of the hydrological cycle are represented using a simplified scheme or model. Traditional hydrological modelling is part art and part science. Execution of the modelling process is considered to span both philosophical domains since the model designer must combine established knowledge of observed physical processes with conceptual representations of those unknown principles that control the mechanisms which are being modelled (James, 1970). The latter will include making appropriate decisions on the use of internal components and measurement records and the manner in which such items are interconnected. The challenge in most cases is to find a meaningful relationship between the two approaches. Hydroinformatics is also part art and part science. It offers a strong medium through which both approaches can be combined. It is a branch of informatics that uses information and communication technologies to address important problems related to the equitable and efficient use of water for many different purposes. The numerical simulation of water flows and related processes remains a central component, with a strong focus being placed on technological developments and the appropriate application of related tools in a social context.

Hydroinformatics has a strong interest in the use of techniques originating from the field of artificial intelligence. NN modelling is a major component; support vector machines and evolution-based algorithms form two other related components. Hybrid approaches are also popular. Such tools might be used with large collections of observed data sets for the purposes of data mining and knowledge discovery, or on output data sets generated from existing conceptual or distributed-process models, for the purposes of generating a more efficient and effective emulator, e.g. providing a faster or more robust version of the original model for operational purposes. The nature of the modelling procedures involved is often less dependent on the development or improvement of conceptual relationships and instead maintains a strong focus on the execution of complex computational procedures. However, perhaps more important would be the different underlying reasons for producing a model since such matters will inform decisions on the nature of the route that is to be followed. It is axiomatic that hydrological models can be developed to serve a number of different purposes, and two main competing objectives can be identified: (i) the provision of accurate forecasts and predictions for operational management purposes; and (ii) the development of models for scientific reasons such as hypothesis testing operations or to assist in the process of knowledge acquisition through the mechanisms of model construction and development.

The last decade and a half has witnessed a major upsurge in the trialling and testing of NN modelling procedures across a broad range of different disciplines. Time series forecasting has been a particular focus of interest with efficient and effective models having been developed in several different areas of hydrological science. The extent of their hydrological application is vast, with reported implementations that range from traditional rainfall-runoff modelling operations (Riad

et al., 2004) and groundwater level prediction (Giustolisi and Simeone, 2006) to the regional downscaling of climate change predictions (Dibike and Coulibaly, 2004). Numerous extended summaries and reviews exist, and the interested reader is referred to the following: Maier and Dandy (2000), American Society of Civil Engineers (2000a,b), Dawson and Wilby (2001) and Rajurkar et al. (2004). Two edited compilations have also been published: Govindaraju and Rao (2000) and Abrahart et al. (2004).

Daniell (1991) provides the first recorded paper on the use of NN tools for hydrological modelling, listing 10 potential applications in hydrology and water resources. Two illustrative examples were also offered: (i) a 'security of supply' investigation that predicted per capita water consumption demands based on climate data sets; and (ii) a 'regional flood estimation procedure' that used catchment parameters to estimate flood frequency distributions, i.e. log Pearson Type 3 Average Recurrence Interval. The scientific press has thereafter experienced an incremental growth in hydrological publications related to the development and application of experimental solutions. Important papers in this process were the pioneering explorations of French et al. (1992) on the emulation of a space–time mathematical model that produced simulated rainfall patterns, Minns and Hall (1996) on the emulation of a conceptual model with a nonlinear reservoir that produced simulated discharge sequences and Hsu et al. (1995) on modelling the rainfall-runoff relationship for a medium-size river in the USA.

NNs provide a novel and appealing solution to the problem of relating input variables to output variables for the purposes of hydrological modelling. It is important to establish from the outset that NN models are no more 'neural' than physical models are 'physical': the inspiration behind an artificial neural network is biological; physical models are inspired from observed processes; both types of model can be implemented in the manner of hardware (electro-mechanical surrogates) or software (computer code); and it is axiomatic that model builders should opt to adopt the relevant terminologies that stem from their originating fields – methodological similarities, subject estrangement and the potential for outright confusion notwithstanding. These tools offer efficient and effective solutions for modelling and analysing the behaviour of complex dynamical systems. The broad range of traditional modelling operations, to which neural solutions could be applied, can be expressed in terms of the following scientific procedures: function approximation, pattern recognition, classification and processing, e.g. filtering, smoothing, compression.

Zhang et al. (2004b) highlight several advantages related to the use of such tools in environmental science and engineering. The power to represent a set of complex nonlinear relationships that are contained in an input–output data set, but without the need for a priori knowledge on the exact nature of such relationships, is a major factor. The power of such tools to generalise to unseen data sets is another. Several other properties distinguish neural solutions from standard algorithmic or rule-based approaches. The power and skill to discover relationships from data sets is important. Each model is trained to represent the implicit but nonetheless concealed internal relationships that exist within a given data set. Individual networks offer great modelling flexibility: their capacities to extract and represent different levels

of generalisation are important; likewise their capabilities to operate on different types and mixed types of input or output material, e.g. nominal data; fractal dimension; use of binaries to mark the start and end of rainfall (Hall and Minns, 1993); use of additional outputs that preserve or monitor global features, related to the overall structure, or to particular aspects of a data set (French et al., 1992); use of alternative representations that produce better modelling solutions, e.g. Fourier series parameters instead of actual hydrological variables (Smith and Eli, 1995). Numerous potential opportunities for scientific activities related to other, perhaps lesser appreciated, properties might also be exploited: e.g. self-organised development, faster processing speeds and their 'ability to scale', i.e. a generic solution can be used to support larger or smaller data sets, or could be increased or decreased in size, or operational processing capabilities, to meet different challenges.

Each solution will also possess inherent operational advantages related to the use of parallel computational structures and distributed memories that are realised in terms of fault-tolerant properties such as 'robustness' and 'graceful degradation'. To provide a robust solution each model must exhibit a consistent or stable behaviour and be insensitive to potential uncertainties in the construction and parameterisation process, e.g. problems related to measurements that cannot be obtained with sufficient accuracies or in the calibration of relationships that do not remain constant over long(er) periods. To be reliable and trusted an operational model must also exhibit the properties of 'graceful degradation'; a gradual and progressive reduction in overall performance such that the model continues to operate and function in a normal manner, but provides a reduced level of service, as opposed to taking incorrect actions or crashing, i.e. total stoppage of processing activities occurs. NN applications can also be developed and deployed quickly and easily with very little programming owing to the existence of a number of user-friendly software packages and ongoing research into model protocol development. Moreover, for flood forecasting purposes, neural solutions and generic packages offer practical advantages related to operational costs and socio-economic resources that would be of potential interest in developing countries: e.g. parsimonious requirements, rapid development process and open source code (Shamseldin, in press).

However, all that glisters is not gold. NN tools are not perfect, can be misused, and should be seen as an adjunct to more traditional approaches. Established problems include overfitting, overtraining and chance effects (Livingstone et al., 1997). NN models develop 'data reliant' solutions: requiring representative data sets of sufficient quantities and qualities to permit the automated capture of important relationships – but offering no rules as to what might be considered a set of right or wrong inputs and outputs (Zhang et al., 2004ab; Sha, 2007; Sahoo and Ray, 2007). The act of building a neural solution is also something of a black art on two counts: (i) no fixed rules or regulations exist regarding the development of individual models; (ii) intuition and personal experience are important factors in the construction process. Each modeller applies their individual set of trial and error procedures, until at some point in the development process, it is possible to select a 'winning solution'. Reed and Marks (1999) refer to the model development process as 'neural smithing', i.e. drawing analogies to the act or art of forging and working metals,

such as iron, into a desired shape. NN development and testing operations nevertheless involve the purposeful application of recognised computational algorithms in an organised scientific manner. Thus, useful solutions will be forged; but it is not a 'black art' since no magical spells, that harness occult forces or evil spirits, to produce the desired effects are involved! Total freedom in the model building process does not, however, generate high levels of end user trust; nor does it offer a preferred solution to a specific hydrological modelling issue that others might be willing to acknowledge as a suitable replacement for more established and accepted methods.

Moreover, in certain quarters, concerns have been expressed that further investigation into the use of such tools might prove to be a hydrological modelling culde-sac (Wilby et al., 2003, p. 164). NN models are difficult to interpret and offer no explanations for their answers; most neural investigations have also (to date) done little or nothing to build upon existing hydrological knowledge, or to provide greater understanding of hydrological processes per se. It is perhaps logical to expect some degree of natural distrust in solutions that are not transparent, not based on theoretical support, and offer no statements regarding the reasons for their answers. Trust is nevertheless about a set of individual beliefs:

- in the scientific principles and modelling approaches that were used to develop a model;
- in the scientists and programmers who constructed it; and
- in the institutions or companies that market it.

Established reputations and personal acquaintances with conventional tools and methods are thus an important part of the 'confidence equation', whereas poor understanding and limited knowledge of strange and unfamiliar products will erect barriers to progress. It is also questionable as to whether or not the scientific application of neural solutions should be 'demonised through association' with the ideological shortfalls of black box modelling. This chapter is not the place to present a philosophical or historical account on the nature and potential benefits of black box modelling. However, the physical relationship that exists between observed hydrological processes and behaviours, and the internal components of a neural solution is a topic of great interest: see, for example, the chapter in this volume by See et al. (2008), or the reported findings of Wilby et al. (2003), Jain et al. (2004), Sudheer and Jain (2004) and Sudheer (2005).

3.2 Building Stronger Foundations

NNs are at a critical stage in their 'product development cycle'. Expressed in terms of 'crossing the chasm' (Moore, 2002): the early adaptors and innovators have opened up a market; the majority now need to be catered for; some will never be convinced about the benefits of change. To meet the requirements of a different market it is argued that important adjustments are needed to further the art and science or application and practice of NN hydrological modelling. The honeymoon period

has ended! NN should no longer be viewed as novel or innovative mechanisms; peer-reviewed papers need no longer start with a prolonged introduction or attempt to explain their inner workings or related procedural issues such as the backpropagation of error algorithm. These items are explained in numerous standard texts which should henceforth be referred to in a manner similar to that used for other numerical tools and procedures in different sectors of the hydrological modelling world, e.g. conceptual or statistical modelling.

The main obstacle to wider acceptance and broader use of such tools within the different hydrological modelling communities is perhaps at this stage in the cycle related to the large number of simplistic 'button-pushing' or 'handle-churning' exercises that are still being reported – although submission rejection rates are starting to increase as the peer review process becomes more discerning. Neurohydrologists it appears are also still attempting to crack small nuts with large sledgehammers! They are applying powerful state-of-the-art tools in a traditional manner and performing limited conventional implementations, i.e. something that existing and more established, or perhaps better or equal, traditional methods are well equipped to do. In many cases, traditional inputs are used to produce traditional outputs, with no innovations being attempted. The extent to which the acclaimed advantages of such tools are exploited, or even subject to comprehensive evaluation in the hydrological sciences domain, instead remains scant. The construction of emulators and the development of modelling solutions based on a committee approach are two cases in point that are in dire need of more comprehensive testing. The content of reported applications is nevertheless now starting to be improved such that the route to publication requires a change in mindset, exhibiting a reawakening of the need and desire to explore, discover or perform fresh adventures and innovations in hydrological modelling.

Of equal concern is the fact that numerous reported applications have focused on reporting the numerical differences obtained from a set of simplistic curve-fitting exercises; different models developed on different algorithms or different architectural configurations are fitted to various data sets and an assortment of metrics is used to select some sort of 'winner'. The point of such activities is to record the level of improvement that results from comprehensive testing, although such improvements are often marginal, and of limited scientific meaning or hydrological merit. Few scientific insights are acquired and little methodological progress is achieved. This is not a purposeful activity: it lacks imagination; it lacks direction; and serves only to reinforce the status quo in that a strong attraction remains to what is perceived as being important vis-à-vis traditional roles and values. It also provides a depressed or diminished view of potential developments in this field. The eventual outcome of such activities is also open to disapproval and bad press: perhaps, for example, a neural approach based on traditional inputs offers limited improvement in terms of evapotranspiration estimation procedures (Koutsoyiannis, 2007; Kisi, 2007b); but, on the other hand, if a generic solution can provide similar levels of response to a dedicated mathematical model what does that tell us about the mathematical model, i.e. in what way does the domain knowledge which is encapsulated in the mathematical model count or affect the nature of that model?

It is sometimes difficult to see what the scientific contribution of a specific paper is or to what extent that paper furthers either NN science or hydrological modelling. Lots of people are still searching for some magical number of hidden units, related to this or that particular model, which produced a better score on this or that particular data set. This operation, however, equates to a particular form of optimisation and, as might perhaps be expected, few useful findings have so far emerged. There is also a worrying trend in that most reported investigations correspond to 'local case studies' based on one or perhaps two catchments. Yet the application of innovative methodologies to one or two isolated catchments or data sets seldom, if ever, produces universal insight(s). Moreover, the result from one investigation cannot be compared with the findings of another in a direct sense since, although the specified problem might well be similar in certain respects, it is no longer identical. The specific challenge(s) in each individual scenario will be different such that it is impossible to differentiate with precision the extent to which the published result is based on the method as opposed to the 'problem situation'. Most published investigations are instead designed to unveil the potential merit(s) of some specific tool or method. Parallel comparisons are often performed against conceptual models, linear solutions or persistence forecasts, with certain advantages being highlighted, but since each product is in most cases somewhat unique no common set of rules or guidelines has resulted. Everyone is still forced to begin their opening explorations from the same initial starting point! Furthermore, most investigations are not repeated, no confirmation studies are performed on identical data sets and reported methodologies are seldom revisited or applied to a different set of catchments or data sets in a purposeful and constructive manner.

The scientific method nevertheless demands that effective comparisons are established and for confirmation studies to be performed. It forms part of a set of tools and procedures that are used (i) to acquire fresh knowledge and (ii) for the purposes of correcting or integrating a set of previous findings. The basic expectation is one of completeness: comprehensive documentation, methodologies and data sets should be made available for careful scrutinisation. Thus, independent scientists and researchers should be offered sufficient access and equal opportunities to perform detailed confirmation studies. For proper verification:

- Experiments must be repeatable: it should be possible for others to perform the same experiment.
- Findings must be reproducible: it should be possible for others to obtain the same result.

For an illustration of such activities in action, the interested reader is referred to the discussions of Aksoy et al. (2007) and Kisi (2007a). If the reported experiments cannot be repeated, or the reported findings confirmed, it also devalues the original investigation on two counts since the user is required to place excessive reliance on personal trust models (developed through direct or indirect one-on-one relationships with people and products) and each such outcome must be viewed as an isolated or individualistic 'single-case-based' result (Barlow and Hersen, 1984). The two most

important functions of single-case-based studies are as ‘awareness demonstrators’ and for the generation of ‘fresh ideas’, for later, more rigorous, experimentation.

It is also important to consider different goals. Hydrological modelling requires consistent measures of merit and trust. Hillel (1986, p. 42) advocated that hydrological modelling solutions should be ‘parsimonious’ – each model should contain a minimum number of parameters that can be measured in the field; ‘modest’ – the scope and purpose to which a specific model can be applied must not be overstated; ‘accurate’ – the correctness of the forecast or prediction need not be better than the correctness of the input measurements; and ‘testable’ – the limits within which the model outputs are valid can be defined. Most papers are focused on one aspect of merit and trust: the production of more accurate outputs. However, as mentioned earlier, other qualities and issues are also important with respect to practical operational implementations. Yet mechanistic properties such as ‘robustness’ and ‘graceful degradation’ will not in all cases have an optimal relationship with model output accuracies and so must be treated as independent properties that impart a set of constraints. Environmental modelling investigations into the changing nature of NN outputs related to the provision of degraded inputs are reported for hydrological forecasting in Abrahart et al. (2001) and for sediment transfer in Abrahart and White (2001). For more detailed discussion on the requirements constraint issue the interested reader is referred to Alippi (2002).

From this bewildering situation of different targets and trajectories arises a pressing need for fundamental experiments that can help settle substantive issues within the hydrological sciences. Koutsoyiannis (2007) is to be commended for providing a spirited account of hydrological concerns that beset the wider acceptance of NN approaches among scientists and practitioners. The need for two different sorts of publication are identified: (i) papers that assess their hydrological limitations or usefulness under differing circumstances; and (ii) papers that engage with or provide counter reasoning about specific published points (e.g. Abrahart and See, 2007a,b). The authors must present their findings in terms of its contribution to the advancement of either science or engineering. The nature of the hydrological issues involved should be clear; as should the hydrological interpretation and meaning of their reported findings. Each investigation should encompass a worthwhile topic of research and be something that is of wider interest to an international scientific audience. Each paper should address a specific scientific hypothesis or research question. Each paper must explain to what extent a particular set of research questions have been answered. It should discuss the extent to which general principles can be extracted from the reported experimental results, or point to a requirement for subsequent experiments. In short – it must be clear what has been discovered. It must state what important lessons have been learnt. It must relate individual outcomes or some superior performance measure for the best solution to specific hydrological characteristics of different test data sets. It is likewise clear that poor results should not be ‘swept under the carpet’.

Han et al. (2007, p. 223) commented that the large number of unsolved questions continues to hinder the application of such tools among practising hydrologists. This is best exemplified by considering the limited number of operational

applications that have been developed. A final challenge surrounding the use of NN methods is that such tools are relatively new and vastly different from conventional approaches. As such, it is often summarily dismissed by experimenters who do not wish to apply new approaches, even in cases where the use of a specific set of technologies has been proven. The continued testing and evaluation of different technologies and toolsets, in conjunction with an increased number of reported successful applications, will enable this challenge to be addressed. The rest of this chapter attempts to engage in one particular aspect of that debate through the development of some testing and reporting guidelines that could support NN hydrological modelling.

3.3 Future Reporting Requirements

The present challenge is to build on our initial reported successes. Borrowing from the vernacular – it equates to 'crossing the chasm'; bridging the gulf that arises between innovators and early adopters on the one hand and mainstream users on the other. There are a number of different ways in which this can be achieved. The traditional process for disseminating information about the latest tools and techniques is twofold: (i) students are trained in the most recent methods; (ii) the acclaimed virtues of different methods are substantiated through meaningful investigations reported in high-quality journals, some of which are now open-access resources. The literature is full of NN hydrological modelling papers as more and more people start to experiment with this technological toolbox of potential solutions and to learn different ways of realising its untapped potential. NN hydrological modelling has, however, reached the end of the beginning: strong technological barriers and steep learning curves have both been defeated – such that harder scientific questions can henceforth be tackled from an informed position. Fresh entrants will continue to encounter one major 'stumbling block' that is often referred to in terms of an apparent contradiction: 'the only rule is that there are no rules'. To address this point various attempts have been made to provide some basic hydrological modelling guidelines on the development process: e.g. Dawson and Wilby (1998) and Maier and Dandy (2000). The main challenges in the next phase of this progressive transition to mainstream acceptance can be put into two groups: to establish significant questions that will help push back the boundaries of science and to exploit, or delimit, the various benefits and drawbacks that are on offer. Thus, at issue is the requirement for published papers to report on substantive matters. Each paper must offer meaningful scientific or operational contributions to hydrological modelling, i.e. simple demonstrations or curve-fitting comparisons applied to different, perhaps unique, data sets are no longer sufficient. Each paper should instead possess at least the following desirable characteristics:

1. Each paper should possess a clear aim and provide sound findings expressed in terms of an explicit contribution to hydrological science. The individual nature and overall impact of each scientific or operational contribution must be

transparent to the reader. The reported findings must be related to previous investigations in that field based on an assessment of similarities and differences in methodologies or data sets. Each paper must describe the manner in which its contribution moves things forward. It must go above and beyond existing material that is already reported in the literature. Papers that report the simple application of similar tools to a different catchment are no longer innovative; papers that report identical findings on similar or identical data sets should be viewed as confirmation studies and published as supporting material in appropriate outlets. The conclusions should be related to the original aims and objectives of each individual paper and provide some meaningful consideration of hydrological factors or catchment properties. The conclusions must also summarise the main contribution of that paper to either scientific understanding or hydrological modelling procedures per se.

2. Each paper must provide a comprehensive account of current and previous studies on that topic. It must also be a critical review; there is a trend in recent papers to provide nothing other than a long string of references related to previous reported applications or other relevant material. The cited papers are doubtless related in some manner or other to the topic of investigation, but it is important to explain what was being studied in each previous paper and to highlight the main similarities and differences involved, i.e. between the studies that are being cited and the one that is being reported. It is likewise important to discuss the potential range of alternative approaches to solving particular problems. This will enable current research to be placed in context and in so doing demonstrate that the reported investigation both draws upon existing knowledge and thereafter expands it.
3. Each paper should provide a hydrological description of the study area(s) and modelling data set(s). The extent to which each data set and subset that is used in the modelling process offers (or handicaps) a comprehensive or meaningful representation of the processes that are being modelled is important. This issue is of particular concern with small data sets that have a higher potential for sampling bias, i.e. the chances of obtaining a poorer overall representation are increased. Each hydrological modelling subset must be described using a range of different numerical indicators; popular descriptive statistics would include minimum, maximum, mean, standard deviation, skewness and kurtosis. The analysis should also be extended to include a consideration of input drivers as well as output predictands. The full details for each catchment should include a description of important hydrological factors and a map that shows the location of the different stations that were used in the modelling exercise. It is important for the results to be contextualised in terms of hydrological considerations related to different data sets and pertinent catchment properties, e.g. strong snowmelt component.
4. Models and approaches should be tested over a range of conditions to establish the level of reliance that can be placed on individual results. Identical methods and procedures should be used to develop models on multiple catchments or data sets since the uniqueness of a given site or set of measurement records

is important. It is possible that different findings could result from different catchments or data sets, and it is essential to ascertain the extent to which findings are consistent across individual hydrological scenarios and different scales of interest. It is also the case that one-step-ahead forecasts equate to near linear modelling situations. This is not that challenging. If no second catchment is available it is possible to extend the forecasting horizon and offer a comparison of one-step-ahead (near linear) and four-step-ahead (more nonlinear) solutions. It would also be good to test the relationship between model generalisation capabilities and model skill under different situations. For example, are the developed models performing better on certain types of catchment? It would also be desirable to consider transferring a model developed on one catchment to a different catchment or to experiment with models developed on multiple catchments to discover to what extent a common solution can be obtained.

5. Each paper should provide a detailed report on pre-project planning operations. This would include reporting on the type of model that was to be applied and providing a justification for that selection. It would also need to offer some reasons for choosing the final architectural configuration and, if applicable, provide some basic details on experiments performed using different architectures. The range of results from such experiments would provide some indication of the extent to which the final model is in fact superior to other potential models, i.e. is the selection process significant or perhaps the result of random operators and non-significant differences? It is also important to establish if a standard software package was used and if so which one, e.g. SNNS, TRAJAN, MATLAB. If the authors developed their own software program it is important to determine to what extent that program has been subjected to alpha or beta testing, or tested against other programs, and thus ascertain to what extent the reported software outputs can be trusted. It would be good in such cases if the source code or model could be supplied on the web for cross testing.
6. Each paper must provide a full account of the operational procedures involved. Elaboration on the methods used for pre-processing the data sets, prior to training, as well as post-processing of the output data set is required. Papers should explain in detail how the main drivers were selected, providing theoretical or numerical justification for that selection. The choice of data sampling and reasoning behind the splitting of data into different data sets for training, validation and testing must be explained and justified – with particular reference as to whether or not the data set that is used for model testing purposes provides a fair and rigorous evaluation of its overall capabilities. It must also provide a full report on the training programme; to include a consideration of the activities that were used to produce the required optimal or sub-optimal solution; a consideration of the procedures that were used to prevent overfitting or underfitting; and a consideration of the computational algorithm that was employed to adjust the weights. Full details and a justification for each item should be included. Important factors such as parameter settings, optimisation criteria, stopping criteria, and the pattern of weighted connections should also be reported and discussed even if a set of defaults options are used.

7. Testing should include the use of multiple performance statistics and comparisons. Several recognised evaluation metrics are available and can be used to assess different aspects of model skill, peak flow, drought conditions, total throughput, etc. HydroTest can be used to calculate several different performance statistics as discussed in Dawson et al. (2007) [www.hydrotest.org.uk]. It is also essential that model performance be evaluated in terms of qualitative techniques such as visual inspection of output hydrographs. This permits more detailed interpretation and hydrological insight than is otherwise possible with the use of traditional global statistics. It is also important to perform a comparison against other models. NN results should be compared with different mechanisms such as conceptual models or other state-of-the-art solutions. Linear regression comparisons should also be performed so as to evaluate the extent of the nonlinearities involved. It is not otherwise possible to establish the relative performance of the NN model, in terms of how challenging a specific application is, since it is somewhat illogical to fit a complex nonlinear solution to a near linear problem and expect the outputs to be vastly better than that produced using a simple linear model. The fact that a neural solution can adapt to different situations is important, but the potential demand for such modelling complexities remains an open question. The need to establish measures of confidence or uncertainty remains a major challenge since models that appear to perform well when judged using a set of global performance statistics may also be highly uncertain in their predictions.
8. The data sets and models used should be provided online in an open-access environment. This will empower other researchers to repeat published experiments and reproduce identical confirmations, providing formal benchmarks against which subsequent research can be compared. The power of the web can also be harnessed in other ways; published papers in open-access (or traditional) journals offer highlighted snippets of the actual results. If scientists and practitioners could be encouraged to store all modelling inputs and outputs in open-access repositories, such activities would ensure maximum impact across the board. The speed at which the science and practice of a discipline is advanced would be accelerated; duplication of effort would be avoided; gaps in understanding would be highlighted; numerical comparisons would be made easier, thus effectively and efficiently reusing published results; and everything would be grounded to a common framework of investigation. Lack of access to essential resources would no longer be an impeding factor; full open access would facilitate meta-analysis of models and their outputs for knowledge discovery and extraction.

3.4 Final Thoughts

Hard science involves tackling difficult problems; it is not about taking simple options or shortcuts; it is not about performing numerous automated curve-fitting exercises. It requires the development of purposeful solutions, in complex situations,

or under challenging conditions. Hard science relies on the use of experimental procedures, quantifiable data sets and the scientific method. The focus is on producing accurate and objective findings, on producing results that can be rigorously tested and proven. NN hydrological modelling has progressed to the point at which the mere existence of a neural solution to a particular problem is no longer interesting. Hard science is required. Long-term interest in related research can only be sustained if neural applications (i) can be shown to outperform conventional algorithms, exhibiting clear superiorities on important aspects in specific fields of investigation; (ii) can compete on a different basis through doing non-standard tasks that will permit their true powers to be exploited; (iii) can do the same things but in a manner that requires other qualities to be present in an operational solution, e.g. robustness; (iv) or can be used in situations where attempting a specific procedure using other means is either not possible or not worthwhile. The eight guidelines suggested above should help towards sustaining interest in this field of research.

Hydroinformatics also has a strong social component. Hydrologists have a natural reluctance to share their data sets; the present problem of needing to run NN models on multiple catchments offers NN modellers the chance to help overcome one major barrier to progress by taking a lead role in the formation of a 'collective modelling environment' comprising open-access repositories for international data sets and archived model outputs. Placing digital deliverables on a download site for other scientists to draw upon would ensure maximum impact of individual outputs and provide a common ground for subsequent investigations as opposed to the current slow and somewhat piecemeal approach that is limited to the publication of key findings in journals.

NN models could also be put on the web in a real-time environment – if practitioners could be encouraged to use such tools in the field then perhaps wider acceptance might follow. There is no great copyright issue in terms of the final product; no intellectual property rights are tied up in software coding; and the provision of transparent open source neural solutions computational algorithms and procedures in a generic solution would of course lead to and thus better all-round science (Harvey and Han, 2002).

To conclude, it is important to stress that future challenges for neural modellers are neither simple nor straightforward. The fight for recognition and funding is not over; it is just starting. Hard science awaits us and sceptics abound. Two immortal quotations nevertheless spring to mind:

> We choose to go to the moon in this decade, and do the other things – not because they are easy; but because they are hard; because that goal will serve to organize and measure the best of our energies and skills; because that challenge is one that we're willing to accept; one we are unwilling to postpone, and one we intend to win – and the others, too.
>
> *President John F. Kennedy, Rice University Speech, 13 Sept. 1962*

> We shall not fail or falter; we shall not weaken or tire … Give us the tools and we will finish the job.
>
> *Sir Winston Churchill, BBC radio broadcast, 9 Feb. 1941*

References

Abrahart RJ, See LM (2007a) Neural network emulation of a rainfall-runoff model. Hydrology and Earth System Sciences Discussions 4: 287–326

Abrahart RJ, See LM (2007b) Neural network modelling of non-linear hydrological relationships. Hydrology and Earth System Sciences 11(5): 1563–1579

Abrahart RJ, Kneale PE, See LM (2004) Neural Networks for Hydrological Modelling. Rotterdam, Taylor & Francis

Abrahart RJ, White SM (2001) Modelling sediment transfer in Malawi: Comparing backpropagation neural network solutions against a multiple linear regression benchmark using small data sets. Physics and Chemistry of the Earth (B) 26(1): 19–24

Abrahart RJ, See LM, Kneale PE (2001) Applying saliency analysis to neural network rainfall-runoff modelling. Computers and Geosciences 27: 921–928

Aksoy H, Guven A, Aytek A, Yuce MI, Unal NE (2007) Discussion of "Generalized regression neural networks for evapotranspiration modelling" by O Kisi. Hydrological Sciences Journal 52(4): 825–828

Alippi C (2002) Selecting accurate, robust, and minimal feedforward neural networks. IEEE Transactions on Circuits and Systems – I: Fundamental Theory and Applications 49(12): 1799–1810

American Society of Civil Engineers [Task Committee on Application of Artificial Neural Networks in Hydrology] (2000a) I: Preliminary Concepts. Journal of Hydrological Engineering 5(2): 115–123

American Society of Civil Engineers [Task Committee on Application of Artificial Neural Networks in Hydrology] (2000b) Artificial neural networks in hydrology. II: Hydrologic applications. Journal of Hydrological Engineering 5(2): 124–137

Barlow DH, Hersen M (1984) Single Case Experimental Designs. 2nd Edition. Needham Heights, MA: Allyn and Bacon

Dawson CW, Abrahart RJ, See LM (2007) HydroTest: a web-based toolbox of evaluation metrics for the standardised assessment of hydrological forecasts. Environmental Modelling & Software 22(7): 1034–1052

Dawson CW, Wilby R (1998) A artificial neural network approach to rainfall-runoff modelling. Hydrological Sciences Journal 43(1): 47–66

Dawson CW, Wilby RL (2001) Hydrological modelling using artificial neural networks. Progress in Physical Geography 25(1): 80–108

Daniell TM (1991) Neural networks – applications in hydrology and water resources engineering. Proceedings, International Hydrology and Water Resources Symposium, Vol. 3, 797-802, Nat. Conf. Publ. 91/22, Inst. of Eng., Australia, Barton, ACT, Australia

Dibike YB, Coulibaly P (2004) Temporal neural networks for downscaling climate variability and extremes. Neural Networks 19(2): 135–144

French MN, Krajewski WF, Cuykendall RR (1992) Rainfall forecasting in space and time using a neural network. Journal of Hydrology 137 (1–4): 1–31

Govindaraju RS, Rao AR (2000) Artificial Neural Networks in Hydrology. Kluwer Academic Publishers, Amsterdam.

Giustolisi O, Simeone V (2006) Optimal design of artificial neural networks by a multiobjective strategy: groundwater level predictions. Hydrological Sciences Journal 51(3): 502–523

Hall MJ, Minns AW (1993) Rainfall-runoff modeling as a problem in artificial intelligence: experience with a neural network. BHS 4th National Hydrology Symposium, Cardiff 5.51–5.57.

Han D, Kwong T, Li S (2007) Uncertainties in real-time flood forecasting with neural networks. Hydrological Processes 21(2): 223–228

Harvey H, Han D (2002) The relevance of open source to hydroinformatics. Journal of Hydroinformatics 4(4): 219–234

Hillel D (1986) Modeling in soil physics: A critical review. In: Future Developments in Soil Science Research, Soil Society of America. Madison, Wisconsin, USA, pp. 35–42

Hsu K-L, Gupta HV, Sorooshian S (1995) Artificial neural network modeling of the rainfall-runoff process. Water Resources Research 31(10): 2517–2530

Jain A, Sudheer KP, Srinivasulu S (2004) Identification of physical processes inherent in artificial neural network rainfall runoff models. Hydrological Processes 18(3): 571–581

James LD (1970) Watershed modelling: an art or a science? Paper presented at the Winter Meeting, American Society of Agricultural Engineers, Chicago. Cited in McCuen RH (1973) The role of sensitivity analysis in hydrologic modelling. Journal of Hydrology 18(1): 37–53

Kisi O (2007a) Reply to Discussion of "Generalized regression neural networks for evapotranspiration modelling" by H Aksoy et al. Hydrological Sciences Journal 52(4): 829–831

Kisi O (2007b) Reply to Discussion of "Generalized regression neural networks for evapotranspiration modelling" by D Koutsoyiannis. Hydrological Sciences Journal 52(4): 836–839

Koutsoyiannis D (2007) Discussion of "Generalized regression neural networks for evapotranspiration modelling" by O Kisi. Hydrological Sciences Journal 52(4): 832–835

Livingstone DJ, Manallack DT, Tetko IV (1997) Data modelling with neural networks: Advantages and limitations. Journal of Computer-Aided Molecular Design 11(2): 135–142

Maier HR, Dandy GC (2000) Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. Environmental Modelling & Software 15(1): 101–124

Moore GA (2002) Crossing the Chasm: Marketing and Selling High-Tech Products to Mainstream Customers. HarperCollins Publishers Inc., New York

Minns AW, Hall MJ (1996) Artificial neural networks as rainfall-runoff models, Hydrological Sciences Journal 41(3): 399–417

Rajurkar M, Kothyari U, Chaube U (2004) Modeling of the daily rainfall-runoff relationship with artificial neural network. Journal of Hydrology 285: 96–113

Riad S, Mania J, Bouchaou L, Najjar Y (2004) Predicting catchment flow in a semi-arid region via an artificial neural network technique. Hydrological Processes 18(13): 2387–2393.

Reed RD, Marks II RJ (1999) Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks., The MIT Press, Cambridge, MA

Sahoo GB, Ray C (2007) Reply to comments made on "Flow forecasting for a Hawaii stream using rating curves and neural networks" by W Sha. Journal of Hydrology 340(1–2): 122–127

See LM, Jain A, Dawson CW, Abrahart RJ (2008) Visualisation of hidden neuron behaviour in a neural network rainfall-runoff model. In: Abrahart RJ, See LM, Solomatine DP (eds) *Practical Hydroinformatics*: Computational Intelligence and Technological Developments in Water Applications. Springer-Verlag

Sha W (2007) Comment on "Flow forecasting for a Hawaii stream using rating curves and neural networks" by GB Sahoo and C Ray. Journal of Hydrology 340(1–2): 119–121

Shamseldin AY (in press) Artificial neural network model for flood forecasting in a developing country. To appear in: Journal of Hydroinformatics.

Smith J, Eli RN (1995) Neural network models of rainfall-runoff process. Journal of Water Resources Planning Management ASCE 121(6): 499–508

Sudheer KP, Jain A (2004) Explaining the internal behaviour of artificial neural network river flow models. Hydrological Processes 18(4): 833–844

Sudheer KP (2005) Knowledge extraction from trained neural network river flow models. Journal of Hydrologic Engineering 10(4): 264–269

Wilby RL, Abrahart RJ, Dawson CW (2003) Detection of conceptual model rainfall-runoff processes inside an artificial neural network. Hydrological Sciences Journal 48(2): 163–181

Zhang QJ, Cudrak AA, Shariff R, Stanley SJ (2004a) Implementing artificial neural network models for real-time water colour forecasting in a water treatment plant. Journal of Environmental Engineering and Science 3 S1: 15–S23

Zhang QJ, Smith DW, Baxter CW (2004b) Introduction. Journal of Environmental Engineering and Science 3 S1: iii–iv.

Chapter 4
Neural Network Solutions to Flood Estimation at Ungauged Sites

C.W. Dawson

Abstract Artificial neural networks are *universal approximators* that have seen widespread application in a number of fields in science, engineering, medicine and finance. Since their re-popularisation in the mid-1980s, they have been applied successfully to a number of problems in hydrology. However, despite their widespread use, these tools have only been applied in a limited number of studies to the problem of estimating flood magnitudes in ungauged catchments. Using data from the Centre for Ecology and Hydrology's Flood Estimation Handbook, this chapter aims to show how neural network models can be developed to predict 20-year flood events and the index flood (the median of the annual maximum series) for ungauged catchments across the UK. In addition, the chapter provides a discussion of how different choices in the use of the available data can significantly affect the accuracy of the models that are developed.

Keywords Artificial neural networks · flood estimation · ungauged catchments · Flood Estimation Handbook

4.1 Introduction

Although artificial neural networks (ANNs) have been applied in a number of hydrological studies since the early 1990s (for example, see the reviews of ASCE 2000; Dawson and Wilby, 2001), they have only been used in a limited number of cases for predicting flows in ungauged catchments (examples include Dastorani and Wright, 2001; Dawson et al., 2005; Dawson et al., 2006). In this chapter, this particular problem domain is used to explore how the composition of training data can have a significant impact on model performance (e.g. see Han et al., 2007) by exploring the importance of utilising all available data for network calibration. In

C.W. Dawson
Department of Computer Science, Loughborough University, Leicestershire, UK,
e-mail: C.W.Dawson1@lboro.ac.uk

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

order to achieve this aim, three experiments are discussed in which the data used for training the ANNs are systematically reduced according to previous studies. The original data set for the first set of reported experiments consists of 850 catchments (as used in Dawson et al., 2006). A second set of experiments was undertaken with a subset of these catchments consisting of 719 data points based on the Hi-Flows data of the EA (2006). Finally, a much smaller data set of 88 catchments was used as per the study of Ashfaq and Webster (2002) (based on the Institute of Hydrology's Flood Estimation Handbook).

It is not the intention of this chapter to provide a detailed overview of artificial neural networks as these details are provided in numerous other texts and papers. However, for completeness it is worth noting that the network used in this study was the popular multilayer perceptron (MLP) trained using error backpropagation. In this case, the transfer function used throughout was the sigmoid function, the learning rate was set to 0.1 and momentum was included at a level of 0.9. These values were selected based on previous experience.

The remainder of this chapter is structured as follows. Section 4.2 discusses the data that were used in this study and how these data were processed for modelling. Section 4.3 discusses the experiments that were performed and how the results of these experiments were analysed. Section 4.4 discusses the results of the experiments and Section 4.5 provides the conclusions from this work.

4.2 Data Sets

The data set used in this investigation was obtained from the Flood Estimation Handbook CD-ROM (Reed and Robson, 1999) which contains data for 1,000 catchments in mainland Britain, Northern Ireland, the Isle of Wight and Anglesey. These data were provided in three separate files for each site – file #1 contains the catchment descriptor data; file #2 contains annual maximum series (AMS) data and file #3 contains peaks-over-threshold data (not used in this study). The AMS covers a range of years, with some files containing over 100 years of data while others contain only 5 or 6 years of data. From an analysis of the data sets, 16 descriptors were chosen as predictors from the catchment descriptor files:

- DTM AREA – catchment drainage area (km^2)
- BFIHOST – base flow index
- SPRHOST – standard percentage runoff
- FARL – index of flood attenuation attributable to reservoirs and lakes
- SAAR – standard period (1961–1990) average annual rainfall (mm)
- RMED-1D – median annual maximum 1-day rainfall (mm)
- (RMED-2D) – median annual maximum 2-day rainfall (mm)
- RMED-1H – median annual maximum 1-hour rainfall (mm)
- SMDBAR – mean soil moisture deficit for 1941–1970 (mm)
- PROPWET – proportion of time when soil moisture deficit < 6 mm during 1961–1990

- LDP – longest drainage path (km)
- DPLBAR – mean distance from each node on a regular 50 m grid to the catchment outlet (km)
- ALTBAR – mean altitude of catchment above sea level (m)
- DPSBAR – mean of all inter-nodal slopes in catchment (m/km)
- ASPVAR – variation of slope directions in catchment
- URBEXT1990 – extent of urban and suburban land cover in 1990 (%).

Using the annual maximum series for each catchment, the 20-year flood event estimates for each catchment were derived based on the method of Shaw (1994) in which a Gumbel Type 1 distribution is assumed. In order to provide some consistency and accuracy in the derivation of this flood event, only those catchments with at least 10 years of AMS data were selected. This reduced the available number of catchments from 1,000 to 850 for the first stage of the modelling. In this case, the 20-year flood event is estimated as

$$Q_T = \bar{Q} + K(T)S_Q \tag{4.1}$$

where

$$K(T) = -\frac{\sqrt{6}}{\pi}\left(\gamma + \ln\ln\left[\frac{T(X)}{T(X)-1}\right]\right) \tag{4.2}$$

in which $\bar{Q}$ is the mean of the annual maximums, S_Q is the standard deviation of the annual maximums, $K(T)$ is a frequency factor, $T(X)$ is the return period in years and γ is 0.5772. For the 20-year flood event, this reduces to

$$Q_T = \bar{Q} + 1.8658S_Q \tag{4.3}$$

As noted in Dawson et al. (2006), other distributions could be used but from reported experience most distributions yield comparable results. Since the purpose of this study was to evaluate the effectiveness of neural network solutions to model 20-year flood events, it did not matter which of the comparable distributions was selected since the neural networks would in all cases be modelling a pseudo-20-year flood event. The index flood was also calculated from the AMS as the median of these data.

4.3 Experiments

Three sets of experiments were undertaken during the course of this study. First, all the available data (850 catchments) were used to train and evaluate ANNs for predicting the 20-year flood event and the index flood. A cross-validation approach was used in this part of the study whereby three data sets are used. The first data set is used to train a number of ANNs (with different numbers of hidden nodes and for different training periods or epochs) – the training data. The second data set is used

to validate the ANNs produced and select the one that is most 'accurate' according to some criteria (in this case the root mean squared error – RMSE – was used). The third data set is used for final testing of the selected ANN model – testing the model against an unseen sample of the data. From the 850 catchments available, 424 (50%) were chosen at random for training, 213 (25%) were chosen at random for validation and 213 (25%) were chosen at random for testing (this mirrors the work of Dawson et al., 2006). In this study, based on past experience, networks were trained with 3, 5, 10, 15, 20, 30 hidden nodes for between 500 and 10,000 epochs (in steps of 500).

The second set of experiments involved reducing the available 850 catchments to that recommended by the Hi-Flows data of the Environment Agency (2006). In this case, only 719 catchments were available for training and testing and were split randomly into three data sets as follows; 360 (50%) were used for training, 179 (25%) were used for validation and 180 (25%) were used for final testing.

The final set of experiments involved a much smaller data set still – consisting of 88 catchments based on the study of Ashfaq and Webster (2002). In this case, all the data were used to train the neural networks, and the results of the 'best' network are presented when evaluated against the test data of the 850-catchment approach and against the test data of the 719-catchment approach. The purpose of this experiment was to see if reasonable networks can be trained when limited data are available.

Table 4.1 summarises some statistics for selected variables from each of the training data sets used in each of the experiments. As one would expect, the 850-catchment training data set has the biggest range of catchment variables that should help during training. However, it is worth noting that the 719-catchment training data contain the largest 20-year flood event and index flood.

Table 4.1 Statistics for selected variables for training data sets

	DTMAREA (km^2)	BFIHOST	SAAR (mm)	LDP (km)	URBEXT (%)	Index flood (cumecs)	20-year flood event (cumecs)
850 catchments (424 for training)							
Min	1.07	0.18	547	2.69	0.000	0.32	0.61
Max	9,951	0.97	3,473	273.09	0.432	751.11	1,288.80
Mean	409.58	0.50	1,088	40.13	0.023	82.18	138.56
719 catchments (360 for training)							
Min	1.55	0.17	555	1.79	0.000	0.43	0.66
Max	8,234.81	0.96	2,846	265.52	0.403	951.06	1,533.94
Mean	419.43	0.49	1,076	41.54	0.027	90.72	155.47
88 catchments (all for training)							
Min	8.36	0.24	577	5.65	0.000	1.17	2.86
Max	556.56	0.90	2,808	74.93	0.377	367.24	609.94
Mean	184.24	0.49	1,084	29.74	0.034	63.82	111.82

4.3.1 Error Measures

Three error measures were used to calculate the performance of the ANNs in this study. These are calculated according to the following equations:

$$\text{MSRE} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{Q_i - \hat{Q}}{Q_i}\right)^2 \tag{4.4}$$

$$\text{CE} = 1 - \frac{\sum_{i=1}^{n}(Q_i - \hat{Q}_i)^2}{\sum_{i=1}^{n}(Q_i - \bar{Q})^2} \tag{4.5}$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(Q_i - \hat{Q}_i)^2}{n}} \tag{4.6}$$

where Q_i is the observed flood event, $\hat{Q}_i$ is the modelled flood event, $\bar{Q}$ is the mean of the observed flood events and n is the number of flood events that have been modelled. These statistics were calculated using the hydrological testing web site: HydroTest (www.hydotest.org.uk). Discussion of the relative merits of each of these measures is beyond the scope of this chapter. However, the interested reader is directed to Dawson et al. (2007) for a thorough discussion of these measures.

4.4 Results

Table 4.2 presents a summary of the results from the three experiments undertaken in this study. In the following two sections, the results of the 20-year flood event models and the index flood models are discussed in turn.

Table 4.2 ANN performance for flood events during testing

T-year	MSRE	CE	RMSE (cumecs)
20-year			
850 catchments	2.26	85.60	68.56
719 catchments	40.10	83.48	92.81
88 catchments v 850	1.43	60.31	113.80
88 catchments v 719	3.95	64.70	135.65
Index flood			
850 catchments	1.98	90.48	34.16
719 catchments	38.91	90.76	41.49
88 catchments v 850	5.41	73.37	57.14
88 catchments v 719	21.86	69.46	75.42

4.4.1 Twenty-Year Flood Event Models

From the 850-catchment models, the most accurate ANN according to the validation data set was one trained for 1,800 epochs with five hidden nodes (more discussion of why this parsimonious model was selected can be found in Dawson et al., 2006). The accuracy of this model when evaluated with the test data is shown in Table 4.2. From the 719-catchment models, the most accurate ANN according to the validation data was a network with 10 hidden nodes trained for 8,000 epochs. Finally, the 'best' ANNs produced using the 88-catchment data were one with 30 hidden nodes trained for 1,000 epochs (evaluated using the 850-catchment test data) and one with five hidden nodes trained for 3500 epochs (evaluated using the 719-catchment test data).

Table 4.2 shows that the 850-catchment model is the most accurate according to the CE and RMSE error measures with the 719-catchment in second place. What is somewhat surprising is the accuracy of the 88-catchment model when evaluated using the 850-catchment test data according to the MSRE statistic. In this case the 88-catchment is the most accurate while the 719-catchment returns a very poor performance. Looking at these results graphically – Fig. 4.1 shows a scatter plot of the ANN modelled 20-year flood event versus the derived 20-year flood event during testing for the 719-catchment trained model. The graph appears to show reasonable approximation by the ANN model across all catchments except for one outlier – the River Findhorn at Forres which appears to be significantly underestimated by the ANN. The River Findhorn is a medium-sized catchment (781 km^2) with an urban extent of 0.0001% (base flow index: 0.434; average annual rainfall: 1,065 mm; longest drainage path: 100.13 km; mean slope: 119.83 m/km; and has extensive blanket peat cover that drains the Monadhliath Mountains). Thirty-two years of AMS data were available for this catchment and it is classified as *natural* (Institute of Hydrology, 1993). In these circumstances, one would expect the observed data to be sufficient to provide a reasonable estimation of the 20-year flood event. A similar catchment to this – the River Dee at Polhollick – with an area of 697 km^2 and

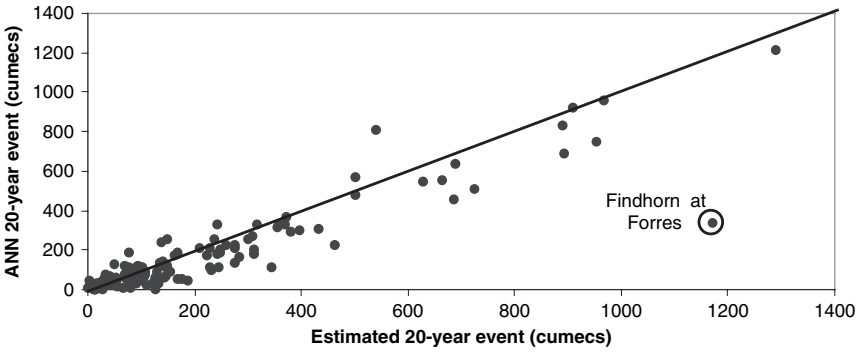


Fig. 4.1 ANN 20-year flood events modelled; 719 catchments used

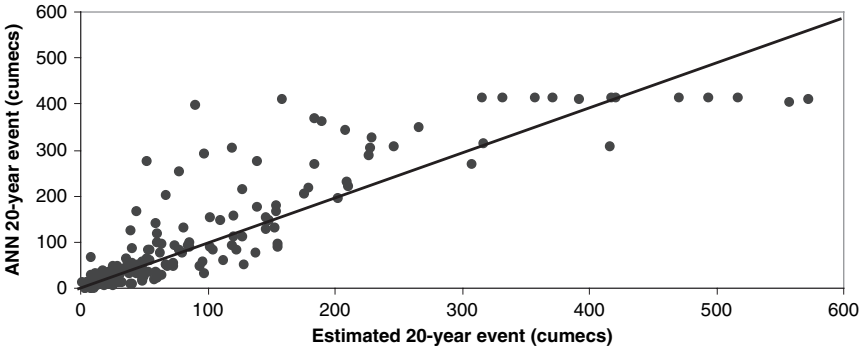


Fig. 4.2 ANN 20-year flood events modelled; 88 catchments used

urban extent of 0.0001% (base flow index: 0.458; average annual rainfall: 1,231 mm; longest drainage path: 62.68 km; mean slope: 224.44 m/km; and described as being a mountain, moorland and pastoral catchment) has a 20-year flood event of 501 cumecs compared with 1,171 cumecs for the River Findhorn. This is also described by the Hydrometric Register as *natural* and thus provides a good comparison of the flood magnitude that might be expected.

Figure 4.2 shows a comparable chart to Fig. 4.1 but for the ANN trained using the smaller 88-catchment data set. While this model produces a lower MSRE statistic than the other two ANN models, the figure shows that it produces significant errors in the prediction of higher flood events and seems to have an upper limit of around 400 cumecs. This is not surprising if one examines the (small) training data set. In this training set, the maximum 20-year flood event available is 609.94 cumecs – around half the amount of the 850-catchment training data (1,288.80 cumecs). The 88-catchment ANN is therefore only 'used to' smaller events and this perhaps explains why it has a relatively small MSRE statistic.

4.4.2 Index Flood Models

For the index flood from Table 4.2 it can be seen that the most accurate of the three models is the 850-catchment-based ANN (according to all three error measures). This is followed by the 719-catchment-based ANN (according to the CE and RMSE statistics), although this has a relatively poor MSRE value compared with the 88-catchment-based ANN.

An examination of the training data statistics (Table 4.1) perhaps shows why the 719-catchment model returns a relatively poor MSRE statistic for both the 20-year flood event and the index flood. For the 719-catchment model, the mean of the 20-year flood event for the training data is 155.47 cumecs. This is higher than the mean for the 850-catchment training data and the 88-catchment data. The 719-catchment

model is thus been trained using relatively higher flood events and thus has more difficulty modelling lower events during testing. This leads to a relatively poor MSRE statistic which penalises errors at lower levels. The same explanation applies to the index flood which also shows a much higher mean value for the training data than either of the other two training sets.

4.5 Conclusions

In this chapter, we have explored the use of different-sized data sets in the development of ANN models for predicting 20-year flood events and the index flood in UK catchments based on 16 predictor variables. The results show that limiting the available data can have an adverse affect on the accuracy of the ANNs developed. In addition, the 'quality' of the data that are used (for example, the range of events covered) for training also affects the accuracy of the models developed. One should therefore utilise as much of the available data as possible in the development of such models which are, after all, heavily data dependent. Further work is needed into investigating ways of speeding up training by reducing data sets in a more appropriate way that does not limit their information content. This might involve clustering techniques that enable smaller cross-section samples to be taken from the entire data set while still retaining the information available.

References

ASCE (2000) 'Artificial neural networks in hydrology. I: Preliminary concepts', Journal of Hydrologic Engineering, Vol 5(2), pp. 115–123.

Ashfaq, A. and Webster, P. (2002) 'Evaluation of the FEH Rainfall-Runoff Method for Catchments in the UK', Journal of the Chartered Institution of Water and Environmental Management, Vol 16, pp. 223–228.

Dastorani, M.T. and Wright, N. (2001) 'Application of Artificial Neural Networks for Ungauged Catchments Flood Prediction', Presented at Floodplain Management Association conference, San Diego, CA, March 2001.

Dawson, C.W. and Wilby, R.L. (2001) 'Hydrological modelling using artificial neural networks', Progress in Physical Geography, Vol 25(1), pp. 80–108.

Dawson, C.W. Abrahart, R.J. Shamseldin, A.Y. Wilby, R.L. and See, L.M. (2005) 'Neural network solutions to flood estimation at ungauged sites', Geophysical Research Abstracts, Vol 7, European Geosciences Union, General Assembly, Vienna, Austria, 24–29 April.

Dawson, C.W. Abrahart, R.J. Shamseldin, A.Y. and Wilby, R.L. (2006) 'Flood estimation at ungauged sites using artificial neural networks', Journal of Hydrology, Vol 319, pp. 391–409.

Dawson, C.W. Abrahart, R.J. and See, L.M. (2007) 'HydroTest: a web-based toolbox of evaluation metrics for the standardised assessment of hydrological forecasts', Environmental Modelling and Software, 27, 1034–1052.

Environment Agency (2006) 'Hi-Flows UK Homepage', <http://www.environment-agency.gov.uk/hiflowsuk/>, (accessed 24 July 2006).

Han, D. Kwong, T. and Li, S. (2007) 'Uncertainties in real-time flood forecasting with neural networks', Hydrological Processes, 21, 223–228.
Institute of Hydrology (1993) 'Hydrological data UK', Wallingford, UK.
Reed, D.W. and Robson, A.J. (1999) 'Flood estimation handbook', Vol 3, Centre for Ecology and Hydrology, UK.
Shaw, E.M. (1994) 'Hydrology in practice' (3rd edition), Chapman and Hall, London.

Chapter 5
Rainfall-Runoff Modelling: Integrating Available Data and Modern Techniques

S. Srinivasulu and A. Jain

Abstract The rainfall-runoff process is a highly complex, nonlinear, and dynamic physical mechanism that is extremely difficult to model. In the past, researchers have used either conceptual or systems theoretic techniques in isolation. This chapter presents an approach that combines data and techniques to develop integrated models of the rainfall-runoff process. Specifically, results from five different models are presented and discussed. The results obtained are encouraging and demonstrate the need to make special efforts in rainfall-runoff modelling in order to achieve improved results.

Keywords Hydrologic modelling · data decomposition · conceptual modelling · genetic algorithms · neural networks.

5.1 Background

Water is essential for all kinds of life on Earth. The total available water is estimated to be about 1386 million cubic kilometers (MKm^3), out of which only 10.6 MKm^3 is available as fresh water on land, and the rest is contained either in the oceans (97%) or in the form of frozen ice on mountain tops and glaciers (Subramanya, 1997). The fresh liquid water that is useful for human beings is available as surface water in rivers and reservoirs or as groundwater in aquifers. The total available amount of fresh liquid water has remained constant over the years, but water demands are increasing at a rapid rate due to population growth, economic developments, urbanization, and other factors. Thus, if the available water resource is not utilized efficiently and effectively, the demand for water will soon exceed its available supply. A key component in the planning, development, design, operation, and management

S. Srinivasulu
Department of Civil Engineering, Indian Institute of Technology Kanpur, Kanpur – 208 016, UP, INDIA

A. Jain
Department of Civil Engineering, Indian Institute of Technology Kanpur, Kanpur – 208 016, UP, INDIA, e-mail: ashujain@iitk.ac.in

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

of water resources is the availability of accurate river flow forecasts. Runoff forecasts are normally made through the development of runoff forecast models that use only hydrological data, or through rainfall-runoff models that use both hydrological and meteorological data. In this chapter, the literature in the area of rainfall-runoff modelling is reviewed, and a new methodology is proposed that integrates the available data with modern techniques for the purpose of achieving improved accuracies in runoff forecasts.

5.2 Rainfall-Runoff Modelling

The rainfall-runoff process is an extremely complex physical process that is not understood clearly (Zhang and Govindaraju, 2000). The approaches used for rainfall-runoff modelling cover a wide range of methods from black-box models to very detailed deterministic/conceptual models (Porporato and Ridolfi, 2001). The deterministic/conceptual models need a thorough understanding of the physics involved, a large amount of data for calibration and validation purposes, and are computationally demanding. There is an abundance of literature available in the area of rainfall-runoff modelling using deterministic/conceptual methods. Recently, artificial neural networks (ANNs) have been proposed as efficient tools for modelling complex physical systems. The application of ANNs to the field of rainfall-runoff modelling, which is popularly known as hydroinformatics, started in the 1990s. Karunanithi et al. (1994) applied the NN approach to flow prediction for the Huron River and compared its performance with that of an analytical nonlinear power model in terms of accuracy and convenience. Hsu et al. (1995) found ANNs to be more efficient and provided a better representation of the rainfall-runoff process in the Leaf River basin near Collins, Mississippi, USA, in comparison to the linear ARMAX time series and the Conceptual Rainfall-Runoff (CRR) models. Smith and Eli (1995) and Minns and Hall (1996) used a synthetic domain and data for predicting hourly flows. Tokar and Johnson (1999) investigated the sensitivity of the prediction accuracy to the content and length of the training data in ANN daily runoff forecasting. Jain and Indurthy (2003) performed a comparative analysis of the event-based rainfall-runoff modelling techniques and found that the ANN models consistently outperformed conventional approaches such as the unit hydrograph and regression analysis. The reported ANNs provided a better representation of an event-based rainfall-runoff process, in general, and estimated the peak discharge and time to peak discharge, in particular. Rajurkar et al. (2004) presented an approach for modelling the rainfall-runoff process by coupling a simpler linear (black box) model with an ANN model. The versatility of the approach was demonstrated with satisfactory results for data of catchments from different geographical locations.

It is clear from the reported literature that ANNs have received a great deal of attention by researchers working in the area of hydroinformatics. However, there are certain issues that either have not been explored or have been investigated only partially. This chapter aims to focus on three main issues: (a) integration of modelling techniques, (b) integration of models developed on data divided into

different classes, and (c) training issues of ANN models. In most of the studies reported so far, the techniques available for rainfall-runoff modelling, whether conceptual or based on ANNs, have been used in isolation. It may be possible to achieve better model performance by integrating the modelling techniques. Furthermore, the rainfall-runoff process is an extremely complex, nonlinear, dynamic, and fragmented physical process, and the functional relationships between rainfall and runoff can be quite different for low, medium, and high magnitudes of runoff. Some researchers have attempted to achieve improved model performance using data decomposition techniques (e.g. Zhang and Govindaraju, 2000; Abrahart and See, 2000; Hsu et al., 2002; and Anctil and Tape, 2004). Most of the studies involving data decomposing have focused on either statistical or soft decomposing techniques.

The data decomposition based on physical concepts may provide increased accuracies in runoff forecasts. The rising limb of a flood hydrograph is influenced by varying infiltration capacities, catchment storage characteristics, and the nature of the input. i.e. intensity and duration of the rainfall. On the other hand, the falling limb of a hydrograph is influenced more by the storage characteristics of the catchment and meteorological characteristics to some extent. Therefore, the use of a single ANN or a single technique to represent the input–output mapping of the whole flood hydrograph may not be as efficient and effective as compared to developing two different mappings representing the two limbs of the hydrograph. In fact, one can go a step further and divide the rising limb (or a falling limb) into different components and develop different models for different components to investigate if the prediction accuracies improve. Finally, the third issue being dealt with here is that of training of an ANN rainfall-runoff model. Most of the ANN applications reported for rainfall-runoff modelling have employed the backpropagation (BP) training algorithm proposed by Rumelhart et al. (1986) or its variations. Curry and Morgan (1997) concluded that the use of gradient search techniques, employed in the BP algorithm, often results in inconsistent and unpredictable performance of the neural networks. Many researchers have found that ANN rainfall-runoff models trained using BP method are biased towards a certain magnitude of flows (e.g. Hsu et al., 1995; Ooyen and Nichhuis, 1992; Tokar and Markus, 2000; and Jain and Srinivasulu, 2004). There appears to be a need to explore alternative training methods to overcome such limitations.

The objectives of this chapter are to present a case study of rainfall-runoff modelling that aims at (a) integrating conceptual and neural techniques, (b) decomposing data corresponding to a flood hydrograph using physical concepts so as to permit the development of a separate model for each class, (c) employing a real-coded genetic algorithm (RGA) for ANN training and comparing its performance with BP trained models and (d) comparing the results of the integrated ANN models with a CRR model and a black-box type ANN model in terms of certain standard statistical performance evaluation measures. The approaches proposed in this chapter are tested using the observed data derived from the Kentucky River Basin. The chapter begins with a brief description of the study area and model performance statistics employed to evaluate various models. The development of various types of models is presented next before discussing the results and making concluding remarks.

5.3 Study Area and Model Performance

5.3.1 Study Area and Data

The data derived from the Kentucky River Basin were employed to train and test all the models developed in this study. The drainage area of the Kentucky River at Lock and Dam 10 (LD10) is approximately 10,240 km^2 and the time of concentration of the catchment is approximately 2 days. The data used in this study comprise the average daily streamflow (m^3/s) from the Kentucky River at LD10, and daily average rainfall (mm) from five rain gauges (Manchester, Hyden, Jackson, Heidelberg, and Lexington Airport) scattered throughout the Kentucky River catchment. A total length of 26 years of data (1960–1989 with data in some years missing) was available. The data were divided into two sets: a training data set consisting of the daily rainfall and flow data for 13 years (1960–1972) and a testing data set consisting of the daily rainfall and flow data of 13 years (1977–1989).

5.3.2 Model Performance

The performance of the models developed in this study was evaluated using six different standard statistical measures: threshold statistic (TS), average absolute relative error (AARE), the Pearson coefficient of correlation (R), Nash-Sutcliffe efficiency (E), normalized mean bias error (NMBE), and normalized root mean squared error (NRMSE). The TS and AARE statistics have been used by the authors extensively in the past (Jain et al., 2001; Jain and Ormsbee, 2002). The NMBE statistic indicates an over-estimation or under-estimation in the estimated values of the physical variable being modelled, and provides information on long-term performance. The NRMSE statistic provides information on short-term performance by allowing a term-by-term comparison of the actual differences between the estimated and observed values. The equations to compute TS, AARE, NMBE, and NRMSE are provided here:

$$\mathrm{TS}_x = \frac{n_x}{N} \times 100\% \tag{5.1}$$

$$\mathrm{AARE} = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{QO(t) - QP(t)}{QO(t)}\right| \times 100\% \tag{5.2}$$

$$\mathrm{NMBE} = \frac{\frac{1}{N}\sum_{t=1}^{N}(QP(t) - QO(t))}{\frac{1}{N}\sum_{t=1}^{N} QO(t)} \times 100\% \tag{5.3}$$

$$\text{NRMSE} = \frac{\left(\frac{1}{N}\sum_{t=1}^{N}(QP(t)-QO(t))^2\right)^{1/2}}{\frac{1}{N}\sum_{t=1}^{N}QO(t)} \tag{5.4}$$

where n_x is the total number of flow data points predicted in which the absolute relative error (ARE) is less than $x\%$, N is the total number of flow data points predicted, $QO(t)$ is the observed flow at time t, and $QP(t)$ is the predicted flow at time t. The threshold statistic for the ARE levels of 1, 5, 25, 50, and 100% were computed in this study. The TS and AARE statistics measure the 'effectiveness' of a model in terms of its ability to accurately predict flow from a calibrated model. The other statistics, R, E, NMBE, and NRMSE, quantify the 'efficiency' of a model in capturing the extremely complex, dynamic, nonlinear, and fragmented rainfall-runoff relationships. A fuller discussion on the merits of these different measures can be found in Dawson et al. (2007).

5.4 Model Development

Five different models of the rainfall-runoff process are presented in this chapter. The first two models are the standalone models that employ the ANN and conceptual techniques in isolation. The first model (called the ANN model in this chapter) uses a multilayer perceptron (MLP) neural network trained using the BP algorithm on the raw rainfall and runoff data. The second model is a conceptual model in which infiltration was modelled using the Green-Ampt equations (Chow et al., 1988), the soil moisture was modelled using the law of conservation of mass, evapotranspiration was modelled using the Haan (1972) method, and the complex rainfall-runoff process in the catchment was modelled by assuming it to be a linear reservoir. The third model uses a self-organizing map (SOM) to first divide the rainfall-runoff data into different classes and then develop different MLP neural network models. The fourth model (called the Integrated-BP model) decomposes the rainfall and runoff data corresponding to a flow hydrograph into four segments and then develops different models for different segments. The fifth model (called the Integrated-GA model) is the same as the fourth model except that the ANN model components were trained using a RGA.

The ANN model consisted of three layers: an input layer, a hidden layer, and an output layer. The inputs to the ANN model were decided based on the cross-correlation analyses of the rainfall and runoff data. Five significant inputs were identified: total rainfall at times t, $t-1$, and $t-2$ $\{P(t), P(t-1)$, and $P(t-2)\}$ and the observed flow values at times t–1 and $t-2$ $\{Q(t-1)$ and $Q(t-2)\}$. The only neuron in the output layer represented the flow at time t $\{Q(t)\}$ being modelled. The number of hidden neurons was determined based on trial and error. Four

hidden neurons were found suitable based on error minimization criteria. The standard BP algorithm was used for training. The stopping criteria were decided based on experience so as to prevent over-training or under-training. Once the training step was over, the trained ANN model was used to calculate the runoff response using training and testing data sets. The structure of the conceptual model is outlined in Jain (Chap. 21). The remaining models employed effective rainfall calculated using the Green-Ampt method as inputs instead of the total rainfall. The third model presented here employed a SOM to first classify the effective rainfall-runoff data into four separate categories. Once the input output space was divided into four classes using a SOM, the data from each class were used to develop independent ANN models using supervised training with BP. The input architecture of the individual ANN model was decided using a cross-correlation analysis between input and output data sets of the corresponding class of observations from the training data set. The following inputs were found to be important for the four classes: $P(t)$, $P(t-1)$, $P(t-2)$, $Q(t-1)$, and $Q(t-2)$ for class 1; $P(t)$, $P(t-1)$, $Q(t-1)$, and $Q(t-2)$ for class 2; $P(t)$, $Q(t-1)$, and $Q(t-2)$ for class 3; and $P(t)$, $Q(t-1)$, and $Q(t-2)$ for class 4. Thus, the ANN architectures of 5-N1-1, 4-N2-1, 3-N3-1, and 3-N4-1 were explored for the four classes, respectively. The optimum number of hidden neurons for the four classes was determined to be 4, 3, 3, and 3 through trial and error. Once the model structures for the individual classes were identified, they were used to calculate runoff for both the training and testing data sets. The development of the integrated models is described next.

5.4.1 Integrated Models

The integrated models were developed by dividing the effective rainfall and runoff data associated with a flood hydrograph into the different segments corresponding to the different dynamics. This approach of decomposing a flood hydrograph is based on the concept that the different segments of a flood hydrograph are produced by the different physical processes in a catchment. Since the rising and falling limbs in a flood hydrograph are produced by the different physical processes in a catchment, the first step in decomposing a flood hydrograph can be to separate the data into two categories corresponding to the rising and falling limbs, respectively. This can be achieved by breaking the flood hydrographs using the peak flow value for each flood hydrograph where the slope of the flow hydrograph changes sign. The effective rainfall and runoff data before the peak from all the flood hydrographs in a data set can be combined together for the rising limb, and the data after the peak of all the flood hydrographs can be combined together for the falling limb. The next step is to sub-divide the data on the rising limb (or the falling limb) corresponding to the different dynamics into different classes. The question(s) that need to be answered are as follows: Into how many segments should the rising limb (or the falling limb) be divided and how? Answering such question(s) is not simple as it would depend on the hydrological and meteorological characteristics associated

with a catchment. This task can be made simple by choosing a flow value on the rising limb (or the falling limb) and calculating certain error statistics in predicting the flood hydrographs. In the present study, the rising and falling limbs were divided into two segments each using a trial and error method to minimize three error statistics, namely the t-statistic, NMBE, and NRMSE in predicting the flow hydrographs. The flow values of 11.33 m^3/s (400 ft^3/s) and 70.8 m^3/s (or 2500 ft^3/s) were found suitable for dividing the rising and falling limbs, respectively. Using these flow values, the effective rainfall and runoff data were separated into four different classes. Once the effective rainfall and runoff data associated with the flood hydrographs have been divided into different segments, the next step is to develop models to capture the fragmented functional relationships inherent in each data set using a different technique. The ANN technique was found suitable to model the segments of the flow hydrograph close to the peak, and a conceptual technique was found suitable to model the initial and final segments of a flow hydrograph away from the peak. The concept of flow recession was used to model the segments of a flood hydrograph away from the peak. The computation of the flow at time t, Q_t, using the concept of flow recession can be represented by the following equations:

$$Q_t = K_t \, Q_{t-1} \tag{5.5}$$

$$K_t = \frac{Q_{t-1}}{Q_{t-2}} \tag{5.6}$$

where Q_t is the flow at time t, and K_t is the recession coefficient for the flow at time t that can be determined adaptively at each time step. The value of K_t at the beginning of a flood hydrograph can be computed using the observed flood data of the previous flow hydrograph. It is to be noted that the concept of flow recession can be used to model not only the falling limb of a flood hydrograph but also the segments of the rising limb of a flood hydrograph. This is possible as the shape of a flow hydrograph at the beginning of the rising limb can be assumed to be close to the mirror image of the flood hydrograph at the final segment of the falling limb. The rising segment just before the peak and the falling segment just after the peak were modelled by ANNs trained using the BP algorithm. The input architecture of the ANN models was decided using a cross-correlation analysis between input and output data sets of the corresponding class of observations from the training data set. The following inputs were found to be important based on these analyses: $P(t)$, $P(t-1)$, $P(t-2)$, $Q(t-1)$, and $Q(t-2)$ for modelling the rising segment of the flow hydrograph just before the peak; and P(t), $Q(t-1)$, and $Q(t-2)$ for modelling the falling segment of the flow hydrograph just after the peak. The optimum numbers of hidden neurons were determined to be 4 and 3 for the rising and falling segments, respectively. The values of 0.01 and 0.075 for the learning coefficient and the momentum correction factor, respectively, were used to train all the ANN models/components in this study. The Integrated-BP model used the BP algorithm and the Integrated-GA model used a RGA to train the ANN components on the rising and falling limbs. Once the model

structures for the individual classes were finalized and trained, they were used to calculate runoff of both the training and testing data sets.

5.5 Results and Discussions

The results in terms of various statistics are presented in Table 5.1. A number in bold represents the best performance in terms of a particular statistic (within that column) either during training or testing. The results indicate that the integrated models perform better than the individual models. The performance of the ANN model was good in terms of R, E, and NRMSE but was very poor in terms of TS and AARE. On the other hand, the performance of the conceptual model was good in terms of TS and AARE but was not so good in terms of R, E, and NRMSE.

Thus, there appears to be a trade-off when selecting a model developed on a single technique. The SOM model obtained the best R and E values of 0.9804 and 0.9612, respectively, during training but did not perform as well as the Integrated-GA model in terms of TS and AARE. The Integrated-GA model obtained the best TS values for all ARE levels during both training and testing. It also obtained the best AAREs of 16.47% during training and 17.96% during testing. Although the ANN model obtained the best R, E, and NRMSE during testing, it performed the worst in terms of TS and AARE. In the experience of the author, ANN models in hydrology tend to perform very well in terms of global error statistics (R, E, and RMSE, etc.) but are very poor in estimating runoff values accurately in terms of TS and AARE. The global error statistics are biased towards high-magnitude runoff due to the square of the deviations or its variations in the expressions while the TS and AARE are unbiased statistics that are relative in nature. A robust model will capture the complex, dynamic, nonlinear, and fragmented physical rainfall-runoff process not only in terms of global statistics but also in terms of unbiased statistics

Table 5.1 Performance evaluation statistics during training

Model	TS1	TS5	TS25	TS50	TS100	AARE	R	E	NMBE	NRMSE
During training										
ANN	2.97	13.59	53.47	70.42	83.15	54.45	0.9770	0.9544	1.52	0.347
Conceptual	6.10	8.87	68.52	89.26	97.81	23.57	0.9363	0.8436	8.95	0.639
SOM	4.27	22.06	72.27	90.89	98.44	20.80	**0.9804**	**0.9612**	0.12	**0.320**
Integrated-BP	6.25	26.70	74.71	88.55	95.40	23.85	0.9780	0.9570	**−0.06**	0.339
Integrated-GA	**6.31**	**28.71**	**79.73**	**94.22**	**99.24**	**16.47**	0.9785	0.9575	−0.76	0.335
During testing										
ANN	2.68	12.96	54.43	70.40	80.57	66.78	**0.9700**	**0.9410**	0.75	**0.388**
Conceptual	4.84	18.32	67.89	88.66	97.59	24.68	0.9332	0.8344	9.46	0.649
SOM	2.59	13.73	61.75	86.17	97.61	26.51	0.9620	0.9100	9.63	0.478
Integrated-BP	7.01	28.07	72.34	89.62	97.62	21.63	0.9678	0.9366	0.65	0.402
Integrated-GA	**7.12**	**28.22**	**76.14**	**93.13**	**99.28**	**17.96**	0.9686	0.9389	**0.26**	0.397

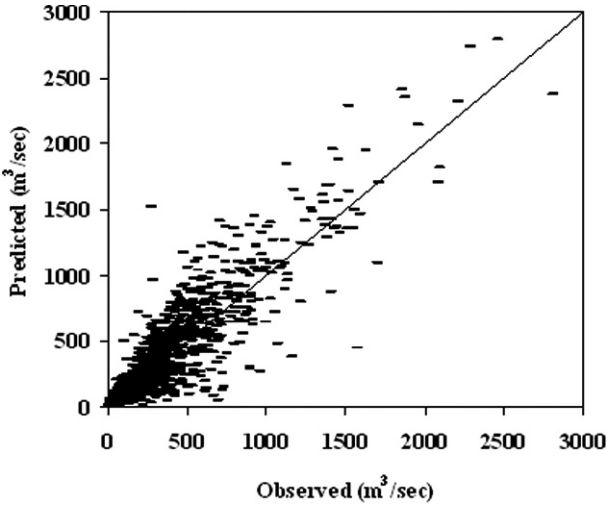


Fig. 5.1 Scatter plot from simple conceptual model

such as TS and AARE, which is required by water resource planning, design, and management activities. The results obtained in this study demonstrate that the integrated models that exploit the advantages of both conceptual and ANN techniques are much better than their individual counterparts.

The graphical results are shown in Figs. 5.1–5.4. Figures 5.1 and 5.2 show the scatter plots during the 13-year testing period obtained from the conceptual and Integrated-GA models, respectively. It can be seen that the wide spread around the

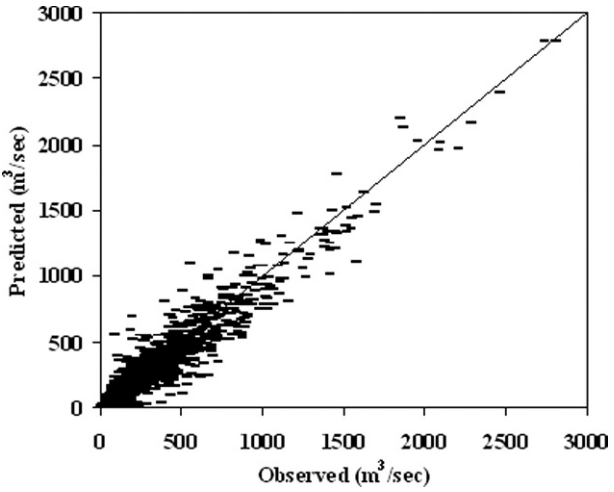


Fig. 5.2 Scatter plot from Integrated-GA model

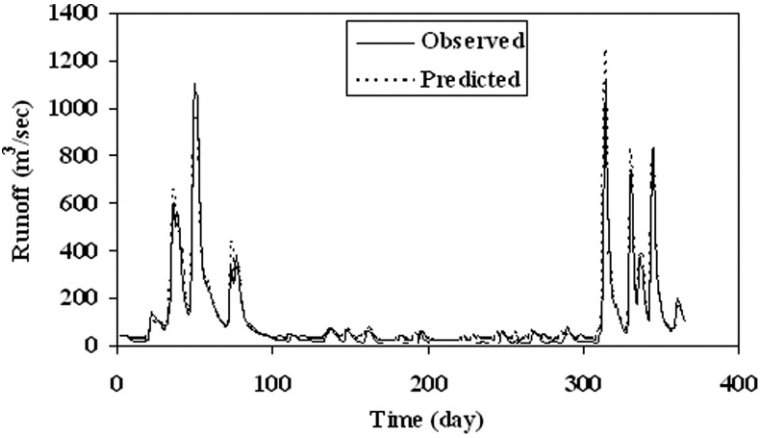


Fig. 5.3 Observed and estimated flows for 1986 from simple ANN model

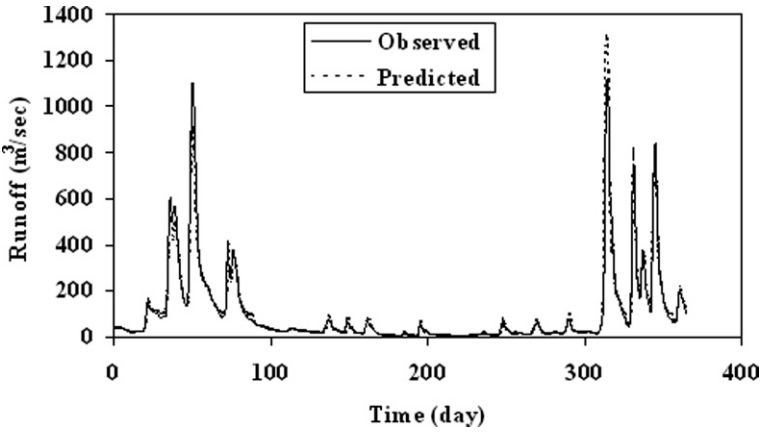


Fig. 5.4 Observed and estimated flows for 1986 from Integrated-GA model

ideal line from the conceptual model has been significantly reduced in the integrated model trained using a RGA. Figures 5.3 and 5.4 show the time series plots of the observed and estimated runoff during a sample year (1986) taken from the testing data set for the ANN and Integrated-GA models, respectively. It is very encouraging to note that the problem of over-estimating the low-magnitude flows from the ANN model during the summer months (see days 100–300 in Fig. 5.3) is significantly overcome by the Integrated-GA model (see Fig. 5.4). These results demonstrate that the integration of data and modelling techniques can significantly improve the overall performance in rainfall-runoff modelling.

5.6 Summary and Conclusions

This chapter presents a new approach for integrating available data and modern modelling techniques for developing better models of the complex rainfall-runoff process. Two different approaches based on data decomposition are investigated. The first approach uses the self-organizing capability of SOM models for data classification and then develops different MLP neural network models for four different classes. The second approach decomposes the rainfall and runoff data corresponding to a flood hydrograph into four classes based on physical concepts before developing a combination of conceptual and ANN models for different segments of the flow hydrograph. The results from an ANN and a conceptual model are also presented for comparison purposes. Three-layer feedforward ANNs with BP were employed to develop all ANN models. A real-coded GA was also employed in the integrated model to assess its capability in developing a better generalization of the rainfall-runoff process. Rainfall and flow data derived from the Kentucky River Basin were used to test the proposed methodologies.

The results obtained in this study indicate that the conceptual and ANN models were good based on certain types of statistics yet poor when considering others. The integrated models developed on (both SOM and physically based) decomposed data performed better than the individual models developed on a single technique. The integrated model that decomposed the data corresponding to a flood hydrograph into four classes, modelled segments away from peak using the concept of flow recession and modelled segments closer to peak using ANNs trained by RGA was found to be the best for rainfall-runoff modelling for the data considered in this study. The rainfall-runoff process in the Kentucky River Basin was found to have four different classes corresponding to different dynamics in the underlying physical processes; however, different catchments with different climatic and hydrologic conditions may have different numbers of classes corresponding to different dynamics involved since each catchment is unique in its response to rainfall. This study is able to demonstrate that it is possible to achieve improved performance in rainfall-runoff modelling by specialized efforts focused in the direction of integrating available data and modern modelling techniques. More research is needed in this direction to verify the reported findings here in order to improve the accuracy of runoff forecasts required by current water resource applications.

References

Abrahart RJ, See LM (2000) Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecasts in two contrasting catchments. Hydrol Proc 14:2157–2172

Anctil F, Tape DG (2004) An exploration of artificial neural network rainfall-runoff forecasting combined with wavelet decomposition. J Env Engg Sci 3(1):S121–S128

Chow VT, Maidment DR, Mays LW (1988) Applied Hydrology. McGraw Hill, New York, USA

Curry B, Morgan P (1997) Neural network: A need for caution. Omega-Intl J Mgmt Sci 25(1):123–133

Dawson CW, Abrahart RJ, See L (2007) HydroTest: a web-based toolbox of statistical measures for the standardised assessment of hydrological forecasts. Environmental Modelling and Software 27:1034–1052.

Haan CT (1972) A water yield model for small watersheds. Wat Resour Res 8(1):58–69

Hsu K-L, Gupta HV, Sorooshian S (1995) Artificial neural network modeling of the rainfall-runoff Process. Wat Resour Res 31(10):2517–2530

Hsu K-L, Gupta HV, Gao X, Sorooshian S, Imam B (2002) Self-organizing linear output map (SOLO): An artificial neural network suitable for hydrologic modeling and analysis. Wat Resour Res 38(12):1302

Jain A, Indurthy SKVP (2003) Comparative analysis of event based rainfall-runoff modeling techniques-deterministic, statistical, and artificial neural networks. J Hydrol Engg ASCE 8(2):93–98

Jain A, Ormsbee LE (2002) Evaluation of short-term water demand forecast modeling techniques: Conventional methods versus AI. J Am Wat Works Assoc 94(7):64–72

Jain A, Srinivasulu S (2004) Development of effective and efficient rainfall-runoff models using integration of deterministic, real-coded GA, and ANN techniques. Wat Resour Res 40(4):W04302 doi:10.1029/2003WR002355

Jain A, Varshney AK, Joshi UC (2001) Short-term water demand forecast modeling at IIT Kanpur using artificial neural networks. Wat Resour Mgmt 15(5):299–321

Karunanithi N, Grenney WJ, Whitley D, Bovee K (1994) Neural networks for river flow predictions. J Comput Civ Engg ASCE 8(2):201–220

Minns AW, Hall MJ (1996) Artificial neural networks as rainfall runoff models. Hydroll Sci J 41(3):399–417

Ooyen AV, Nichhuis B (1992) Improving convergence of back propagation problem. Neural Net 5:465–471

Rajurkar MP, Kothyari UC, Chaube UC (2004) Modeling of the daily rainfall-runoff relationship with artificial neural network. J Hydrol 285(1–4):96–113

Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323:533–536

Porporato A, Ridolfi L (2001) Multivariate nonlinear prediction of river flows. J Hydrol 248:109–122

Smith J, Eli RN (1995) Neural network models of the rainfall runoff process. J Wat Resour Plng Mgmt ASCE 121:499–508

Subramanya K (1997) Engineering Hydrology. Tata McGraw-Hill, New Delhi, India

Tokar AS, Johnson PA (1999) Rainfall-runoff modeling using artificial neural networks. J Hydrol Engg ASCE 4(3):232–239

Tokar AS, Markus M (2000) Precipitation runoff modeling using artificial neural network and conceptual models. J Hydrol Engg ASCE 5(2):156–161

Zhang B, Govindaraju S (2000) Prediction of watershed runoff using bayesian concepts and modular neural networks. Wat Resour Res 36(3):753–762

Chapter 6
Dynamic Neural Networks for Nonstationary Hydrological Time Series Modeling

P. Coulibaly and C.K. Baldwin

Abstract Evidence of nonstationary trends in hydrological time series, which result from natural and/or anthropogenic climatic variability and change, has raised a number of questions as to the adequacy of conventional statistical methods for long-term (seasonal to annual) hydrologic time series forecasting. Most conventional statistical methods that are used in hydrology will suffer from severe limitations as they assume a stationary time series. Advances in the application of artificial neural networks in hydrology provide new alternative methods for complex, nonstationary hydrological time series modeling and forecasting. An ensemble approach of competitive recurrent neural networks (RNN) is proposed for complex time-varying hydrologic system modeling. The proposed method automatically selects the most optimally trained RNN in each case. The model performance is evaluated on three well-known nonstationary hydrological time series, namely the historical storage volumes of the Great Salt Lake in Utah, the Saint-Lawrence River flows at Cornwall, and the Nile River flows at Aswan. In each modeling experiment, forecasting is performed up to 12 months ahead. The forecast performance of the proposed competitive RNN model is compared with the results obtained from optimal multivariate adaptive regression spline (MARS) models. It is shown that the competitive RNN model can be a good alternative for modeling the complex dynamics of a hydrological system, performing better than the MARS model, on the three selected hydrological time series data sets.

Keywords Nonstationarity · hydrologic time series · modeling and forecasting · recurrent neural networks · multivariate adaptive regression splines.

P. Coulibaly
Department of Civil Engineering/School of Geography and Earth Sciences, McMaster University, Hamilton, Ontario, L8S 4L7 Canada, e-mail: couliba@mcmaster.ca

C.K. Baldwin
Utah Water Research Laboratory, Utah State University, Logan, Utah 84322-8200, USA

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

6.1 Introduction

The forecasting of environmental time series variables requires modeling of the underlying physical mechanism responsible for their generation. In practice, many real-world dynamical systems exhibit two distinct characteristics: nonlinearity and nonstationarity in the sense that statistical characteristics change over time due to either internal or external nonlinear dynamics. Explicit examples of such series are the historical storage volumes of the Great Salt Lake (GSL), the Saint-Lawrence River (SLR) flow records, and the Nile River flow records. Large or regional hydrosystems are inherently dynamic and subject to climatic variability and change. The evidence of nonstationarity of some existing long hydrological records has raised a number of questions as to the adequacy of the conventional statistical methods for long-term regional water resources forecasting. The fundamental assumption in most empirical or statistical approaches (Box and Jenkins, 1976) is stationarity over time. Therefore, nonstationary time series have to be reduced to being stationary, for example through differencing. This procedure has the disadvantage of amplifying high-frequency noise in the data (Young, 1999). The simplest alternatives to linear regression methods are dynamic regression models (Young, 1999). Although these methods may indeed be useful for the analysis of nonstationary time series, their inherently static regression feature cannot fully explain many complex geophysical data sets. The identification of a nonlinear dynamical model that relates directly to the underlying dynamics of the system being modeled remains a challenging and active research topic. Owing to the difficulty of identifying an appropriate nonlinear model structure, very few nonlinear empirical models have been proposed in the literature for complex hydrological time series modeling (Coulibaly et al., 2000a). In watershed and aquifer system modeling, recent attempts have resorted to nonlinear data-driven methods, specifically artificial neural networks (ANNs). These tools are found to be more suited to nonlinear input–output mapping, and recent reviews reveal that more than 90% of ANN applications in water resources modeling used standard feedforward neural networks (FNNs) (Coulibaly et al., 1999; Maier and Dandy, 2000; ASCE Task Committee, 2000). Even though an optimal FNN model can provide accurate forecasts for simple rainfall-runoff problems, it often yields sub-optimal solutions even with lagged inputs or tapped delay lines (Coulibaly et al., 2001). Standard FNNs are not well suited for complex temporal sequence processing owing to their static memory structure (Giles et al., 1997; Haykin, 1999). FNNs are similar to nonlinear static regression models in that they are unable to adapt properly to the temporal relationship of real-world data. A promising class of ANNs for nonstationary time series modeling is dynamically driven recurrent neural networks (RNNs) (Haykin and Li, 1995; Haykin, 1999). In theory, a RNN can learn the underlying dynamics of a nonstationary environment if the training sample is representative of the time-varying behavior of the system that is being modeled (Haykin, 1999). However, designing such a network for nonlinear forecasting of physical nonstationary time series is a difficult task because (1) traditional gradient-descent learning methods are unstable, and have to be properly adapted; and (2) the network needs to learn online and adapt to statistical variations

of the time series while simultaneously performing its filtering role. Recent attempts in using RNNs for nonstationary time series forecasting have demonstrated the potential of the method for modeling complex time-varying patterns (Saad et al., 1998; Iatrou et al., 1999; Coulibaly et al., 2000a). However, none of these applications has considered nonstationary hydrological time series.

The purpose of this chapter is to introduce a dynamic RNN approach for the forecasting of nonstationary hydrological time series. The proposed method uses an ensemble of parallel and competitive sub-networks (or "subnets") to directly identify the optimal dynamic model for the system being considered. The performance of the method is assessed against that of an optimal multivariate adaptive regression spline (MARS) model (Friedman, 1991; Lall et al., 1996) in experiments based on complex hydrological time series modeling for three physically different hydroclimatic regions.

The reminder of the chapter is organized as follows. Section 6.2 provides a brief description of the experimental data, related studies, and a wavelet analyses of the data sets. In Sect. 6.3, we describe the dynamic RNN and the ensemble competitive approach. In Sect. 6.4, results from the modeling experiment are reported and discussed. Finally, in Sect. 6.5, our conclusions are presented.

6.2 Experimental Data Sets and Wavelet Analyses

Three hydrological time series that have been analyzed extensively by numerous authors (Eltahir, 1996; Bernier, 1994; Sangoyomi, 1993) and whose statistical characteristics have been shown to change over time (i.e., corresponding to the term "nonstationary process" used here) are considered. These are the monthly flow records of the Nile River at Aswan, from 1871 to 1989, the monthly flows of the Saint-Lawrence River (SLR) at Cornwall, from 1861 to 1999, and the bi-weekly storage volumes of the Great Salt Lake (GSL), from 1847 to 1999. The selected hydrological series were also proven to exhibit significant linkages with low-frequency (interannual or decadal) climatic variability indices. It has been reported that the underlying mean of the Nile River flow is both time and El-Niño/Southern Oscillation (ENSO) dependent, resulting in a nonstationary process (Eltahir, 1996). The changes in the mean of the SLR flows have been analyzed by Bernier (1994), who showed that the 1861–1991 annual flows of the SLR can be divided into three periods (1861–1891, 1892–1968, 1969–1991) with significantly different means. The nonlinear time-varying dynamics of the GSL have been documented by Sangoyomi (1993) and Lall et al. (1996). Furthermore, the connection between the climatic signals and the fluctuations in the volume of the GSL has been proven and is well documented by Lall and Mann (1995) and Mann et al. (1995). Figure 6.1 shows the deviations from the historical mean for the three hydrological series under study. It appears that globally the selected time series exhibit long-term (greater than 10 years) as well as low-frequency (interannual to decadal) downward and/or upward trends in addition to their inherent seasonal fluctuations. Prior to 1900, there are

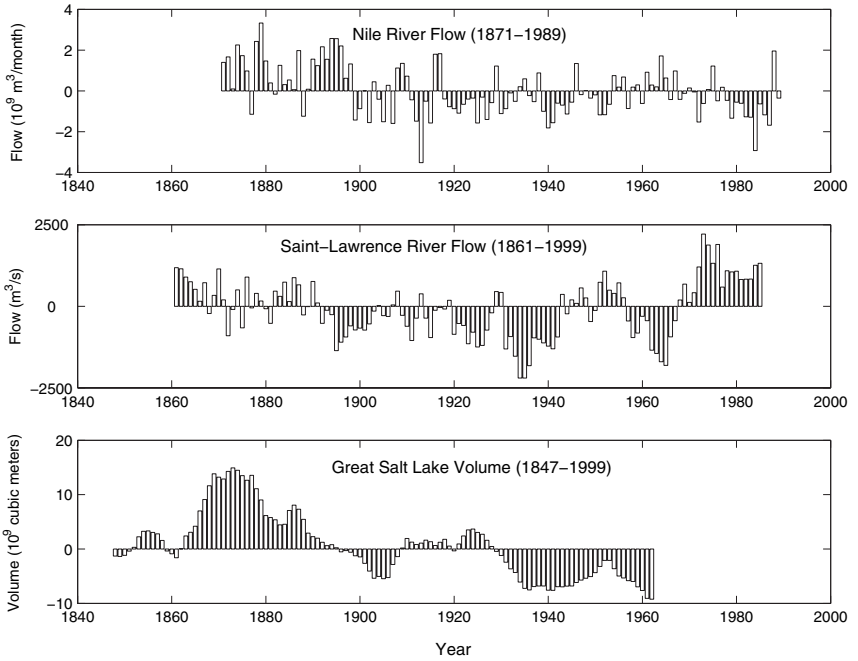


Fig. 6.1 Deviations of annual hydrological records from their historical means

essentially positive deviations from the historical mean of the series, whereas after 1900 the negative deviations are relatively dominant up to 1999, except for SLR flows in the period 1969–1999 and the GSL in the period 1985–1990. More specifically, there are apparent periodical patterns in each series from 1900 onwards. These visual structures are supported by more rigorous statistical analysis (Lall and Mann, 1995; Eltahir, 1996; Bernier, 1994). Nevertheless, to further assess the complexity of the selected time series, a Morlet wavelet analysis of the selected hydrological time series is carried out.

Wavelet analysis. The theoretical basis for using the wavelet transform to analyze a time series that contains nonstationary power at different frequencies has been documented by Daubechies (1990). The Morlet wavelet is a complex nonorthogonal wavelet function which has been shown to be particularly well adapted for analyzing nonstationary variance at different frequencies within hydrologic and geophysical time series (Foufoula-Georgiou and Kumar, 1995; Torrence and Campo, 1998; Coulibaly et al., 2000a; Labat et al., 2000). In this analysis, the wavelet analysis toolkit kindly provided by Torrence and Campo (1998) (at URL: http://paos.colorodo.edu/research/wavelets/) is used.

With the wavelet analysis, localized time-scale variations can be seen within the GSL volume series (Fig. 6.2). It should be noted that only the structures (or components) corresponding to the highest wavelet coefficient are shown. The contour lines in Fig. 6.2b enclose structures of greater than 95% confidence for a red-noise

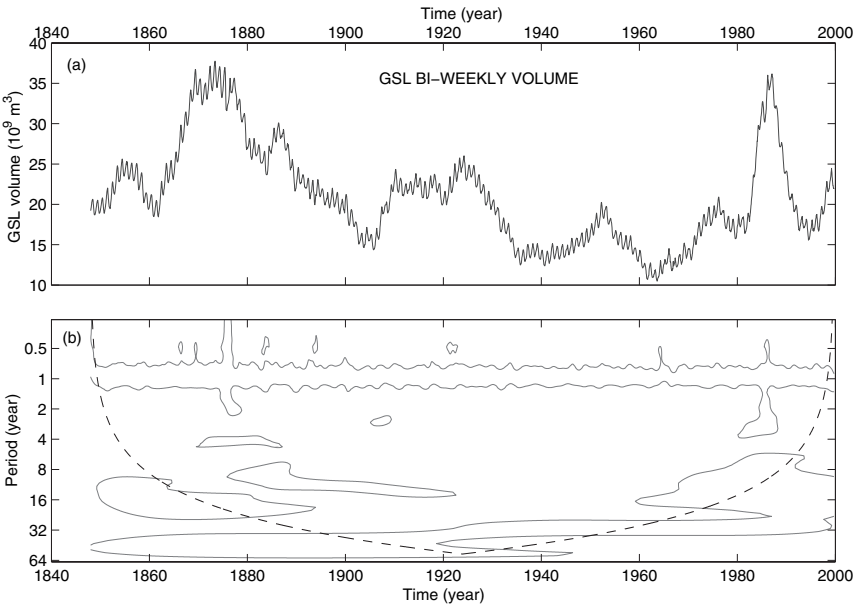


Fig. 6.2 (**a**) The GSL bi-weekly volume series used for the wavelet analysis. (**b**) The local wavelet power spectrum of the GSL bi-weekly volume series using the Morlet wavelet. The contours enclose structures of greater than 95% confidence for a red-noise process with a lag-1 coefficient of 0.98. The *dashed curve* depicts the cone of influence beyond which the edge effects become important

process with a lag-1 coefficient of 0.98. In Fig. 6.2b, the Morlet wavelet power spectrum indicates distinct periodic processes (or components) characterized by a variable period (ranging from 0.5 to 32 years). In addition to the persistent annual component, there are different multi-annual components. From 1870 to 1920, there is a clear shift from a period of around 4 years to a period of the order of 8–14 years (specifically from 1875 to 1920), while from 1920 to 1999, the shift is from longer to shorter periods. It also appears that larger GSL volumes ($> 30 \times 10^9\,\mathrm{m}^3$) can be associated with the 2–4 year processes observed during 1870–1890 and 1985–1990 which in turn can be related to the strong El-Niño (ENSO) events observed at these time periods (Torrence and Campo, 1998).

Similarly, the Morlet wavelet spectra for the SLR and the Nile River flows are depicted in Fig. 6.3. The contour lines in Fig. 6.3a,b enclose structures of greater than 95% confidence for a red-noise process with a lag-1 coefficient of 0.90 and 0.61 for the Nile and the SLR monthly flows, respectively. The wavelet power spectra reveal more localized (in time) processes specifically before the year 1920, while from 1920 onwards, only the power spectrum of the SLR flows exhibits a longer period ($\sim$24 year) component. The annual structure is visible in both spectra, but with striking ruptures for the SLR flow (Fig. 6.3b). The annual component clearly disappears in 1965–1970, while intraannual structures appear in the

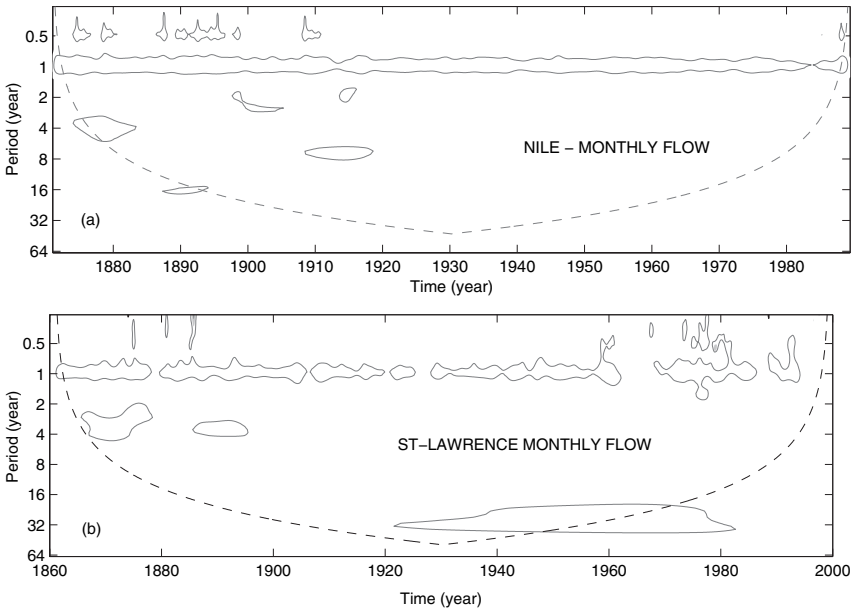


Fig. 6.3 (**a**) The local wavelet power spectrum of the Nile River monthly flow using the Morlet wavelet. The contours enclose structures of greater than 95% confidence for a red-noise process with a lag-1 coefficient of 0.90. (**b**) The local wavelet power spectrum of the SLR monthly flow using the Morlet wavelet. The contours enclose structures of greater than 95% confidence for a red-noise process with a lag-1 coefficient of 0.61. Other features are identical to Fig. 6.2b

last two data decades (1975–1995). Note that the year 1970 has been recently identified as a possible change point in the Canadian Southeastern streamflow which highlights the nonstationarity of the streamflow time series in that region (Anctil and Coulibaly, 2004). Different studies have shown that the multi-annual (or low-frequency) components revealed by the wavelet analysis of streamflows can be related to climatic fluctuations (Coulibaly et al., 2000a; Anctil and Coulibaly, 2004) by performing cross-wavelet (or covariance) analysis. However, this is beyond the objective of our analysis. Here the wavelet analysis is used to highlight the time-scale variability of the hydrological time series, and the results show that the selected hydrological time series are nonstationary since their variance changes in frequency and intensity through time.

6.3 Ensemble of Competitive Dynamic Recurrent Neural Networks

The main objective of this part of the study is to identify an optimal dynamically driven recurrent neural network (RNN) that can capture the complex time-varying structure of nonstationary hydrological series without any data pre-processing or

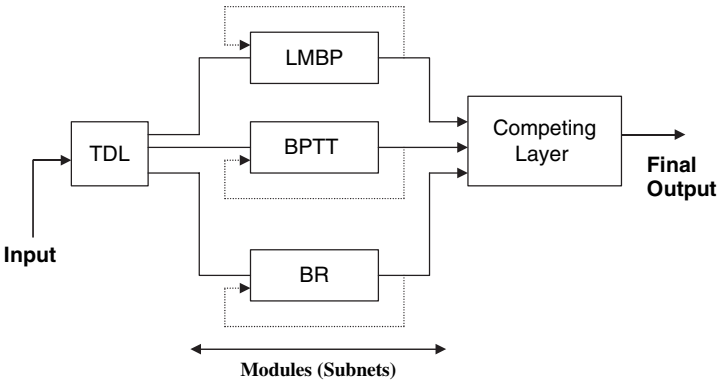


Fig. 6.4 Schematic of the ensemble of competitive recurrent neural networks. TDL, time delay line; LMBP, Levenberg Marquardt Backpropagation; BPTT, Backpropagation through time; BR, Bayesian regularization

transformation. The proposed model is essentially an ensemble of time delay RNNs (Fig. 6.4) (Coulibaly and Baldwin, 2005). The network consists of three layers: an input layer with a time delay line (TDL), a hidden layer with three independent modules (or "subnets") of recurrent (or state) nodes, and a competitive output layer. Each sub-network (or "subnet") contains three state nodes (selected by trial and error) that feed back into themselves through a set of synchronous delay lines or context units (Fig. 6.4). This recurrence facility confers on the subnet dynamic properties which make it possible for each module to have internal memory. The recurrent dynamic of the state nodes allows the information to be recycled over multiple time steps, and thereby to discover temporal information contained in the sequential input that is relevant to the target function. Thus, each subnet (or module) can be viewed as a single time delay RNN with an inherent dynamic memory (related to the context units) and a static memory structure (related to the input delay line). It is basically an Elman-type RNN (Elman, 1990) with an input delay line.

Given an input variable $x(t)$, the delay line operating on $x(t)$ yields its past values $x(t-1)$, $x(t-2)$, …, $x(t-p)$ where $p = 3$ is the optimal delay memory order selected by trial and error in this case. Therefore, the input signal $s_i^{(t)}$ to a node i of a subnet is given by the convolution sum

$$s_i^{(t)} = \sum_{k=0}^{p} w_i^{(k)} x^{(t-k)} + b_i = \mathbf{w}_i \mathbf{x}^{(t)} \tag{6.1}$$

where $\mathbf{w}_i$ is the weight vector for node i and $\mathbf{x}^{(t)}$ denotes the vector of delayed values from the input time delay line. It is important to note that the use of time delay in this context is different from the "sliding or moving window" method because the input signal to the activation function consists of the convolution summation of the sequences of the input variable and synaptic weights of the node plus a bias b_i. More precisely, the connection weight $\mathbf{w}_i$ is a vector rather than a single value as used in

the sliding (or moving) window approach commonly used for temporal processing with feedforward neural networks.

The recurrence equation that defines the temporal dynamics of the RNN typically filters the weighted sum of the inputs and states through a nonlinear mapping function. This equation has the general form:

$$\mathbf{S}^{(t+\mathbf{1})} = \Phi\left(\mathbf{I}^{(t)}, \mathbf{S}^{(t)}, \mathbf{W}, \Theta\right) \tag{6.2}$$

where $\mathbf{S}^{(t)}$ is the state vector representing the values of all the state nodes at time t, $\mathbf{I}^{(t)}$ is the input vector at time t, $\Phi(.)$ denotes a logistic function characterizing the hidden nodes, $\mathbf{W}$ is a set of weight matrices defining the weighted connections between the subnet layers, and Θ is a set of biases on the state nodes. Note that (6.2) is typically a standard definition of the $\{state - input - next\text{-}state\}$ mapping found in the definitions of both finite state automata (Kohavi, 1978) and nonlinear dynamical system models (Narendra and Parthasarathy, 1990). Finally, the output of each sub-network depends not only on the connection weights and the current input signal but also on the previous states of the network, as follows:

$$y_j^{(t)} = \mathbf{A}\mathbf{S}^{(t)} \tag{6.3}$$

with

$$\mathbf{S}^{(t)} = \Phi\left(\mathbf{W}_h \mathbf{S}^{(t-1)} + \mathbf{W}_{h_0} \mathbf{I}^{(t-1)}\right) \tag{6.4}$$

where $y_j^{(t)}$ is the output of a subnet assuming a linear output node j, $\mathbf{A}$ is the weight matrix of the output layer node j connected to the hidden nodes, the matrix $\mathbf{W}_h$ represents the weights of the h ($h = 1, \ldots, 3$) hidden nodes that are connected to the context units, and $\mathbf{W}_{h_0}$ is the weight matrix of the hidden units connected to the input nodes. Each subnet in the network is a typical *state-space model* since (6.4) performs the state estimation and (6.3) the evaluation. Here, the RNN model is developed using the Neural Network Toolbox 3 (The Mathworks Inc., Natick, MA).

A major difficulty when using time delay RNNs is the training complexity because the computation of $\nabla E(w)$, the gradient of the error E with respect to the weights, is not trivial since the error is not defined at a fixed point but rather is a function of the network's temporal behavior. Here, to identify an optimal training method and also minimize the computational cost, each module (or subnet) of the network is trained with a different algorithm, in parallel, using the same delayed inputs. The network specifically makes use of (1) an adapted recursive form of Levenberg-Marquardt backpropagation (LMBP; Hagan and Menhaj, 1994); (2) a variant of backpropagation through time (BPTT; Williams and Peng, 1990) which is considered the best online technique for practical problems (Pearlmutter, 1995); and (3) a variation of Bayesian regularization (BR; Jones and MacKay, 1998).

Finally, the competitive output layer stores all the outputs of the subnets, computes a vector of probabilities for each subnet, and selects the output that has the maximum probability of being correct given the target pattern. The output layer is similar to the competitive output layer used in probabilistic neural networks

(Coulibaly et al., 2001). The performance of the optimal (or "winning") network is assessed using the root mean squared error (*RMSE*) and the model efficiency index (R^2) (Nash and Sutcliffe, 1970). Given that all the outputs are automatically stored, a comparative performance study can also be completed. To further assess the forecast improvement obtained with the optimal RNN model as compared to the best MARS model, a mean error decrease (MED) is used. The MED is the average decrease (%) in the RMSE of the optimal RNN model as compared to the best MARS model. The MED is also referred to herein as the model forecast improvement.

Two model calibration experiments are performed using two different methods of data division (i.e., split-sampling). First, the proposed network is trained (or calibrated) using about 95% of each of the hydrological data sets, and the validation is performed on the last five years of each series. The same calibration and validation sets are used for identification and evaluation of the MARS benchmark models that provide relative performance comparisons. The practical objective is to provide long-term (12-months-ahead) forecasts of monthly flows and bi-weekly lake volumes. The temporal distribution of flow and volume throughout the year is particularly important for water resources planning and management.

6.4 Results and Discussion

The root mean squared error (*RMSE*) and the model efficiency index (R^2) are used to evaluate the model performance. In general, a R^2 value greater than 0.9 indicates a very satisfactory model performance, while a R^2 value in the range 0.8–0.9 indicates an acceptable (or good) model, and values less than 0.8 indicate an unsatisfactory model (Shamseldin, 1997). Note that only the 12-month-ahead forecasts are of concern as specified in the previous section, and are thus analyzed in detail hereafter. Figure 6.5 shows the average R^2 index statistics for the validation period 1995–1999 for the SLR and GSL, and 1985–1989 in the case of the Nile River. From Fig. 6.5, it can be seen that the optimal methods for modeling the three complex time series are the RNN trained with either the BR or the BPTT. However, the overall best training algorithm appears to be the BR. The LMBP appears to be the least suitable for modeling the three time series. These results indicate that the performance of even powerful online RNN training methods can be case dependent. A salient advantage of the proposed ensemble competition RNN method is that it can directly identify the optimal RNN model for each hydrological time series in any case. Furthermore, the ensemble competition approach significantly reduces ($\sim$60%) the computing time as compared to an individual training of each subnet. However, it appears that for the SLR flows, even the optimal algorithm (BR) does not provide good forecasts (since $R^2 < 0.8$). This may be due to the more complex nonstationary structure of the SLR flows characterized by periodic ruptures of the annual process (Fig. 6.3b). For the GSL volume forecasting, the BPTT and BR algorithms provide slightly similar and good results ($0.8 < R^2 \leq 0.9$), while the LMBP is clearly inefficient in this case. The Nile River flows can be effectively predicted by any of the three algorithms

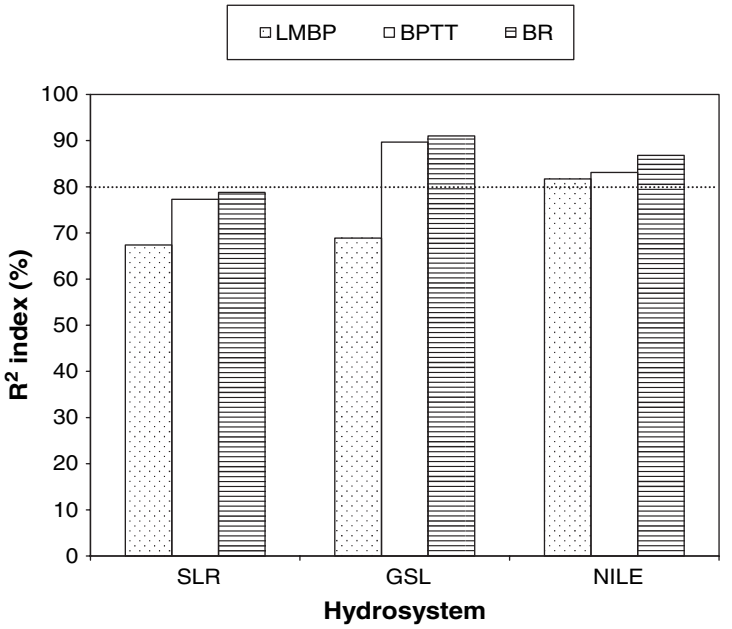


Fig. 6.5 Comparative model efficiency index for the three sub-networks

(Fig. 6.5). However, the best model still appears to be the RNN trained with BR (termed RNN-BR) whatever the system in this case. The poor performance of the LMBP may be due to the complex and inaccurate approximation of the second-order derivatives. These results may also indicate that the LMBP is less suitable for time delay RNN training than the BR and the BPTT algorithms.

Thus, the ensemble competition approach automatically selects the optimally trained network, and thereby provides the best results for each hydrological time series being modeled. To further assess the performance of the proposed method, the best RNN model identified (namely RNN-BR) is compared to an optimal MARS model using the corresponding validation *RMSE* and R^2 statistics (Table 6.1). The MARS model is developed using S-PLUS Software (Insightful Corporation, Seattle, WA) and the same calibration and validation data sets used with the RNN model. It appears from Table 6.1 that the optimal RNN model outperforms the MARS model for all three time series. For the GSL storage volumes, the optimal RNN model (RNN-BR) and the best MARS model have respectively a R^2 index value of 0.91 and 0.87, indicating that the RNN-BR model provides "very satisfactory" forecasts while the MARS model results are "fairly acceptable". However, for the Nile River flows, the RNN-BR, and the MARS are, respectively, "fairly acceptable" ($R^2 = 0.86$) and "unsatisfactory" ($R^2 = 0.64$). Conversely, for the SLR flows, none of the proposed methods are satisfactory ($R^2 < 0.8$). However, the RNN-BR forecasting *RMSE* is less than 10% of the mean monthly flow of the SLR for the validation period (Table 6.1), while the MARS model *RMSE* for the SLR flows is

Table 6.1 Model validation statistics for 12-month-ahead forecasts

Model	GSL volume Mean: $19.557 \times 10^9\,m^3$			Nile River flow Mean: $6.9765 \times 10^9\,m^3$/month			SLR flow Mean: $7528\,m^3/s$		
	RMSE ($10^9\,m^3$)	R^2	MED(%)	RMSE ($10^9\,m^3$/month)	R^2	MED(%)	RMSE (m^3/s)	R^2	MED(%)
RNN-BR	0.70	0.91	53	2.43	0.86	64	455	0.78	41
MARS	1.49	0.88		6.90	0.64		770	0.31	

RNN-BR, recurrent neural networks with Bayesian regularization; MARS, multivariate adaptive regression splines; RMSE, root mean squared error; R^2, model efficiency index; MED, mean error decrease; GSL, Great Salt Lake; SLR, Saint-Lawrence River.

about twice that of the optimal RNN. This clearly indicates a good potential of the dynamic RNN for complex hydrological time series forecasting assuming that an appropriate training method is found.

In general, the optimal RNN provides significant forecast improvement in terms of mean error decrease (MED) (Table 6.1), exceeding 40% for all three series as compared to the corresponding values of the best MARS model. For the GSL volumes and the Nile River flows, the average forecast improvement is about 53 and

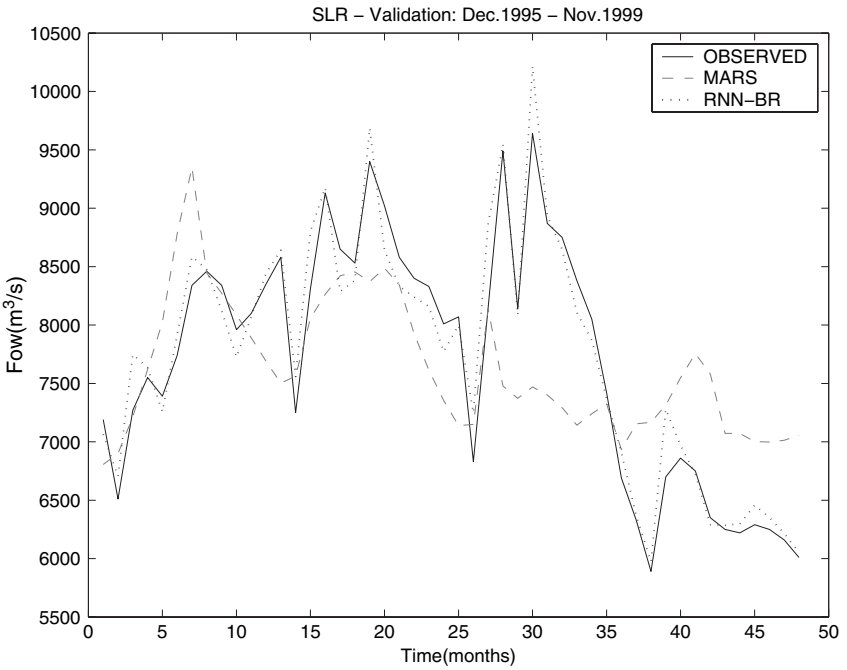


Fig. 6.6 Observed and predicted SLR monthly flows (Dec. 1995–Nov. 1999)

64%, respectively, while for the SLR flow, the average forecast improvement is about 41%. However, despite the significant improvement of the SLR flow forecasts, the forecasting results remain on the border of acceptable ($R^2 = 0.78$). This may indicate that additional explanatory variables (e.g., precipitation, climatic indices) should be included. In this case, the ensemble competition approach appears particularly useful to assess the system complexity before resorting to the provision of additional information.

To further assess the model forecasting accuracy, plots of observed and predicted flow for the SLR are shown in Fig. 6.6. Neither the RNN-BR nor the MARS performs very well particularly for peak flows. However, it appears that the RNN-BR model performs well for low flow forecasting as compared to the MARS model. Although neither of the two models provides very good flow forecasts, it appears that the proposed method has a considerable potential for an improved long-term (12-month-ahead) forecast of the SLR flows. It is anticipated that the use of additional information such as precipitation and El-Niño (ENSO) indices would improve the model forecasting performance. The GSL storage volumes and Nile River flows are shown in Figs. 6.7 and 6.8, respectively. It appears that the RNN-BR performs very well for high as well as low lake volumes (Fig. 6.7) as compared to the MARS model. For the Nile River flows (Fig. 6.8), the proposed method is less accurate in predicting the peak flows than the low flows. As regards the other two cases, the

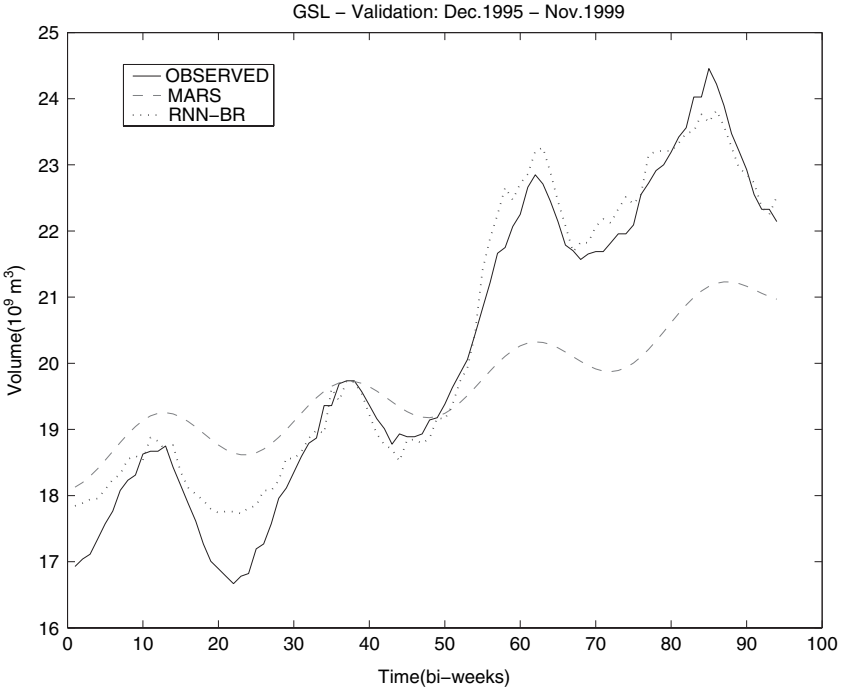


Fig. 6.7 Observed and predicted GSL bi-weekly volumes (Dec. 1995–Nov. 1999)

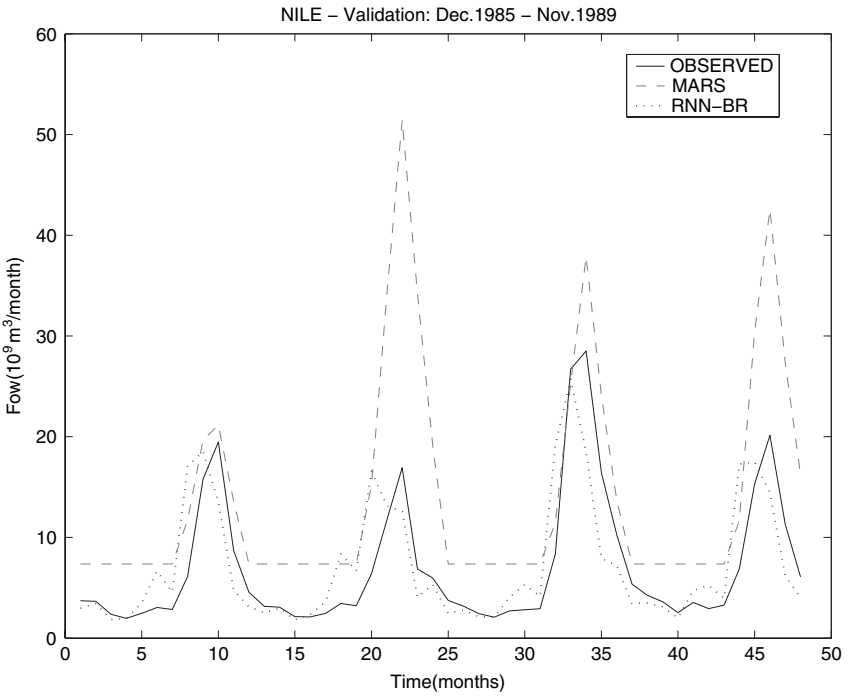


Fig. 6.8 Observed and predicted Nile River monthly flows (Dec. 1985–Nov. 1989)

RNN-BR model appears to be significantly better than the MARS model for the modeling of the Nile River flows.

To summarize, the optimal model directly identified provides significant forecast improvement over the MARS model with a dramatically reduced ($\sim$60% less) computational time. Therefore, the proposed method can be a good practical alternative for the modeling of complex hydrological systems. The proposed approach can be extended to include more recent RNN training algorithms such as the real-coded genetic algorithm (Blanco et al., 2001), and/or the extended Kalman filter method (Sum et al., 1999). Furthermore, it could also be improved by considering recently proposed data division techniques (Bowden et al., 2002), as well as additional explicative input variables.

6.5 Conclusions

An optimally trained dynamic RNN can be an effective method for modeling nonstationary hydrological time series. The proposed method allows a fast identification of an optimal time delay RNN model for complex hydrological system modeling. The optimum RNN-based model proposed in this study shows very promising results

for improving nonstationary hydrological time series forecasting without any data preprocessing (e.g., differencing or trend removal). Significant improvements are shown in the GSL storage volume and the Nile River flow forecasts as compared with those of the MARS model. However, the optimal RNN model does not provide satisfactory forecasts for the SLR flows, indicating that additional explanatory variables should be considered in this case. This study shows the promising potential of the dynamic RNN method in this context. Furthermore, it is anticipated that the method could be improved by including exogenous explanatory input variables such as low-frequency climatic indicators, as well as by considering other recent RNN training algorithms and data division techniques.

Acknowledgements The first author was partfunded by a grant from the Natural Sciences and Engineering Research Council (NSERC) of Canada.

References

American Society of Civil Engineers (ASCE) Task Committee on Application of Artificial Neural Networks in Hydrology, 2000. Artificial neural networks in hydrology, II, Hydrologic applications. J. Hydrol. Engng. ASCE 5(2), 124–137.

Anctil, F., Coulibaly, P., 2004. Wavelet analysis of the interannual variability in Southern Quebec streamflow. J. Clim. 1, 163–173

Bernier, J., 1994. Statistical detection of changes in geophysical series. In: Duckstein L., and Parent, E. (Eds.), Engineering Risk in Natural Resources Management, Kluwer Academic Publishers, Dordrecht, The Netherlands, pp. 159–176.

Blanco, A., Delgado, M., Pegalajar, M.C., 2001. A real-coded genetic algorithm for training recurrent neural networks. Neural Networks 14, 93–105.

Bowden, G.J., Maier, H.R., Dandy, G.C., 2002. Optimal division of data for neural network models in water resources applications. Water Resour. Res. 38(2), 2–11.

Box, G.E.P., Jenkins, G.M., 1976. Time Series Analysis: Forecasting and Control Holden-Day, San Francisco, CA, USA.

Coulibaly, P., Anctil, F., Aravena, R., Bobée, B., 2001. Artificial neural network modeling of water table depth fluctuations. Water Resour. Res. 37(4), 885–897.

Coulibaly, P., Anctil, F., Rasmussen, P.F., Bobée, B., 2000a. A recurrent neural networks approach using indices of low-frequency climatic variability to forecast regional annual runoff. Hydrol. Processes 14(15), 2755–2777.

Coulibaly, P., Baldwin, C.K., 2005. Nonstationary hydrological time series forecasting using nonlinear dynamic methods. Journal of Hydrology 307 (1–4), 164–174.

Coulibaly, P., Anctil, F., Bobée, B., 1999. Hydrological forecasting using artificial neural networks: The state of the art. Can. J. Civ. Engng. 26(3), 293–304.

Daubechies, I., 1990. The wavelet transform time-frequency localization and signal analysis. IEEE Trans. Inform. Theory 36, 961–1004.

Elman, J.L., 1990. Finding structure in time. Cogn. Sci. 14, 179–211.

Eltahir, E.A.B., 1996. El-Niño and the natural variability in the flow of the Nile River. Water Resour. Res. 32(1), 131–137.

Foufoula-Georgiou, E., Kumar, P., 1995. Wavelets in Geophysics, Academic Press, New York, USA.

Friedman, J.H., 1991. Multivariate adaptive regression splines. Ann. Stat. 19, 1–141.

Giles, C.L., Lawrence, S., Tsoi, A.C., 1997. Rule inference for financial forecasting using recurrent neural networks. Proc. IEEE/IAFE Conf. on Computational Intelligence for Financial Eng., IEEE press, Piscataway, N.J., pp. 253–259.

Hagan, M.T., Menhaj, M.B., 1994. Training feedforward networks with Marquardt algorithm. IEEE Trans. Neural Networks 5(6), 989–993.

Haykin, S., Li, L., 1995. Nonlinear adaptive forecasting of nonstationary signals. IEEE Trans. Signal Processing 43(2), 526–535.

Haykin, S., 1999. Neural Networks: A Comprehensive Foundation, Prentice-Hall, Upper Saddle River, N.J.

Iatrou, M., Berger, T.W., Marmarelis, V.Z., 1999. Modeling of nonlinear nonstationary dynamic systems with a novel class of artificial neural networks. IEEE Trans. Neural Networks 10(2), 327–339.

Jones, C.A.L., MacKay, D.J.C., 1998. A recurrent neural network for modeling dynamical systems. Comput. Neural Systems 9, 531–547.

Kohavi, Z., 1978. Switching and Finite Automata Theory, McGraw-Hill, New York, USA.

Labat, D., Ababou, R., Mangin, A., 2000. Rainfall-runoff relations for karstic springs. Part II: continuous wavelet and discrete orthogonal multiresolution analyses. J. Hydrol. 238, 149–178.

Lall, U., Mann, M., 1995. The Great Salt Lake: A barometer of low-frequency climatic variability. Water Resour. Res. 31(10), 2503–2515.

Lall, U., Sangoyomi, T., Abarbanel, H.D.I., 1996. Nonlinear dynamics of the Great Salt Lake: Nonparametric short-term forecasting. Water Resour. Res. 32(4), 975–985.

Maier, H.R., Dandy, G.C., 2000. Neural networks for the prediction and forecasting of water resources variables: A review of modelling issues and applications. Environ. Model. Software 15, 101–124.

Mann, E., Lall, U., Saltzman, B., 1995. Decadal-to-centenial-scale climate variability: Insights into the rise and fall of the Great Salt Lake. Geophys. Res. Lett. 22(8), 937–940.

Narendra, K., Parthasarathy, K., 1990. Identification and control of dynamical systems using neural networks. IEEE Trans. Neural Networks 1, 4–27.

Nash, J.E., Sutcliffe, J.V., 1970. River flow forecasting through conceptual models, Part I: A discussion of principles. J. Hydrol. 10, 282–290.

Pearlmutter, B.A., 1995. Gradient calculations for dynamic recurrent neural networks: A survey. IEEE Trans. Neural Networks 6(5), 1212–1228.

Saad, E.W., Prokhorov, D.V., Wunsch, D.C., 1998. Comparative study of stock trend forecasting using time delay, recurrent and probabilistic neural networks. IEEE Trans. Neural Networks 9(6), 1456–1470.

Sangoyomi, T.B., 1993. Climatic variability and dynamics of Great Salt Lake hydrology, Ph.D. dissertation, Utah State Univ., Logan, UT., 247p.

Shamseldin, A.Y. 1997. Application of a neural network technique to rainfall-runoff modeling. J. Hydrol. 199, 272–294.

Sum, J., Leung, C., Young, G.H., Kan, W., 1999. On the Kalman filtering method in neural network training and pruning. IEEE Trans. Neural Networks 10(1), 161–166.

Torrence, C., Campo, G.P., 1998. A practical guide to wavelet analysis. Bull. Am. Meteor. Soc. 79, 61–78.

Williams, R.J., Peng, J., 1990. An efficient gradient-based algorithm for on-line training of recurrent network trajectories. Neural Comput. 2, 490–501.

Young, P.C., 1999. Nonstationary time series analysis and forecasting. Progress Environ. Sci. 1, 3–48.

Chapter 7
Visualisation of Hidden Neuron Behaviour in a Neural Network Rainfall-Runoff Model

L.M. See, A. Jain, C.W. Dawson and R.J. Abrahart

Abstract This chapter applies graphical and statistical methods to visualise hidden neuron behaviour in a trained neural network rainfall-runoff model developed for the River Ouse catchment in northern England. The methods employed include plotting individual partial network outputs against observed river levels; carrying out correlation analyses to assess relationships among partial network outputs, surface flow and base flow; examining the correlations between the raw hidden neuron outputs, input variables, surface flow and base flow; plotting individual raw hidden neuron outputs ranked by river levels; and regressing raw hidden neuron outputs against river levels. The results show that the hidden neurons do show specialisation. Of the five hidden neurons in the trained neural network model, two appear to be modelling base flow, one appears to be modelling surface flow, while the remaining two may be modelling interflow or quick sub-surface processes. All the methods served to provide confirmation of some or all of these findings. The study shows that a careful examination of a trained neural network can shed some light on the sub-processes captured in its architecture during training.

Keywords Neural networks · hidden nodes · physical interpretation

L.M. See
School of Geography, University of Leeds, Woodhouse Lane, Leeds, LS2 9JT, UK, e-mail: l.m.see@leeds.ac.uk

A. Jain
Department of Civil Engineering, Indian Institute of Technology Kanpur, Kanpur 208 016, UP, INDIA

C.W. Dawson
Department of Computer Science, Loughborough University, Loughborough, LE11 3TU, UK

R.J. Abrahart
School of Geography, University of Nottingham, Nottingham, NG7 2RD, UK

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

7.1 Introduction

Neural networks (NNs) are often criticised for their black box nature, i.e. they transform a set of inputs to a required set of outputs with no knowledge of the physical processes involved. This aspect of neural networks has discouraged their widespread use despite ample evidence in the literature of good model performance, rapid model development times and speed of processing once trained (Hsu et al., 1995; Minns and Hall, 1996; Khonder et al., 1998; Tingsanchali and Gautam, 2000; Chang et al., 2002; Rajurkar et al., 2002; Baratti et al., 2003; Campolo et al., 2003; Agarwal and Singh, 2004). If the internal processing of neural networks could be visualised and understood, this might be one step towards encouraging their use in an operational environment. Interest in this area is not new but previous work has, in most cases, been restricted to the process of rule extraction (e.g. Andrews et al. 1995; Benitez et al. 1997; Lozowski et al. 1996) or the use of sensitivity analysis to provide improved understanding (Sudheer, 2005). More recently, exploration in hydrology has switched to an investigation of the functional behaviour of the hidden processing units (Wilby et al., 2003; Jain et al., 2004; Sudheer and Jain, 2004), which has suggested that hidden node specialisation exists. Moreover, individual hidden units might be related to hydrological processes.

In this chapter, a neural network model is developed for the Ouse catchment in northern England using historical levels and precipitation. The outputs corresponding to the hidden neurons are then examined for hidden neuron functional behaviour. A basic neural network structure is first provided to indicate the different hidden neuron outputs that can be examined. This is followed by a short description of the previous research in this area by Wilby et al. (2003), Jain et al. (2004) and Sudheer and Jain (2004) along with the methods adopted in this study. The results are discussed in light of what can be inferred about the behaviour of the hidden processing units. The chapter ends with a series of recommendations for further studies into the physical interpretation of neural networks.

7.2 Hidden Neuron Outputs

A basic feedforward neural network with one hidden layer is shown on the top of Fig. 7.1. Each layer consists of a number of processing units or neurons connected via weights. Data enter the network through the input units ($X_1 \ldots X_n$), which are then passed forward through the hidden layer to emerge from the output unit (O), where the network is used to predict a single output. Each hidden neuron first computes an intermediate value that comprises the weighted sum of all its inputs $I = \Sigma W_{ji}\ X_j$. This value is then passed through an activation or transfer function such as a logistic or sigmoid function. More information on neural network architecture can be found in Haykin (1999).

In order to examine the internal behaviour of a NN it is possible to inspect four different outputs, as shown in Fig. 7.1, where i is the number of hidden units:

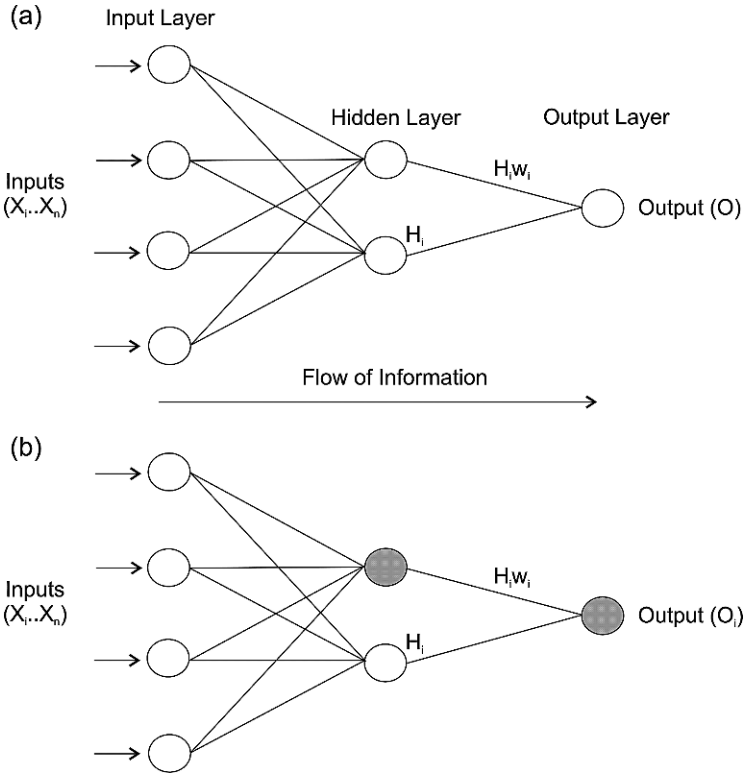


Fig. 7.1 Neural network architecture and possible hidden node outputs

- the raw hidden neuron outputs (H_i)
- the weighted hidden neuron outputs ($H_i w_i$)
- the partial network outputs ($O_{i,}$) comprising the contributions from the hidden neurons H_i to the total output O (see grey shaded neurons in Fig. 7.1(b) for an example)
- the total network output (O).

Ideally, one would also like to examine the partial contributions from more than one hidden neuron (e.g. $O_{1,2}$ which represents the sum of the weighted partial contributions from hidden neurons 1 and 2). However, this is not possible because summing the individual weighted O_i terms violates the principle of superposition. In order to do this, we suggest a fourth method outlined in more detail in Sect. 7.4.

7.3 Previous Studies

In this section, the earlier methodologies that were used in previous studies to examine hidden unit behaviour are presented. The first study by Wilby et al. (2003) used

precipitation, evaporation and discharge data for the groundwater-dominated Test River Basin in Hampshire to develop a conceptual and neural network rainfall-runoff model. Three experiments were undertaken in which the level of information presented to the neural networks was progressively decreased to determine the extent to which the model could emulate hydrological processes with different sets of information. In the first experiment, the neural network was developed using seven inputs with known correlations to daily discharge including moving averages. The second experiment used 12 inputs to reflect current daily and lagged inputs but without smoothing of the data inputs. This experiment was designed to examine the effect of antecedent hydrological conditions on hidden unit specialisation. In the third and final experiment, the neural network was developed using the same inputs as the conceptual hydrological model, i.e. only current values of daily precipitation and potential evapotranspiration, so no use was made of antecedent or smoothed values. The partial network outputs from each hidden unit (O_i) were then plotted against actual discharge to see what physical processes could be detected. The experiments showed that when the neural network model was provided with antecedent precipitation and evaporation, partial output associated with two of the hidden units was suggestive of base flow and quick sub-surface flow components. The third hidden unit appeared to map seasonal variations in soil moisture deficit.

Jain et al. (2004) and Sudheer and Jain (2004), in contrast to the previous study, reported experiments based on examining the raw hidden unit outputs (i.e. H_i). Jain et al. (2004) used a neural network trained with backpropagation to model the Kentucky River watershed in the United States at the 'Lock and Dam 10' Gauging Station on a 1-day time step for the periods 1960–1972 (training data set) and 1977–1989 (testing/validation data set). Trial and error procedures were used to determine the number of hidden neurons needed to model the rainfall-runoff process. The final chosen network had a 5:4:1 architecture and the hydrological function modelled was

$$Q_t = F(Q_{t-1}, Q_{t-2}, P_t, P_{t-1}, P_{t-2}) + e_t$$

where Q_t and P_t are observed river discharge and precipitation at time t, respectively, and e_t is the model error term. The authors found that the raw hidden unit outputs H_1, H_2 and H_4 had a strong negative correlation with past river discharge records. It was suggested that these units might be modelling either the base flow or the surface flow component of the rainfall-runoff process. H_3 outputs had a strong negative correlation with present and past rainfall and so might be modelling effective rainfall or infiltration processes.

The hidden neuron output values were also matched against five components simulated in a deterministic conceptual rainfall-runoff model: total computed flow, base flow, surface flow, soil moisture content and actual incremental infiltration. Each of the hidden unit outputs exhibited a negative correlation with total computed discharge. The strongest negative correlations with regard to other variables were: H_1 and H_4 with base flow; H_2 and H_4 with surface flow; H_1 and H_4 with soil moisture; H_3 and H_4 with incremental infiltration. Such division was considered to be consistent with their previous results in terms of correlations of H_is with the inputs, and it

was therefore concluded that H_1 might be a base flow component, H_2 a quick surface flow component, H_3 a rainfall infiltration component and H_4 a delayed surface flow component.

The final study by Sudheer and Jain (2004) involved the use of a neural network to model the Narmada River watershed in India at the Mandala Gauging Station on a 1-hour time step for the monsoon seasons of 1989–1991 (training data set) and 1992–1993 (testing/validation data set). Trial and error procedures were used to test a set of models that contained between 2 and 10 hidden units and the final architecture was a 6:3:1 NN with the following inputs: Q_{t-1}, Q_{t-2}, Q_{t-3}, Q_{t-4}, Q_{t-5} and Q_{t-6}. The three hidden neurons seemed to divide the activation function space into three parts as hypothesised by them. The raw hidden neuron outputs ($H_1 - H_3$) were ranked by discharge and plotted in the form of a rank curve or a flow duration curve. The rank curves of individual hidden neurons suggested that the neural network might be attempting to develop a set of hierarchical sub-domains in which specific hidden neurons were matched to specific subsets of the discharge record: H_1 to higher magnitudes; H_2 to lower magnitudes; and H_3 to medium magnitudes. The relationship between the hidden unit outputs and the discharge record was also reported to be positive for H_1 and H_2 and negative for H_3.

The first paper considered the relative difference between partial network outputs (O_i) and observed measurement records. No attempt was made to examine the contribution from more than one hidden node. The last two papers considered the relative difference between the raw hidden unit outputs (H_i) and observed measurement records. This chapter will apply the methods proposed by Wilby et al. (2003), Jain et al. (2004) and Sudheer and Jain (2004) to the Ouse River catchment. An additional method based on multiple linear regression analysis is then suggested that allows one to take contributions from more than one hidden node into account.

7.4 Study Area Data and Methodology

The data set used in this study comes from a set of stations on the River Ouse catchment in North Yorkshire, northern England (Fig. 7.2). This large catchment of over 3000 km^2 contains an assorted mix of urban and rural land uses with dissected uplands in the west that experience substantial precipitation, to cultivated lowlands in the east. Further details of this catchment are provided in Abrahart et al. (2005).

The City of York, which is situated near the catchment outlet, is prone to flooding problems. The neural network model developed as part of the study predicts river level at the Skelton hydrological gauging station, which is situated 5.5 km upstream of York. The lead time for prediction is 6 hours and input data are as follows: the river level at Skelton and three upstream gauging stations (US1 to 3) as well as data from five rain gauges (RG1 to 5). Data are sampled at 6 hourly intervals. Input data lags were correlated with the outputs to determine the optimal input data set. The

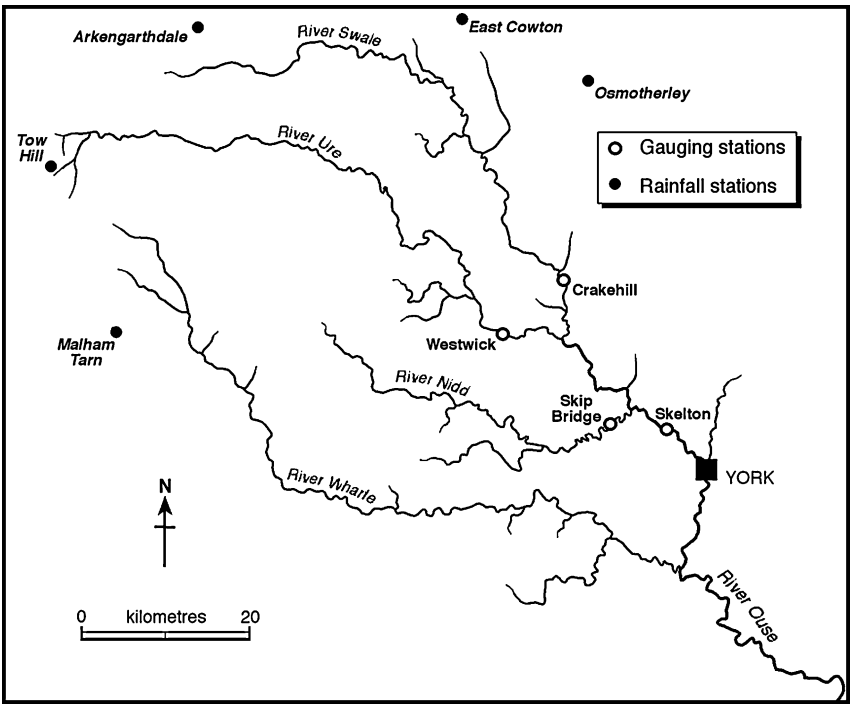


Fig. 7.2 Location of the catchment

model inputs are as follows: $U1_{T-1}$, $U2_{T-1}$, $U3_{T-2}$, $R1_{T-5}$, $R2_{T-5}$, $R3_{T-5}$, $R4_{T-6}$, $R5_{T-5}$, Q_{T-1} where $T-1$ represents one previous time step of 6 hours and the output is Q_T. No moving averages were used in this experiment. The data set used consists of historical records for the winter period (October–March) for 1993/1994.

A feedforward neural network with five hidden nodes and nonlinear activation functions was trained with backpropagation and momentum. Five hidden nodes were chosen so that a sufficient number was available for specialisation of different hydrological processes. Four different methods were then used to examine the hidden node behaviour:

1. Partial network outputs vs. observed values: the observed values of discharge and rainfall were plotted against O_1-O_5 as undertaken by Wilby et al. (2003). The flows were also separated into base and surface flow components and correlations undertaken with the partial network outputs.
2. Raw hidden neuron output correlations: the raw hidden neuron outputs were correlated against model inputs, surface flow and base flow as undertaken by Jain et al. (2004).
3. Raw hidden neuron outputs ranked by observed discharge values: the values of raw hidden neuron outputs H_1-H_5 were ranked against observed values of discharge as performed previously by Sudheer and Jain (2004).

4. Linear regression and examination of the residuals: the raw hidden node outputs $H_1 - H_5$ were regressed against observed river level, adding each output sequentially according to the strength of the relationship with the river level. The residuals were then plotted to examine the progressive contribution of the hidden nodes.

7.5 Results and Discussion

7.5.1 Partial Network Outputs vs. Observed Values

The partial network outputs from hidden neurons 1–5 and the observed river levels were plotted against time as shown in Fig. 7.3 (for a subset) and as scatterplots in Fig. 7.4. The flow was also separated into base flow and surface flow components using the local minimum method with the Web-Based Hydrograph Analysis Tool (WHAT) developed by Lim et al. (2005). These separated flows were then further correlated against the partial network outputs, $O_1 - O_5$. These correlations are provided in Table 7.1.

It can be observed from Fig. 7.3 that the partial network output O_2 has the least variation over time and O_4 has the largest. Furthermore, O_4 operates at the highest magnitudes whereas all other partial network outputs ($O_1 - O_3$ and O_5) operate at comparable levels, with O_2 enveloping all the others. A close look at Fig. 7.4 suggests that the dependence of O_1 and O_2 on O is similar in nature. Similarly, the behaviour of O_3 and O_5 appears to be similar, while the behaviour of O_4 stands

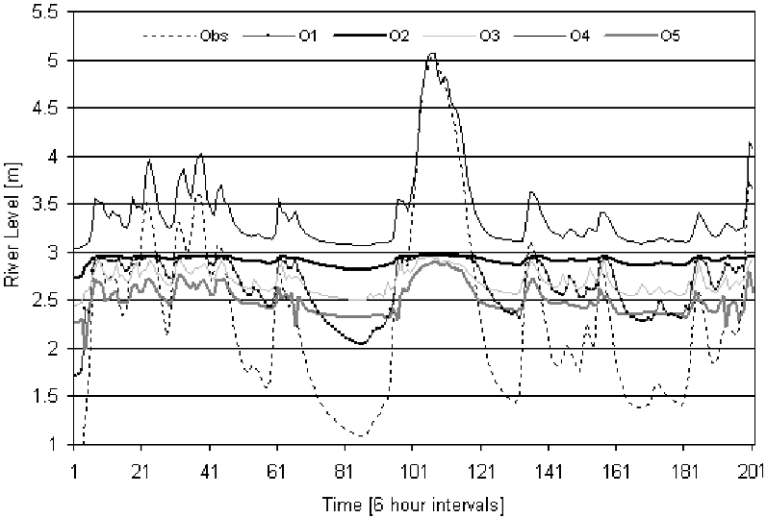


Fig. 7.3 Plot of contributory outputs $O_1 - O_5$ against observed river level for a subsection of the data

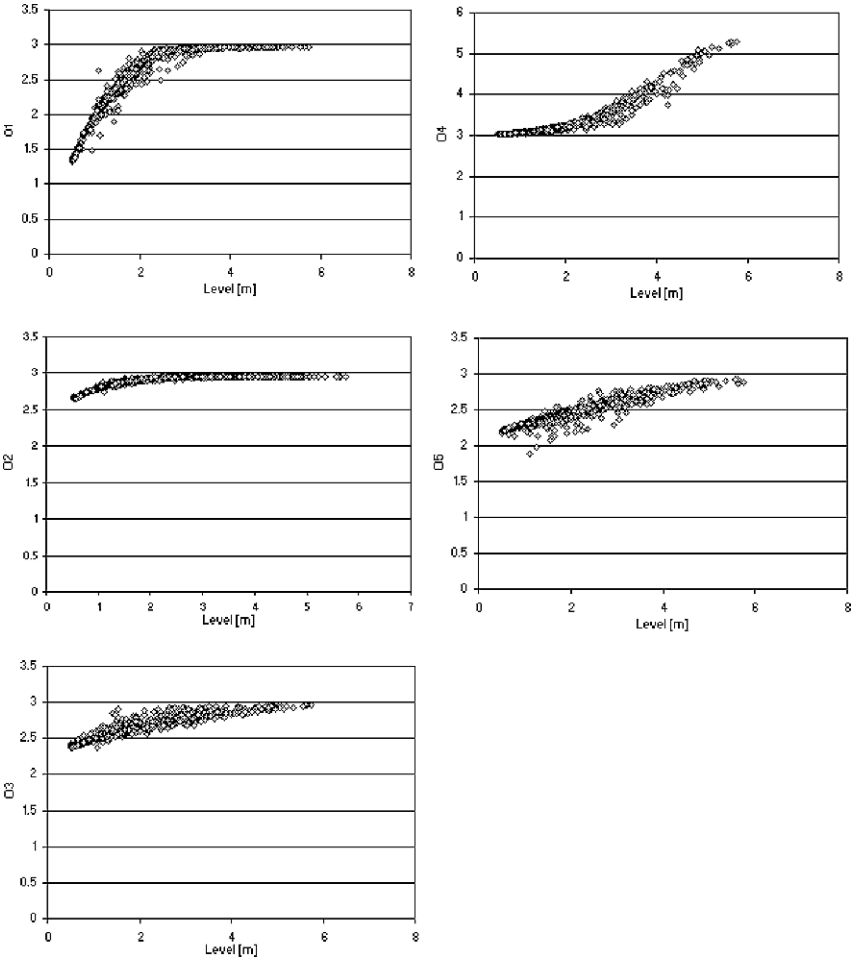


Fig. 7.4 Scatterplots of $O_1 - O_5$ against river level

out as different from the rest. These patterns indicate that the hidden neurons corresponding to O_1 and O_2 may be modelling the same or a similar hydrological subprocess. Likewise, the hidden neurons O_3 and O_5 may be modelling the same or a similar hydrological sub-process while O_4 appears to be modelling a completely different one.

Next we look at the magnitudes of the correlations of $O_1 - O_5$ with surface and base flow to investigate further which hidden neuron is more associated with one of these two sub-processes. From Table 7.1, it is clear that O_1 and O_2 (which are modelling the same or a similar sub-process) are strongly correlated with base flow but are poorly correlated with surface flow. Although other partial network outputs are also strongly correlated with base flow ($O_3 - O_5$), they are also reasonably or strongly correlated with surface flow. This suggests that the hidden neurons

Table 7.1 Correlation coefficients for O_1-O_5 with base and surface flow components

Contributory node	Surface flow component	Base flow component
O_1	0.2931	0.7249
O_2	0.2459	0.6526
O_3	0.4918	0.7055
O_4	0.7405	0.5263
O_5	0.5324	0.7406

corresponding to O_1 and O_2 are modelling base flow. The strong correlation of O_4 with surface flow indicates that the hidden neuron corresponding to O_4 is modelling surface flow (as its correlation with base flow is the weakest from among all O_i). Furthermore, the correlations of O_3 and O_5 (which seem to have similar behaviour) with base flow and surface flow are moderate, indicating the hidden neurons corresponding to them may be modelling interflow. In summary, it may be said that O_1 and O_2 are capable of capturing the long-term memory effects in the catchment (since base flows take longer to appear at the catchment outlet), which are mostly prevalent in the saturated sub-surface zone of the catchment. Similarly, O_4 is capable of modelling the short-memory effects (surface flow) in the catchment. Since O_3 and O_5 are operating somewhere in between, they may be modelling the unsaturated zone dominated by interflow or quick sub-surface flow.

7.5.2 Correlations with Raw Hidden Neuron Outputs

Correlation coefficients were calculated to determine the relationships between the raw hidden neuron outputs, H_1-H_5, and the model inputs as shown in Table 7.2. Unlike the findings of Jain et al. (2004), there was less observable specialisation in the hidden neurons. All hidden nodes are strongly correlated with the level at Skelton as well as the values at upstream stations but correlations are generally lower for the rainfall. Also presented in Table 7.2 are the correlations of H_1-H_5 with surface and base flows. It may be noted that H_4 has the highest correlation with surface flow confirming the earlier finding that H_4 is modelling surface flow. Furthermore, H_1 and H_2 have strong correlations with base flow and the weakest correlations with surface flow indicating that they are modelling base flow. This is further strengthened by the poor correlations of H_1 and H_2 with all five rainfall inputs. The correlations of H_3 and H_5 with surface and base flow are moderate confirming that they are modelling interflow or the unsaturated zone of sub-surface processes. This is further strengthened by the fact that H_3 and H_5 are moderately correlated with rainfall, which affects soil moisture. In fact the correlation of H_3 with rainfall is the highest, suggesting that it is strongly related to soil moisture.

Table 7.2 Correlation coefficients for $H_1 - H_5$ and the model inputs

Input	H_1	H_2	H_3	H_4	H_5
Q	−0.8217	−0.7825	−0.9017	0.8714	−0.9334
US1	−0.6914	−0.6461	−0.8465	0.9432	−0.8877
US2	−0.7145	−0.6754	−0.8551	0.9021	−0.9374
$US3_{t-1}$	−0.7637	−0.7385	−0.8777	0.8668	−0.8340
$RG1_{t-4}$	−0.2936	−0.2875	−0.4944	0.4877	−0.3205
$RG2_{t-4}$	−0.3266	−0.3115	−0.4801	0.4480	−0.2211
$RG3_{t-4}$	−0.1845	−0.2081	−0.3576	0.1363	−0.1591
$RG4_{t-5}$	−0.1971	−0.2088	−0.1569	0.1320	−0.1414
$RG5_{t-4}$	−0.2888	−0.3073	−0.5555	0.3840	−0.3083
QS_t	−0.5272	−0.4960	−0.7002	0.8466	−0.7265
QB_t	−0.8417	−0.8079	−0.8401	0.6630	−0.8610

Note: Q, US1, US2 and US3 represent river levels; RGs represent rainfall; and QS and QB represent surface flow and base flow, respectively.

7.5.3 Raw Hidden Neuron Outputs Ranked by Observed Values

Figure 7.5 contains the ranked raw hidden neuron outputs ($H_1 - H_5$) plotted against river level. It can be noticed that the hidden neuron H_4 operates at high levels, which confirms the role of this neuron in modelling surface flow. H_5 operates at medium levels, which suggests that this neuron may be modelling interflow. H_1 and H_2 operate at low levels and therefore do appear to be modelling surface flow.

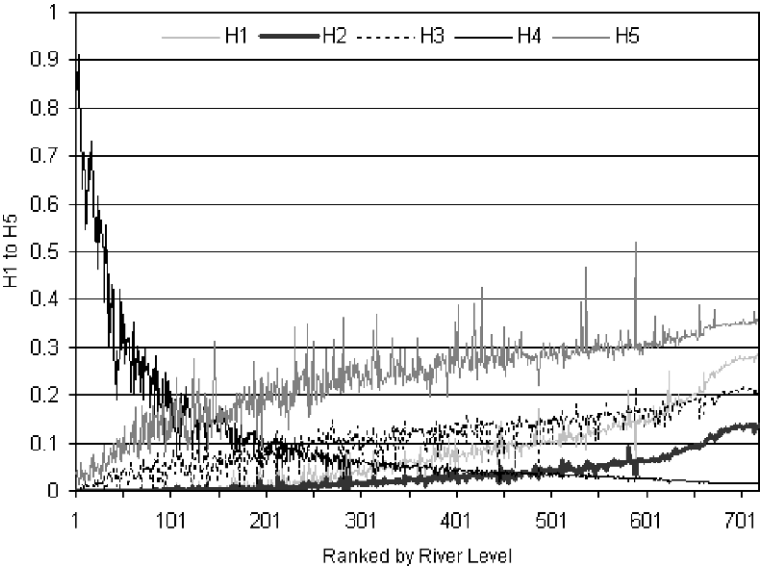


Fig. 7.5 Plots of $H_1 - H_5$ ranked against river level

Conclusions about H_3 are more difficult to make from this analysis as it appears to be operating in a similar range to H_1 and H_2. The maximum values of the ranges in which the hidden neurons operate are 0.285, 0.138, 0.289, 0.917 and 0.521 for H_1-H_5, respectively. This clearly suggests that H_1 and H_2 are modelling base flow, H_3 and H_5 are modelling interflow and H_4 is modelling surface flow. Therefore, the three methods investigated so far appear to corroborate with each other very well. The fourth method of linear regression and analysis of residuals is presented next.

7.5.4 Linear Regression and Examination of the Residuals

The observed river level was regressed using stepwise linear regression against values of H_1-H_5. The resulting equations are (in stepwise order)

$$\text{Level} = -11.927 + 5.702^{*} H_5 \; [R^2 = 0.881]$$

$$\text{Level} = -9.853 + 3.404^{*} H_5 + 1.061^{*} H_4 [R^2 = 0.930]$$

$$\text{Level} = -7.217 + 1.104^{*} H_5 + 1.331^{*} H_4 + 0.852^{*} H_1 [R^2 = 0.976]$$

$$\text{Level} = -2.589 + 1.097^{*} H_5 + 1.305^{*} H_4 + 1.201^{*} H_1 - 1.870 \; {}^{*} H_2 [R^2 = 0.976]$$

$$\text{Level} = -1.750 + 0.909^{*} H_5 + 1.246^{*} H_4 + 1.212^{*} H_1 - 2.632 \; {}^{*} H_2 + 0.749^{*} H_3 [R^2 = 0.977]$$

These equations were applied to the data and the residuals were plotted against the river level. By adding in the effect of the hidden nodes one at a time, we may be able to understand more about their individual behaviours. Figure 7.6 provides three plots of the residuals. The first is based on only H_5 and it shows that the scatter at high-magnitude flows is less as compared to that at low-magnitude flows. This indicates that H_5 may be modelling high or moderate magnitude flow (surface or interflow), which partially agrees with the earlier findings.

The second graph on the right of Fig. 7.6 shows the effect of when H_4 and H_1 are added. This has significantly reduced the over and underprediction at both low and high levels. This is unsurprising given that H_4 is thought to model surface flow and therefore contributes to better predictions at high levels while H_1 is thought to model base flow, thereby improving the low flow predictions. Finally, the third figure shows the residuals using all hidden neurons. There is very little difference except to reduce overprediction at medium to low levels. Therefore, the results of the regression also confirm the earlier findings but not as strongly as the other methods.

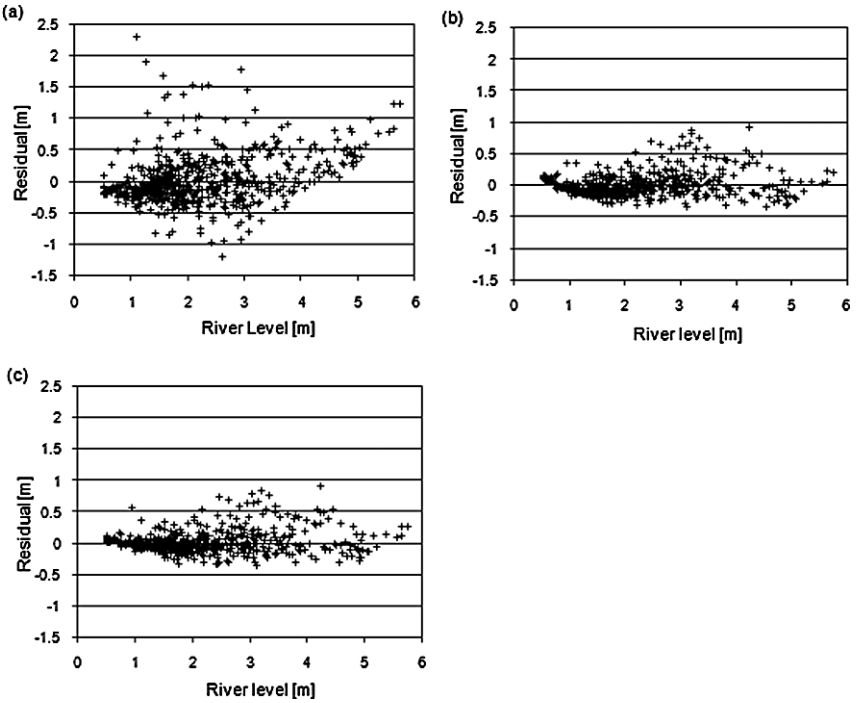


Fig. 7.6 Plot of residuals for (**a**) H_5 against observed flow (**b**) with the addition of H_1 and H_4 (**c**) and with all hidden nodes in the regression equation

7.6 Conclusions

Four different methods were used to visualise hidden neuron specialisation in a trained neural network rainfall-runoff model, which was developed for the River Ouse catchment in northern England. The methods investigated included plotting partial network outputs against observed river levels; separating the level into base flow and surface flow components for correlation with partial network outputs; examining the correlations of raw hidden neuron outputs with input variables, surface flow and base flow components; plotting individual raw hidden neuron outputs ranked by river levels; and regressing raw hidden neuron outputs against river levels. The results showed that the hidden neurons do show specialisation: two of the hidden neurons appear to be modelling base flow, one appears to be strongly modelling surface flow, while the remaining two may be modelling quick sub-surface flow or interflow. All the methods investigated served to provide confirmation of some or all of these findings.

The results also seemed to indicate that the second and possibly third hidden neurons were doing very little overall. This might provide an argument for reducing the number of hidden nodes to three or four and repeating the exercise to determine the effect on specialisation in relation to different hydrological sub-processes. It

would also be useful to develop a conceptual model that calculates soil moisture, infiltration, etc. to determine whether these hydrological processes are captured by the hidden nodes.

References

Abrahart RJ, See LM, Heppenstall AJ (2005) Using four different least-squared error functions to train a neural network rainfall-runoff model. In: Proceedings of the 2nd Indian International Conference on Artificial Intelligence (IICAI2005), Pune, India, 20–22 Dec 2005

Agarwal A, Singh RD (2004) Runoff modelling through back propagation artificial neural network with variable rainfall-runoff data. Water Resources Management 18(3): 285–300

Andrews R, Diederich J, Tickle AB (1995) A survey and critique of techniques for extracting rules from trained neural networks. Knowledge Based Systems 8: 373–389

Baratti R, Cannas B, Fanni A, Pintus M, Sechi GM, Toreno N (2003) River flow forecast for reservoir management through neural networks. Neurocomputing 55(3–4): 421–437

Benitez JM, Castro JL, Requena I (1997) Are artificial neural networks black boxes? IEEE Transactions on Neural Networks 8(5): 1156–1164

Campolo M, Soldati A, Andreussi P (2003) Artificial neural network approach to flood forecasting in the River Arno. Hydrological Sciences Journal 48(3): 381–398

Chang FJ, Chang LC, Huang HL (2002) Real-time recurrent learning neural network for stream-flow forecasting. Hydrological Processes. 16(3): 2577–2588

Haykin S (1999) Neural Networks: A Comprehensive Foundation. McMillan, New York

Hsu KL, Gupta HV, Sorooshian S (1995) Artificial neural networks modeling of the rainfall-runoff process. Water Resources Research 31: 2517–2530

Jain A, Sudheer KP, Srinivasulu S (2004) Identification of physical processes inherent in artificial neural network rainfall runoff models. Hydrological Processes 18: 571–581

Khonder MUH, Wilson G, Klinting A (1998) Application of neural networks in real time flash flood forecasting. In Babovic V, Larsen CL (eds) Proceedings of the Third International Conference on HydroInformatics, A.A. Balkema, Rotterdam, pp. 777–782

Lim KJ, Engel BA, Tang Z, Choi J, Kim K, Muthukrishnan S, Tripathy D (2005) Automated Web GIS Based Hydrograph Analysis Tool, WHAT. Journal of the American Water Resources Association 41(6): 1407–1416

Lozowski A, Cholewo TJ, Zurada JM (1996) Crisp rule extraction from perceptron network classifiers. In: Proceedings of the IEEE International Conference on Neural Networks: Plenary, Penal and Special Sessions, Washington DC, pp. 94–99

Minns AW, Hall MJ (1996) Artificial neural networks as rainfall-runoff models. Hydrological Sciences Journal. 41(3): 399–417

Rajurkar MP, Kothyari UC, Chaube UC (2002) Artificial neural networks for daily rainfall-runoff modeling. Hydrological Sciences Journal 47: 865–877

Sudheer KP, Jain A (2004) Explaining the internal behaviour of artificial neural network river flow models. Hydrological Processes 18: 833–844

Sudheer KP (2005) Knowledge extraction from trained neural network river flow models. Journal of Hydrologic Engineering 10(4): 264–269

Tingsanchali T, Gautam MR (2000) Application of tank, NAM, ARMA and neural network models to flood forecasting. Hydrological Processes 14(14): 2473–2487

Wilby RL, Abrahart RJ, Dawson CW (2003) Detection of conceptual model rainfall-runoff processes inside an artificial neural network. Hydrological Sciences Journal 48(2): 163–181

Chapter 8
Correction of Timing Errors of Artificial Neural Network Rainfall-Runoff Models

N.J. de Vos and T.H.M. Rientjes

Abstract In this study, multi-layer feedforward artificial neural network (ANN) models were developed for forecasting the runoff from the Geer catchment in Belgium. The models produced a good overall approximation of the hydrograph, but the forecasts tended to be plagued by timing errors. These were caused by the use of previous discharge as ANN input, which became dominant and effectively caused lagged forecasts. Therefore, an aggregated objective function was tested that punishes the ANN model for having a timing error. The gradient-based training algorithm that was used had difficulty with finding good optima for this function, but nevertheless some hopeful results were found. There seems to be a trade-off between having good overall fit and having correct timing, so further research is suggested to find balanced ANN models that satisfy both objectives.

Keywords Artificial neural network, rainfall-runoff modelling, timing errors, objective functions

8.1 Introduction

River discharge forecasts are required for successfully managing the consequences of hydrological extremes. Forecasting models are often catchment-scale models that simulate the transformation of rainfall into river runoff. Because this transformation involves a number of interacting processes that are complex and spatially/temporally variable, such simulation is not an easy task. One approach to rainfall-runoff modelling is to use the so-called data-driven techniques, which are based on extracting and re-using information that is implicitly contained in hydrological data, and

N.J. de Vos
Water Resources Section, Faculty of Civil Engineering and Applied Geosciences, Delft University of Technology, 2600 GA, Delft, The Netherlands

T.H.M. Rientjes
Department of Water Resources, Institute for Geo-Information Science and Earth Observation (ITC), 7500 AA, Enschede, The Netherlands

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

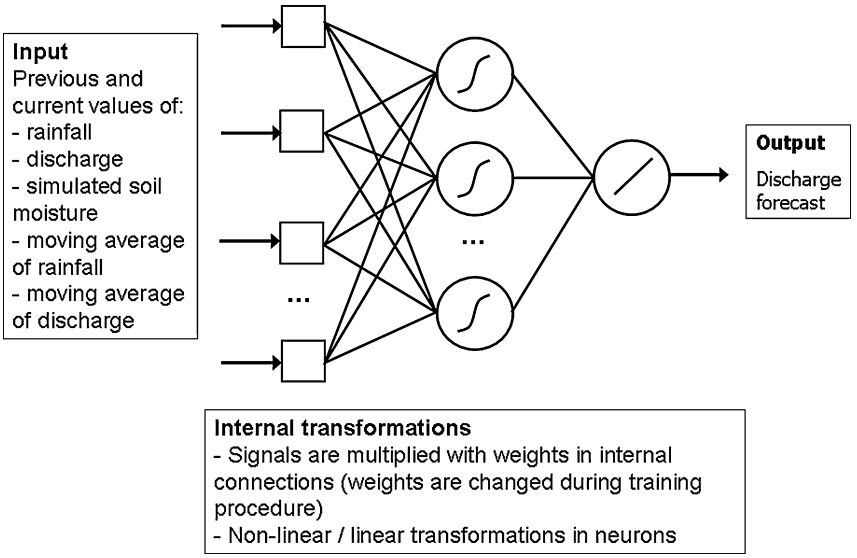


Fig. 8.1 Multi-layer feedforward ANN for rainfall-runoff modelling

which do not directly take into account the physical laws that underlie the rainfall-runoff transformation (as knowledge-driven models such as physically based and conceptual models do).

In this study, we have investigated the popular data-driven approach of artificial neural network (ANN) modelling for forecasting river runoff (see Fig. 8.1). ANNs are mathematical models that consist of simple, densely interconnected elements known as neurons. An ANN receives input signals that are propagated and transformed through the network's neurons towards the output neuron(s). One of the key transformations performed by an ANN is multiplication with weights that express the strength of connections between neurons. During a training procedure, the network weights, and therefore the model's response, are adapted to sample information that is presented to the network. The goal is to minimise an objective function that expresses the difference between the ANN response to sample input and target output data.

An ANN is able to simulate rainfall-runoff processes by mapping the transformation from catchment inputs and/or states (e.g. rainfall, evaporation, soil moisture content) to outputs (e.g. river discharge or water levels). Since the middle of the 1990s, there have been numerous studies on this approach to rainfall-runoff modelling (e.g. Hsu et al. 1995; Smith and Eli 1995; Minns and Hall 1996; Shamseldin 1997; Campolo et al. 1999; Abrahart and See 2000; Toth et al. 2000; Dibike and Solomatine 2001; Anctil et al. 2004; Jain and Srinivasulu 2004; De Vos and Rientjes 2005), but it remains a topic of ongoing research.

The ability to exploit the total information content of ANN inputs depends strongly on the training of the network. First, a training algorithm must be able to search the parameter space extensively and efficiently. Second, the objective

Table 8.1 Numerical performance measures used in this study

Nash–Sutcliffe coefficient of efficiency (CE) (Nash and Sutcliffe 1970)	$1 - \left(\frac{\sum_{k=1}^{K} \left(Q_k - \hat{Q}_k\right)^2}{\sum_{k=1}^{K} \left(Q_k - \bar{Q}\right)^2} \right)$
Persistence Index (PI) (Kitanidis and Bras 1980)	$1 - \left(\frac{\sum_{k=1}^{K} \left(Q_k - \hat{Q}_k\right)^2}{\sum_{k=1}^{K} \left(Q_k - Q_{k-L}\right)^2} \right)$
Mean squared logarithmic error (MSLE)	$\left(\frac{\sum_{k=1}^{K} \left(\ln Q_k - \ln \hat{Q}_k\right)^2}{K} \right)$
Timing error (TE)	Time shift at which CE is maximal
Peak error (PE)	$\max \left(\sum_{k=1}^{K} Q_k - \hat{Q}_k \right)$
Bias (B)	$\frac{\sum_{k=1}^{K} Q_k - \hat{Q}_k}{K}$

function that is used for evaluating model performance should be appropriate for the application under investigation. Singular objective functions based on squared-error-based performance measures, such as the mean squared error (MSE) and the Nash–Sutcliffe coefficient of efficiency (CE), see Table 8.1, are commonly used in rainfall-runoff modelling. However, not all differences between modelled and observed hydrograph characteristics such as timing, volume and magnitudes can be adequately expressed by a single performance measure. The aspect of model evaluation in the training of ANNs for rainfall-runoff modelling has hitherto received little attention.

In this research, we investigate the evaluation of ANN rainfall-runoff models during the training phase and the consequences of choosing certain objective functions. A variety of numerical performance measures, which are employed in both model training and validation, are shown in Table 8.1. They are discussed in more detail in Sect. 8.3.

8.2 Study Site and Data

The River Geer (Fig. 8.2) is located in the north of Belgium, northwest Europe, and contributes to the River Meuse. The river's catchment size is 494 km^2. The mean annual rainfall is approximately 810 mm, and the perennial river has discharge

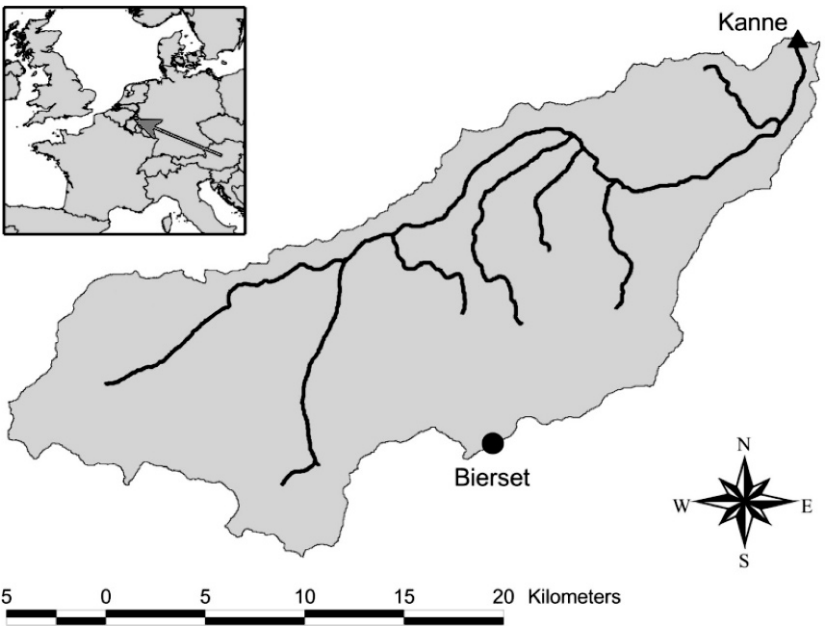


Fig. 8.2 Geer River basin (Belgium)

ranging from 1.8 m^3/s in dry periods to 10 m^3/s during wet periods. The measurement data sets that were available for the period 1993–1997 included hourly rainfall and daily evaporation at station Bierset, and hourly streamflow at the catchment outlet at Kanne. Figure 8.3 shows the hourly catchment discharge at Kanne in combination with the rainfall at Bierset for the complete period. These time series were divided into 55% for training, 25% for cross-validation and 20% for validation, as noted in Fig. 8.3. All three fragments of the time series start with a period of constant low discharge and rainfall. Some descriptive statistics of the discharge in the three periods are presented in Table 8.2.

A number of simulated time series that can be considered indirect indicators of the hydrological state of a catchment were also used as ANN inputs. These include time series of the non-decaying moving average of the discharge (*Qma*) and the rainfall (*Pma*). By trial and error we found that memory lengths of 192 hours (8 days) and 480 hours (20 days) for the *Qma* and *Pma*, respectively, produced the best results. Lastly, a number of simulations using the simple soil moisture reservoir component of the GR4J lumped conceptual rainfall–runoff model (Edijatno et al. 1999; Perrin et al. 2003) were performed to produce a time series of estimated soil moisture (*SM*). The hourly rainfall time series and temporally downscaled evaporation time series served as input to the GR4J soil moisture model component. The only parameter that needed to be defined is the reservoir's maximum capacity *A*, of which a value of 400 mm produced the best results in this study.

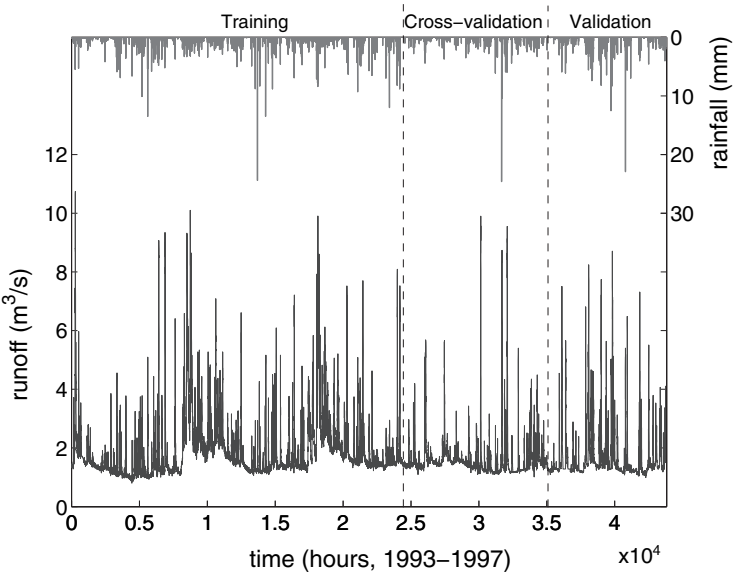


Fig. 8.3 Hourly runoff (Kanne) and rainfall (Bierset) from 1 January 1993 to 31 December 1997

Table 8.2 Descriptive statistics of the (a) precipitation data, (b) discharge data

	(a)					
	Min	Max	Mean	St. dev.	Skewness	Kurtosis
Training	0.00	107.6	2.80	5.72	5.14	53.1
Cross-validation	0.00	35.6	2.28	4.81	3.33	13.1
Validation	0.00	51.1	2.03	4.39	4.00	23.9
	(b)					
	Min	Max	Mean	St. dev.	Skewness	Kurtosis
Training	1.10	14.55	2.61	1.08	3.15	16.8
Cross-validation	0.96	8.96	2.09	0.98	2.83	11.6
Validation	1.10	8.84	1.74	0.78	3.71	18.8

8.3 Methods

8.3.1 Artificial Neural Network Design

The type of ANN used in this study is the static multi-layer feedforward network. Static networks do not have the dimension of time incorporated in the network architecture, as opposed to dynamic networks, which use feedback connections or local memories in neurons. These static ANNs are nevertheless able to represent the dynamics of a system in the network model by using the so-called tapped delay lines, which present a sequence of time series values (e.g. $P(t), P(t-1), \ldots, P(t-m)$)

as separate network input signals. $P(t)$ represents an input variable in time and m is the size of the time window. The number of input units thus increases with the size of this window. Linear correlation between different input variable time series and the target output discharge time series was used to find optimal windows for all input variables. Our ANN models use values of P from t–5 to t–19, along with the last three known values of Q, SM, *Qma* and *Pma* as model inputs to forecast Q at $t+1$. The inputs for forecasts with higher lead times are identical, but with a shifted time window for P. The best ANN topology that was found through trial and error is one hidden layer with four neurons (resulting in a 27-4-1 ANN). Because the hyperbolic tangent transfer functions that are used in the ANN neurons become saturated at a certain range, all input data are linearly scaled to a range of -1 to 1. The output of this transfer function is bounded to the range of -1 to 1, which is why the output data were scaled to a range of -0.8 to 0.7. The reason for setting these ranges narrow is to enable the ANN to extrapolate beyond the training data range. The output data range is asymmetrical because it is more likely that the upper bound of the training data range is exceeded than the lower bound.

8.3.2 Training

The Levenberg–Marquardt algorithm was used for ANN training. This quasi-Newtonian algorithm proved to give fast and accurate results. We followed the common practice of initialising the ANN weights randomly at the start of each training trial. The goal of this randomisation is to force the training algorithm to search other parts of the parameter space, thereby enabling a more robust overall optimisation procedure and increasing the overall chances of finding a global error minimum. A result of this approach is that, because of algorithm imperfections, the performance of an ANN is often different for each training trial, even if it is trained using the same algorithm (De Vos and Rientjes 2005). This is why the results are presented over 50 training trials, thus enabling more reliable conclusions to be drawn.

8.3.3 Model Evaluation

Table 8.1 lists the numerical performance measures used in this research, some of which were used as objective functions in the training of ANNs. The Nash–Sutcliffe coefficient (CE) and the Persistence Index (PI) are based on squaring the residuals and in practice they are therefore more focused on peak flows than on low flows. The mean squared logarithmic error (MSLE) is used because it weighs low flows more than squared-error measures. It is based on the logarithmic function used by Hogue et al. (2000). The timing error (TE) used is defined as the time shift of the entire forecast time series for which the CE is at a maximum, and is therefore a measure of the overall timing error of the model. The time shifts over which this check is performed

varied from −20 to +20 time steps. The peak error (PE) expresses the maximum error of the simulated time series (which usually occurs during peak flows). The bias (B) is a measure of a model's tendency to underestimate or overestimate flows.

Training was also done using a number of aggregated objective functions, which consist of the simple product of two or more of the singular functions used above. In aggregated objective functions that use the TE, we have multiplied the other error with a factor TEf equal to 500 if the TE is non-zero and equal to 1 if the TE is zero. The value of 500 was selected by trial and error from a set of arbitrarily chosen numbers. This way we penalise the model for having a TE other than zero. The idea for this method is taken from Conway et al. (1998) who used a genetic algorithm to train ANN models for predicting solar activity.

8.4 Results

The ANN performance results in terms of mean and standard deviation over the best 40 out of 50 training trials for lead times of 1 and 6 hours are presented in Tables 8.3a and 8.4a. The 10 worst-performing trials were deleted because they are outliers that are not representative for the ANN model behaviour. The single 'best' results out of these trials are shown in Tables 8.3b and 8.4b. The ANNs are trained by five different training objective functions that are shown in the left hand column and performance results over the validation period are presented for all six numerical performance measures of Table 8.1. The models trained on CE and MSLE perform quite well since both the high flows and the low flows are adequately simulated judging from these two criteria. These two objective functions yield similar model performance, and it appears there is a significant correlation between both functions. This is likely due to the fact that the time series shows no clear dry and wet periods but instead a regular seasonal pattern is observed, which is why the ANN model and algorithm do not make a clear distinction between the CE and the MSLE as training objective functions.

However, the simulations with CE and MSLE as objective functions show that the ANN model has problems with the correct timing of forecasts (see TE and PI). The one-step-ahead forecasts suffer from a TE that is as big as the lead time and the PI is close to zero or even negative, indicating that the model is barely any better than a simple persistence model. What the model in fact does is nothing more than presenting the latest measured discharge, which is an input to the ANN model, as a forecast for the lead time.

This phenomenon is illustrated in more detail in Fig. 8.4, which shows forecast results for various lead times, evaluated in terms of CE (shown on the ordinate), and for various shifts in time of the forecasted versus the observed time series (shown on the abscissa). The CE at zero shift corresponds to the actual performance of the models. The predicted time series is subsequently shifted in time against the observed time series, after which CE is recalculated. The time shift at which the CE coefficient is maximised is the TE. This is done for a number of different lead times

Table 8.3 Performance of 1-hour-ahead-predicting ANN models that are trained with various performance measures: (a) mean and standard deviation over 40 simulations, (b) 'best' trained models

		(a)					
		Performance on validation data					
		CE	PI	TE	MSLE	B [$^{*}10^{-1}$]	PE
Training objective function	CE	0.96±0.00	0.13±0.04	−1.00 ± 0.00	24.6±2.3	0.07±0.07	2.75±0.17
	MSLE	0.96±0.00	0.13±0.02	−1.00 ± 0.00	24.4±1.1	0.06±0.04	2.73±0.15
	CE*TEf	0.52±0.48	−10.5 ± 11.5	−0.89 ± 0.39	750±800	1.04±3.21	3.17±1.52
	MSLE*TEf	0.62±0.37	−8.2 ± 9.0	−0.90 ± 0.50	586±816	0.84±2.16	3.47±2.04
	CE*MSLE*TEf	0.89±0.11	−1.55 ± 2.56	−1.00 ± 0.00	115±147	0.38±0.85	3.17±0.88
		(b)					
		Performance on validation data					
		CE	PI	TE	MSLE	B [$^{*}10^{-1}$]	PE
Training objective function	CE	0.97	0.23	−1.00	21.6	0.04	2.01
	MSLE	0.97	0.17	−1.00	22.1	0.08	2.78
	CE*TEf	0.96	0.12	−1.00	22.8	0.08	2.59
	MSLE*TEf	0.96	0.16	−1.00	23.8	0.04	2.64
	CE*MSLE*TEf	0.97	0.20	−1.00	21.7	0.04	2.56

Table 8.4 Performance of 6-hour-ahead predicting ANN models that are trained with various performance measures: (a) mean and standard deviation over 40 simulations, (b) 'best' trained models

		(a)					
		Performance on validation data					
		CE	PI	TE	MSLE	B [$*10^{-1}$]	PE
Training objective function	CE	0.80 ± 0.01	0.51 ± 0.02	-1.05 ± 0.22	130 ± 9	0.28 ± 0.15	3.97 ± 0.72
	MSLE	0.80 ± 0.01	0.49 ± 0.02	-1.15 ± 0.36	138 ± 9	0.30 ± 0.25	4.15 ± 0.57
	CE*TEf	0.18 ± 0.57	-1.04 ± 1.43	-0.93 ± 1.87	1410 ± 2290	1.31 ± 2.57	4.88 ± 5.42
	MSLE*TEf	0.31 ± 0.42	-0.71 ± 1.05	-0.93 ± 1.87	875 ± 656	1.59 ± 1.64	3.82 ± 3.24
	CE*MSLE*TEf	0.46 ± 0.38	-0.35 ± 0.93	-0.61 ± 1.16	632 ± 513	1.57 ± 1.57	3.74 ± 1.60
		(b)					
		Performance on validation data					
		CE	PI	TE	MSLE	B [$*10^{-1}$]	PE
Training objective function	CE	0.82	0.55	−1.00	123.3	0.23	2.75
	MSLE	0.82	0.56	−1.00	115.0	−0.02	3.49
	CE*TEf	0.78	0.46	0.00	150.3	0.18	5.85
	MSLE*TEf	0.78	0.45	0.00	157.6	0.31	5.03
	CE*MSLE*TEf	0.79	0.47	0.00	140.7	0.37	4.88

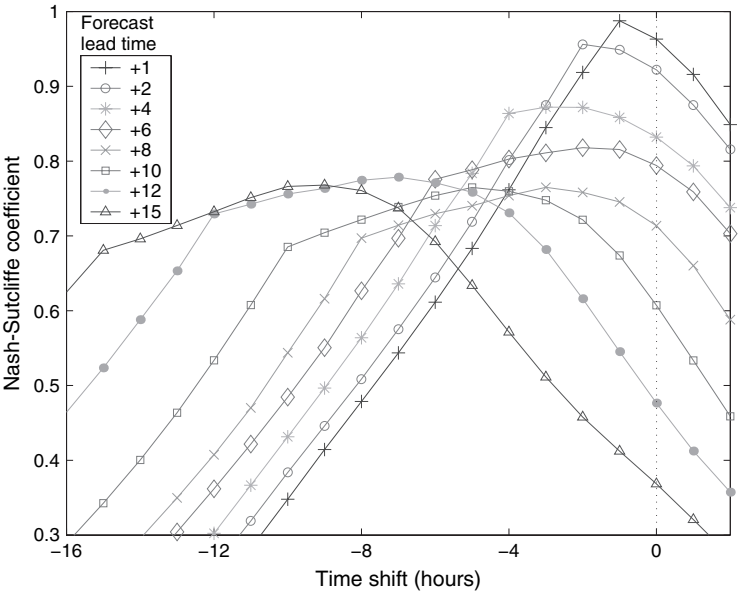


Fig. 8.4 MSE-trained ANN model performance for various lead forecast times, in terms of both Nash–Sutcliffe coefficient and timing error

(the different lines). The idea for this method of timing evaluation is taken from Conway et al. (1998). What Fig. 8.4 shows is that the prediction lag increases with the lead forecast time (i.e. the peaks are further to the left for longer lead times), but not proportionally. This can also be seen in Table 8.4, where the six-step-ahead forecast suffers not from a TE of -6, but only an average of -1. What also can be clearly observed is the dramatic decrease in CE for longer lead times, which can be read from the vertical line at a time shift of 0. The above proves that the training on MSE or CE can be inadequate and that there is much to be gained by correcting ANN models for timing errors (De Vos and Rientjes 2005).

As mentioned before, the main cause of this timing error is that previously observed values of discharge are often used as ANN model inputs, since they are considered indicators of the hydrological state. Such data, however, introduce an autoregressive model component in the ANN model, which becomes dominant when the discharge time series show high autocorrelation (see De Vos and Rientjes 2005). This causes the ANN model to produce a forecast that is very similar to the last known discharge, effectively causing timing errors in the predictions.

The results in Tables 8.3a and 8.4a for the models trained using the TE factor in the objective function show an improvement in timing only at the cost of a degradation of most other performance measures. In Tables 8.3b and 8.4b the 'best' performing ANNs based on expert judgement of all objective functions are presented. These results are promising since some trained networks with a lead time of 6 hours are capable of making correctly timed forecasts, while maintaining reasonably good

performance in terms of other statistics. Thus, it proves that for six-step-ahead forecast models, in which the influence of the last recorded observation is less than in the one-step-ahead models, it is possible to find good optima that have correct forecast timing in combination with good overall fit. The one-step-ahead forecasts seemed not to be affected by the measures to prevent timing errors.

Unfortunately, the training algorithm has more difficulty in finding optima when implementing timing in the objective function. This is probably due to the introduction of the multiplication factor TEf, which makes the gradients of the objective functions extremely irregular. The use of optimisation algorithms that do not rely on gradients (e.g. a genetic algorithm, as used by Conway et al. 1998) might alleviate this problem.

8.5 Conclusions

The ANN models for a mesoscale catchment used in this study were shown to be capable of making reasonably good forecasts of runoff values for lead times of 1 and 6 hours. The overall fit of the forecasted versus observed runoff was good, but correct timing remains a problem for ANN models, especially when previous discharge values are used as model input. This introduces an autoregressive component that can become dominant, effectively causing timing errors in the forecast (De Vos and Rientjes 2005).

The use of a timing error statistic during ANN training as a method of increasing timing accuracy of ANN rainfall-runoff model forecasts is only partly effective: the performance according to other performance measures is decreasing. There seems to be a trade-off between the objectives of correct timing and good overall fit. However, some parameter sets were found that indicate the possibility of finding an acceptable compromise between the two. Further research is therefore suggested in the field of training ANN rainfall-runoff models.

Acknowledgements The authors wish to thank the Royal Meteorological Institute of Belgium in Brussels and specifically Emmanuel Roulin for having provided the hydrometeorological data for this study. This research was partially funded through the DIOC programme 'Transient processes in Hydraulic Engineering and Geohydrology' of the Delft University of Technology.

References

Abrahart RJ, See L (2000) Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecasts in two contrasting catchments. Hydrol. Process. 14(11): 2157–2172

Anctil F, Michel C, Perrin C, Andréassian V (2004) A soil moisture index as an auxiliary ANN input for streamflow forecasting. J. Hydrol. 286: 155–167

Campolo M, Andreussi P, Soldati, A (1999) River flood forecasting with a neural network model. Water Resour. Res. 35(4): 1191–1197

Conway AJ, Macpherson KP, Brown JC (1998) Delayed time series predictions with neural networks. Neurocomputing 18: 81–89

De Vos NJ, Rientjes THM (2005) Constraints of artificial neural networks for rainfall–runoff modelling: trade-offs in hydrological state representation and model evaluation. Hydrol. Earth Syst. Sc. 9: 111–126

Dibike YB, Solomatine DP (2001) River flow forecasting using artificial neural networks. Phys. Chem. Earth (B) 26(1): 1–7

Edijatno N, Nascimento O, Yang X, Makhlouf Z, Michel C (1999) GR3J: a daily watershed model with three free parameters. Hydrol. Sci. J. 44(2): 263–277

Hogue TS, Gupta HV, Sorooshian S, Holz A, Braatz D (2000) A multistep automatic calibration scheme for river forecasting models. J. Hydrometeorol. 1: 524–542

Hsu KL, Gupta HV, Sorooshian S (1995) Artificial neural network modeling of the rainfall-runoff process. Water Resour. Res. 31(10): 2517–2530

Jain A, Srinivasulu S (2004) Development of effective and efficient rainfall–runoff models using integration of deterministic, real-coded genetic algorithms and artificial neural network techniques. Water Resour. Res. 40(4): W04302

Kitanidis PK, Bras RL (1980) Real-time forecasting with a conceptual hydrologic model, 2, applications and results. Water Resour. Res. 16(6): 1034–1044

Minns AW, Hall MJ (1996) Artificial neural networks as rainfall–runoff models. Hydrol. Sci. J. 41(3): 399–417

Nash JE, Sutcliffe JV (1970) River flow forecasting through conceptual models; part I – a discussion of principles. J. Hydrol. 10: 282–290

Perrin C, Michel C, Andréassian V (2003) Improvement of a parsimonious model for streamflow simulation. J. Hydrol. 279: 275–289

Shamseldin AY (1997) Application of a neural network technique to rainfall-runoff modelling. J. Hydrol. 199: 272–294

Smith J, Eli RN (1995) Neural-network models of rainfall-runoff process. J. Water Resour. Plng. Mgmt. 121(6): 499–508

Toth E, Brath A, Montanari A (2000) Comparison of short-term rainfall prediction models for real-time flood forecasting. J. Hydrol. 239: 132–147

Chapter 9
Data-Driven Streamflow Simulation: The Influence of Exogenous Variables and Temporal Resolution

E. Toth

Abstract Data-driven modelling approaches, like artificial neural networks, are particularly sensitive to the choice of input and output variables. This study focuses on the size of the temporal observation interval of input and output data in a river flow prediction application, analysing the simulation performances when considering increasing time aggregations of different input variables. The analyses are carried out on the data registered in a medium-sized (1,050 km^2) watershed located on the Apennine mountains, where hourly meteorological data and streamflow measurements are available over an 8-year period.

Four modelling approaches are considered for the prediction of river flow in the closure section: (1) without exogenous inputs, (2) with the additional input of past precipitation data, (3) with the additional input of past streamflow data measured in the upstream section, (4) with the additional input of both past precipitation and past upstream flow. For each modelling approach, using both (a) input data and output data at the same time scale and (b) input data at a temporal resolution finer than that of the output data, optimal modelling networks are identified and forecast performances are compared. The results highlight how the simulation improves with the addition of exogenous inputs, in particular upstream flow data, and with the use of input data at a temporal resolution finer than that of the output data. The results also show how both such benefits increase for larger temporal aggregation of the forecasts.

Keywords Streamflow modelling. neural networks. temporal resolution. input saliency

9.1 Introduction

Black-box (or *system theoretic*) models are living a period of *renaissance* in hydrological fields, due to the introduction of artificial neural networks (ANNs). The

E. Toth
DISTART – Faculty of Engineering, University of Bologna, Viale Risorgimento, 2 I-40136 Bologna, Italy, e-mail: elena.toth@unibo.it

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

appeal of the use of ANNs as black-box models lies mainly in their capability to flexibly reproduce the highly non-linear nature of the relationship between hydrological variables, and also when such relationships are not explicitly known a priori. Since the real-time framework gives more importance to the simplicity and robustness of the model implementation rather than to an accurate description of the various internal sub-processes, it is certainly worthy considering ANN models as powerful tools for real-time short-term runoff forecasts.

In hydrological applications of ANNs, some studies have been dedicated to the prediction of river flows with no exogenous inputs, that is with only the use of past flow observations measured in the same stream section where forecasts are issued (e.g. Atiya et al. 1999; Abrahart and See 2000; Nagesh Kumar et al. 2004). In these cases, the ANNs are used as univariate time series analysis techniques, forecasting the future discharge (output) on the basis of the last observed values (input). The majority of the applications for river flow prediction consists of modelling the rainfall-runoff transformation, adding precipitation observations to past flows: extremely encouraging results have been obtained in the literature on both real and synthetic rainfall-runoff data (Minns and Hall 1996; Campolo et al. 1999; Zealand et al. 1999; Abrahart et al. 2001; Dawson and Wilby 2001; Cameron et al. 2002; Hsu et al. 2002; Laio et al. 2003; Solomatine and Dulal 2003; Jain and Srinivasulu 2004; Moradkhani et al. 2004; Toth and Brath 2007). ANNs have also been successfully, although less frequently, used as flood propagation models, where the streamflow observed in upstream cross-sections (and sometimes also downstream, especially in case of tidal influence) is provided as input (Karunanithi et al. 1994; Imrie et al. 2000; Chang and Chen 2003; Deka and Chandramouli 2005).

This work presents a comparison of the performances obtainable when providing both precipitation and upstream flow data as inputs to an ANN. It is important to underline that this kind of application is possible (and easily implemented) due to the flexibility and simplicity characterising the management of input and output variables in ANNs, whereas the use of upstream flows would hardly be feasible in a traditional rainfall-runoff model.

The results of a series of ANN implementations for river flow forecasting are presented, analysing on one hand the benefit allowed by the use of different input exogenous variables and on the other hand the influence of the variation of the temporal scale of the input data. In fact, when predicting the streamflow over a future temporal interval, it is possible to provide either past data measured at the same temporal interval as input or, when available, data at a finer temporal resolution. In the first case, the model searches for a relationship between the lagged values of the same variable. This might be an advantage in the use of time series analysis techniques, like ANNs, which allow for the identification of the properties of the series to predict its future evolution. On the other hand, in the second case, there might be a gain in the use of input data of higher quality, which is at a finer resolution, describing in more detail the recent past of the hydro-meteorological forcing taking place in the river and the watershed.

Four modelling approaches are considered for the prediction of river flow at the closure section of the mountain part of the Italian Reno River watershed: (1) without

the use of exogenous input, i.e. feeding the network only with past streamflow data measured at the closure section; (2) with the additional input of past precipitation data; (3) with the additional input of past streamflow data measured in the upstream section; and (4) with the additional input of both past precipitation and past upstream flow. For each modelling approach using both (a) input data and output data at the same time scale and (b) input data at a temporal resolution finer than that of output data, optimal modelling networks are identified and forecast performances are compared.

9.2 Case Study and Data Sets

The case study herein considered is referred to the Reno River basin, located in the Apennines Mountains in north-central Italy (Fig. 9.1). The drainage area of the mountain part of the watershed, closed at the river cross-section of Casalecchio, just upstream of the city of Bologna, is 1,050 km^2 and the main stream is around 76 km long. The average elevation is 635 m above sea level, the highest peak and the outlet being at an altitude of 1,900 and 63 m above sea level, respectively.

Mountain areas cover the major part of the basin and consist primarily of soils and rocks of sedimentary origin (clay and loam), characterised by a low permeability which tends to decrease with increasing altitude, while in the terminal section there are highly permeable alluvial fans. The vegetation cover on the mountain areas consists primarily of broad-leaved woods, while in the lower part of the basin there are also farmlands and urbanised areas. Because of their low permeability and their extension with respect to the total catchment surface, mountain areas contribute remarkably to the formation of flood flows. The majority of precipitation events occur

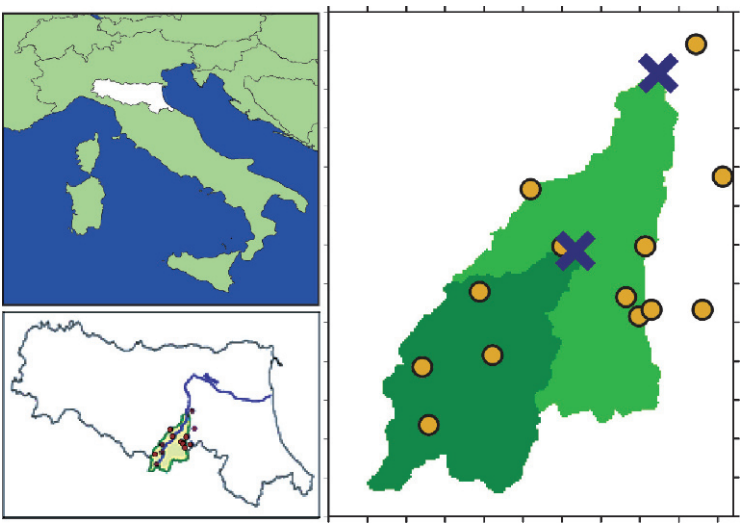


Fig. 9.1 The Emilia-Romagna region and the Reno river watershed, with the location of raingauges (o) and streamgauges (x)

from October to April, November being the wettest month, with the runoff regime closely following the precipitation trend.

The period of observation of the available data set covers 8 years from 1 January 1993 to 31 December 2000. The hydro-meteorological data set consists of hourly discharge at the closure section of Casalecchio and at the internal section of Vergato (where the drainage area is 497 km^2), and the hourly mean precipitation values obtained from precipitation depths measured at 13 rain gauges, even if not all are simultaneously operative for the whole observation period.

In the applications presented in Section 9.3, the entire hydrological year 1994–1995 (from 1 September 1994 to 31 August 1995, for a total of 8,760 hourly observations) that includes the flood event with the highest peak (1,507 m^3/s) in the data set is used in the calibration procedure. The subsequent continuous record (from 1 September 1995 to 31 December 2000) is used for validation purposes: the validation data include the second highest peak (1,274 m^3/s) and consist of a total of 46,776 hourly observations. The calibration period of one hydrological year (beginning at the end of the dry season) and the inclusion of the maximum flood event should allow the model to learn the dominant features of the entire hydrological cycle during both dry and wet seasons and also the conditions leading to extreme events. It should be underlined that a validation period of such length (more that 5 years of hourly data, for a calibration period of only 1 year) is certainly an exacting test for the modelling approaches.

The spatial average of hourly rainfall depths, P, was estimated with an inverse squared distance weighting of the rain gauge observations.

Hourly precipitation and streamflow data were successively aggregated over a 3- and 6-h time span, respectively, in order to test the performances of ANNs for predicting streamflow at varying temporal scales as described in the following sections.

9.3 Artificial Neural Networks for Streamflow Prediction

Artificial neural networks (ANNs) distribute computations to processing units called neurons, grouped in layers and are densely interconnected. Three different layer types can be distinguished: an input layer, connecting the input information to the network (and not carrying out any computation), one or more hidden layers, acting as intermediate computational layers, and an output layer, producing the final output. In correspondence of a computational node, each one of the entering values is multiplied by a connection weight. Such products are then all summed with a neuron-specific parameter, called the bias, which is used to scale the sum of products into a useful range. The computational node finally applies an activation function to the above sum producing the node output. Weights and biases are determined by means of a non-linear optimisation procedure (training) that aims at minimising a learning function expressing a closeness between observations and ANN outputs, in the present case the mean squared error. A set of observed input and output data pairs (called a target to be distinguished from the network final output), i.e. the training data set, is processed repeatedly, changing the parameters until they converge to

values such that each input vector produces outputs as close as possible to the desired target vectors.

The following network characteristics were chosen for all the ANN applications described in the paper:

- Architecture: multi-layer feedforward networks formed by only one hidden layer (according to the "Universal Approximation Theorem", see Hornik et al. 1989);
- Training algorithm: the quasi-Newton Levenberg–Marquardt backpropagation algorithm (Hagan and Menhaj 1994), minimising the mean squared error learning function; in order to prevent the training algorithm being trapped in a local minimum, each ANN is trained starting from 10 different initial networks, randomly initialised, of which the best-performing network on the training data set is chosen as the *trained network;*
- Activation functions: a tan-sigmoidal unit was chosen for the hidden layer:

$$f(x) = \frac{2}{(1+e^{-2x})} - 1 \tag{9.1}$$

where x is the input to the node, i.e. the weighted sum of the outputs from previous nodes and the bias of the node, and $f(x)$ is the node output. A linear transfer function was instead chosen for the output layer: it was, in fact, preferred to choose an output activation function suited to the original distribution of targets, that in the present case are unbounded, rather than to force the data, with a standardisation or rescaling procedure, to conform to the output activation function.

As far as the number of input and hidden nodes is concerned, the investigation of the performances of several combinations of input and hidden layer dimensions will be described in the following sections.

In the present work, a total of 24 ANNs were designed, with differing input variables and at different time scales. Each neural network output consists of a continuous series of one-step-ahead streamflow predictions at 1-, 3- or 6-h time scales in the closure section of the case study watershed (Casalecchio).

To test the value of different hydro-meteorological knowledge that may be provided as input to predicting the river flow, four kinds of models were considered:

ANNs are first used without exogenous input. Only the last discharge values measured in the closure section are provided as inputs to the networks, thus testing a univariate time series analysis technique. Analysing the performance of the forecasts provided for the validation set, the optimal number of inputs may be identified, i.e. the number of past discharge observations that seem to mainly influence future occurrence.

The optimal number of past discharge values identified in the application without exogenous input is then provided as input to the ANNs, along with exogenous input of a different nature, namely:

- past rainfall values, thus testing a rainfall-runoff approach; the introduction of other input variables in the rainfall-runoff transformation, like temperature or evapotranspiration estimates or the period of the year, was initially considered,

but the results indicated no improvement in comparison with the use of precipitation data alone;
- past discharge values measured in the inner, upstream river section, thus testing a flood propagation model;
- both past rainfall and past upstream flow.

To analyse the influence of the time resolution of the input data, the four modelling approaches described above were applied using

(a) input data and output data at the same time scale (1-, 3- and 6-h time steps)
(b) input data at a temporal resolution finer than that of the output data (input variables at a 1-h time scale for 3-h output and either 1- or 3-h input variables for a 6-h output).

As with the large majority of rainfall-runoff models based on ANNs, no application is designed in which the past flow observations are not included as inputs. In fact, the response time of the river depends on the state of saturation of the basin, which is a function of the history of the hydrological processes in the antecedent period. Since the model is not run in a continuous way, where the state of the catchment may be represented by the moisture content in various water retention stores, the only information available on the condition of the basin before the forecast instant, and therefore on the capability of the system to respond to the current rainfall perturbation, is the ongoing runoff in the forecasting section (see Campolo et al. 1999).

9.3.1 Implementation of ANNs with No Exogenous Input

9.3.1.1 Input and Output at the Same Time Resolution

In the first experiment, three neural networks were designed to produce river flow predictions at, respectively, 1-h (Q_t^{1h}), 3-h (Q_t^{3h}) and 6-h aggregations (Q_t^{6h}), providing a number of past flows as inputs, measured at the same time step, varying from 1 to 8, therefore corresponding to networks with 1–8 input nodes and one output node. For each number of input nodes, networks were tested with a number of hidden nodes varying from 1 to 12. The performances of the obtained flow predictions on the validation set were evaluated in terms of the Nash–Sutcliffe efficiency coefficient (Nash and Sutcliffe 1970):

$$\mathrm{NS} = 1 - \frac{\sum (Q_t - Q_t^*)^2}{\sum (Q_t - Q_\mathrm{m})^2} \tag{9.2}$$

where $Q_t^*$is the streamflow forecast at time t, Q_t is the value of the corresponding observed flow and Q_m is the mean observed value.

The comparison of the efficiency coefficients for the validation data (shown for all the best identified architectures in Table 9.1) indicated that the best predictions are obtained with a small number of past streamflow values in input, NI Q, i.e. two past values (Q_{t-1}^{1h},Q_{t-2}^{1h}) for 1-h data and one past value (Q_{t-1}^{3h} and Q_{t-1}^{6h}) for 3- and 6-h time steps.

Table 9.1 Networks with no exogenous input

Out	In	Out/in resolution	NI Q	NH	NS eff
Q_t^{1h}	$Q_{t-1,\ldots,t-NIQ}^{1h}$	Out: $\Delta t = 1$h, In: $\Delta t = 1$h	2	6	0.981
Q_t^{3h}	$Q_{t-1,\ldots,t-NIQ}^{3h}$	Out: $\Delta t = 3$h, In: $\Delta t = 3$h	1	2	0.850
Q_t^{6h}	$Q_{t-1,\ldots,t-NIQ}^{6h}$	Out: $\Delta t = 6$h, In: $\Delta t = 6$h	1	2	0.684
Q_t^{3h}	$Q_{t-1,\ldots,t-NIQ}^{1h}$	Out: $\Delta t = 3$h, In: $\Delta t = 1$h	3	10	0.931
Q_t^{6h}	$Q_{t-1,\ldots,t-NIQ}^{1h}$	Out: $\Delta t = 6$h, In: $\Delta t = 1$h	1	2	0.835
Q_t^{6h}	$Q_{t-1,\ldots,t-NIQ}^{3h}$	Out: $\Delta t = 6$h, In: $\Delta t = 3$h	1	2	0.782

Configuration and NS coefficients of the best forecasting networks for the validation set: output (Out) and input (In) variables and corresponding temporal resolutions, number of input (NI Q) and hidden (NH) nodes, NS efficiency coefficient.

9.3.1.2 Inputs at a Finer Resolution than the Output

In the second experiment, future streamflow at 3- and 6-h time steps is modelled feeding the network with past streamflow values observed at a temporal resolution finer than that of the output data, i.e. past 1-h flow values for predicting 3-h flows and past 1- and 3-h values for predicting 6-h flows. Networks with 1–8 input nodes and 1–12 hidden nodes were trained on the calibration data and used for simulation of the validation data.

Table 9.1 shows also the configuration of the networks that, with these input variables, allowed the highest efficiency coefficients for the validation period.

In the applications described in the following sections, the optimal number of past flow observations in the forecast stream section identified in the applications described in the present section is provided as inputs to the ANN, along with exogenous input data, assuming that such lag corresponds to the number of past values that mainly influence future occurrence.

9.3.2 Implementation of ANNs with Precipitation Input

First, both past streamflow (Q) and past areal precipitation (P) data are given as inputs to the networks at the same temporal resolution as the modelled output. The number of past precipitation values was varied from 1 to 10, while the number of past flow values is the one identified in Sect. 9.3.1, i.e. two past flows for Q_t^{1h}, and one past flow for Q_t^{3h} and Q_t^{6h}. For each combination of inputs, networks were tested with a number of hidden nodes varying from 1 to 12. Table 9.2 shows the configuration and the performance of the networks that, with the addition of a varying number of past precipitation data (and a varying hidden layer dimension), produced the highest efficiency coefficients.

Second, the past streamflow and past precipitation data are given as inputs to the networks at a temporal resolution finer than that of the modelled output. Past precipitation values varying from 1 to 10, along with the number of past flow values identified in Sect. 9.3.1 (Table 9.1), are provided as inputs. Table 9.2 shows the configurations that produced the highest efficiency coefficient with such inputs.

Table 9.2 Networks with precipitation input

Out	In		NI Q	NI P	NH	NS eff
Q_t^{1h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{1h}$	$P_{t-1,\ldots,t-\mathrm{NIP}}^{1h}$	2	4	2	0.981
Q_t^{3h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{3h}$	$P_{t-1,\ldots,t-\mathrm{NIP}}^{3h}$	1	2	2	0.878
Q_t^{6h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{6h}$	$P_{t-1,\ldots,t-\mathrm{NIP}}^{6h}$	1	3	2	0.821
Q_t^{3h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{1h}$	$P_{t-1,\ldots,t-\mathrm{NIP}}^{1h}$	3	4	6	0.940
Q_t^{6h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{1h}$	$P_{t-1,\ldots,t-\mathrm{NIP}}^{1h}$	1	8	4	0.877
Q_t^{6h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{3h}$	$P_{t-1,\ldots,t-\mathrm{NIQ}}^{3h}$	1	2	2	0.848

Configuration and NS coefficients of the best forecasting networks for the validation set: output (Out) and input (In) variables, number of past streamflow (NI Q) and precipitation (NI P) input nodes, number of hidden nodes (NH), NS efficiency coefficient.

9.3.3 Implementation of ANNs with Upstream Flow Input

Table 9.3 shows the results obtained with the best-performing architectures when feeding the networks with both downstream (Q) and upstream (QU) flow data, providing a number of past upstream flow values as inputs varying from 1 to 10, along with the number of past flows measured at the closure section already described. The input vectors are always formed by upstream and downstream flow data at the same temporal aggregation: first the same and then finer than the resolution of the target predictions.

Table 9.3 Networks with upstream flow input

Out	In		NI Q	NI QU	NH	NS eff
Q_t^{1}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{1h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{1h}$	2	2	4	0.988
Q_t^{3h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{3h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{3h}$	1	8	6	0.960
Q_t^{6h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{6h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{6h}$	1	8	6	0.852
Q_t^{3h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{1h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{1h}$	3	6	4	0.978
Q_t^{6h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{1h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{1h}$	1	10	6	0.950
Q_t^{6h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{3h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{3h}$	1	3	6	0.921

Configuration and NS coefficients of the best forecasting networks for the validation set: output (Out) and input (In) variables, number of past streamflow (NI Q) and upstream flow (NI QU) input nodes, number of hidden nodes (NH), NS efficiency coefficient.

9.3.4 Implementation of ANNs with Both Precipitation and Upstream Flow Input

Finally, the architectures and the performance of the ANNs that consider all the available hydro-meteorological variables as inputs, i.e. past downstream flows (Q), past precipitation (P) and past upstream flows (QU), are provided in Table 9.4.

Table 9.4 Networks with precipitation and upstream flow input

Out	In			NI Q	NI P	NI QU	NH	NS eff
Q_t^{1h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{1h}$	$P_{t-1,\ldots,t-\mathrm{NIP}}^{1h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{1h}$	2	4	6	4	0.988
Q_t^{3h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{3h}$	$P_{t-1,\ldots,t-\mathrm{NIP}}^{3h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{3h}$	1	2	6	2	0.964
Q_t^{6h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{6h}$	$P_{t-1,\ldots,t-\mathrm{NIP}}^{6h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{6h}$	1	3	2	8	0.869
Q_t^{3h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{1h}$	$P_{t-1,\ldots,t-\mathrm{NIP}}^{1h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{1h}$	3	4	2	8	0.975
Q_t^{6h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{1h}$	$P_{t-1,\ldots,t-\mathrm{NIP}}^{1h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{1h}$	1	8	10	6	0.964
Q_t^{6h}	$Q_{t-1,\ldots,t-\mathrm{NIQ}}^{3h}$	$P_{t-1,\ldots,t-\mathrm{NIQ}}^{3h}$	$\mathrm{QU}_{t-1,\ldots,t-\mathrm{NIQU}}^{3h}$	1	2	6	2	0.950

Configuration and NS coefficients of the best forecasting networks for the validation set: output (Out) and input (In) variables, number of past streamflow (NI Q), precipitation (NI P) and upstream flow (NI QU) input nodes, number of hidden nodes (NH), NS efficiency coefficient.

A number of past upstream flow values varying from 1 to 10, along with the number of past flow and precipitation values identified in Sect. 9.3.2 (Table 9.2), were tested as inputs.

9.4 Results and Discussion

9.4.1 The Effects of Adding Exogenous Inputs

Figure 9.2 shows the comparison of the efficiency coefficients obtained with and without the exogenous variables, for a temporal resolution of the input data both (a) equal to and (b) higher than that of the output data.

The value of the hydro-meteorological knowledge contained in precipitation and upstream flow measures is manifest, for either input resolutions equal to or higher

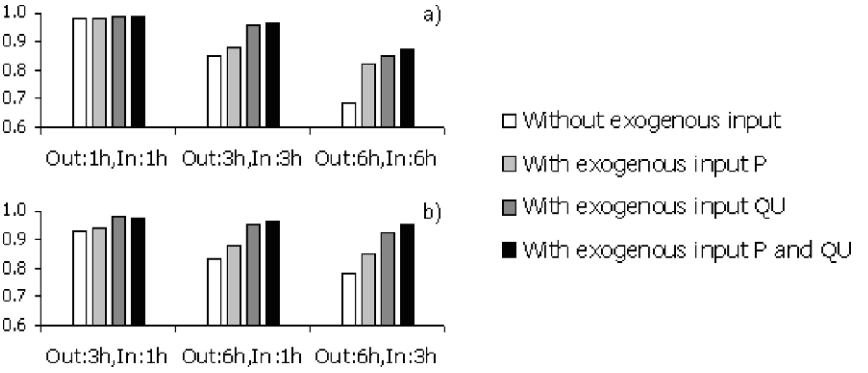


Fig. 9.2 Nash–Sutcliffe coefficients for the validation forecasts, for temporal resolution of input data both (**a**) equal to and (**b**) higher than that of the output data, with and without exogenous inputs

than that of the forecasted variable. When predicting streamflow at an hourly time step, the gain observed by the addition of exogenous inputs is minor, whereas such knowledge becomes crucial for larger temporal aggregations of the output variables.

The forecast performances are better when adding streamflow from the upstream section as input than when providing to the network the precipitation data alone, indicating that the information on past upstream flows is more salient, at least as far the examined outputs are concerned, than that of past precipitation data. The additional gain obtained when using both upstream and precipitation measurements is marginal in comparison to the use of upstream data alone.

9.4.2 The Effects of Altering the Temporal Resolution of the Input Data

When comparing the efficiency coefficients obtained when forecasting future 3- and 6-h streamflow values in Fig. 9.3, it is evident that, for any choice of endogenous and exogenous variables, a remarkable improvement is observed when using input data at a finer resolution.

It may be observed that the benefits of using input data at temporal resolutions finer than the output data are larger with no exogenous input and it grows for increasing output temporal aggregation.

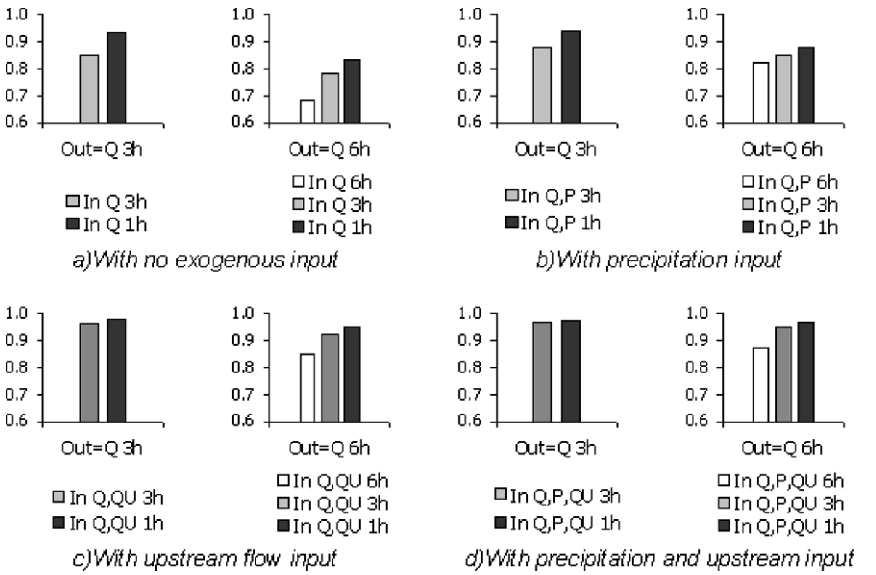


Fig. 9.3 Nash–Sutcliffe coefficients for the validation forecasts of 3-h (Out= Q 3h) and 6-h flows (Out=Q 6h), for temporal resolution of input data equal and higher than that of output data (**a**) with no exogenous input, In Q, and with the addition of inputs of, respectively, (**b**) past precipitation, In Q, P, (**c**) past upstream flows, In Q, QU and (**d**) both precipitation and upstream flows, In Q, P, QU

One may question whether the improvement gained by using input data at a finer temporal resolution is due solely to the persistence property of the streamflow series: in fact, the last observation measured with a shorter time step is temporally nearer to the forecast instant. It is therefore interesting analysing the performances obtainable by setting the forecast equal to the last (more updated) streamflow observation at the finer time step. Three additional models are thus implemented, where future streamflow at a larger time step is set equal to the last observation measured at finer resolution, that is

- Persistent A) $Q_t^{3h} = Q_{t-1}^{1h}$
- Persistent B) $Q_t^{6h} = Q_{t-1}^{1h}$
- Persistent C) $Q_t^{6h} = Q_{t-1}^{3h}$.

Each of the persistent models is compared with the best-performing ANN having as inputs the past flows measured at the corresponding finer time step, that is 1 h for Persistent A and B and 3 h for Persistent C (see Fig. 9.4).

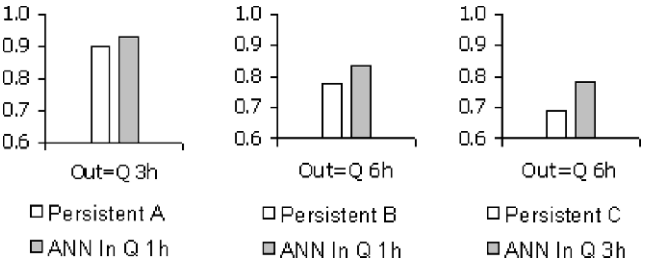


Fig. 9.4 Nash–Sutcliffe coefficients for the validation forecasts of 3- and 6-h flows obtained by Persistent models and by ANNs with temporal resolution of input data finer than that of the output data, with no exogenous input

It can be seen that the predictions resulting from ANNs are always consistently better than the persistent models, especially for 6-h streamflow predictions. The comparison is presented for ANNs without exogenous input only, since persistent models are even largely outperformed by ANNs with exogenous input, which always provide better results in comparison with the use of past streamflow values alone.

9.5 Conclusions

This chapter analyses the value of the inclusion of different hydro-meteorological variables as inputs to ANNs providing streamflow prediction at the closure section of a mid-sized case study watershed. The experiments consider the use of mean areal precipitation data over the watershed and of flow measures in an upstream river section in addition to the endogenous input (streamflow measures in the closure

section). They also examine the influence of the observation time scale of the input variables.

The results highlight the significance of the hydro-meteorological knowledge contained in precipitation and upstream flow measures, for input resolution either equal to or higher than that of the forecasted variable. The gain allowed by the addition of exogenous input increases for larger temporal aggregation of the output variables, and the information on past upstream flows seems more salient than that of past precipitation data.

As far as the resolution of input data is concerned, it may be observed that a remarkable benefit is achieved when using input data at a temporal resolution finer than the output data. Such benefits are larger when no exogenous data are provided as inputs to the networks and they grow for increasing temporal aggregation of the predicted streamflow.

References

Abrahart RJ, See L (2000) Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecasts in two contrasting catchments. Hydrological Processes 14(11):2157–2172

Abrahart RJ, See L, Kneale PE (2001) Investigating the role of saliency analysis with a neural network rainfall-runoff model. Computers & Geosciences 27:921–928

Atiya AF, El-Shoura SM, Shaheen SI, El-Sherif MS (1999) A comparison between neural-network forecasting techniques – Case study: River flow forecasting. IEEE Transactions on neural networks 10(2):402–409

Cameron D, Kneale P, See L (2002) An evaluation of a traditional and a neural net modelling approach for flood forecasting for an upland catchment. Hydrological Processes 16(5):1033–1046

Campolo M, Andreussi P, Soldati A (1999) River flood forecasting with a neural network model. Water Resources Research 35(4):1191–1197

Chang F, Chen Y (2003) Estuary water-stage forecasting by using radial basis function neural network. Journal of Hydrology 270:158–166

Dawson CW, Wilby RL (2001) Hydrological modelling using artificial neural networks. Progress in Physical Geography 25(1):80–108

Deka P, Chandramouli V (2005) Fuzzy neural network model for hydrologic flow routing. Journal of Hydrologic Engineering 10(4):302–314

Hagan MT, Menhaj M (1994) Training feedforward networks with the Marquardt algorithm. IEEE Transactions on Neural Networks 5(6):989–993

Hornik K, Stinchcombe M, White H (1989) Multilayer feedforward networks are universal approximators. Neural Networks 2:359–366

Hsu K, Gupta HV, Gao X, Sorooshian S, Imam B (2002) Self-organizing linear output map (SOLO): An artificial neural network suitable for hydrologic modeling and analysis. Water Resources Research 38(12) doi:10.1029/2001WR000795

Imrie CE, Durucan S, Korre A (2000) River flow prediction using artificial neural networks: generalisation beyond the calibration range. Journal of Hydrology 233:138–153

Jain A, Srinivasulu S (2004) Development of effective and efficient rainfall-runoff models using integration of deterministic, real-coded genetic algorithms and artificial neural network techniques. Wat Resources Research 40 doi:10.1029/2003WR002355.

Karunanithi N, Grenney WJ, Whitley D, Bovee K (1994) Neural networks for river flow prediction. Journal of Computing in Civil Engineering 8(2):201–220

Laio F, Porporato A, Revelli R, Ridolfi L (2003) A comparison of nonlinear flood forecasting methods. Wat Resources Research 39(5) doi:10.1029/2002WR001551

Minns AW, Hall MJ (1996) Artificial neural networks as rainfall-runoff models. Hydrological Science Journal 41(3):399–417

Moradkhani H, Hsu K, Gupta HV, Sorooshian S (2004) Improved streamflow forecasting using self-organizing radial basis function artificial neural networks. Journal of Hydrology 295:246–262

Nash JE, Sutcliffe JV (1970) River flow forecasting through conceptual models. Part 1. A discussion of principles. Journal of Hydrology 10:282–290

Nagesh Kumar D, Srinivasa Raju K, Tathish T (2004) River flow forecasting using recurrent neural networks. Water Resources Management 18:143–161

Solomatine DP, Dulal KN (2003) Model trees as an alternative to neural networks in rainfall–runoff modelling. Hydrological Sciences Journal 48(3):399–411

Toth E, Brath A (2007) Multistep ahead streamflow forecasting: Role of calibration data in conceptual and neural network modeling. Water Resources Research 43, W11405, doi:10.1029/2006WR005383.

Zealand CM, Burn DH, Simonovic SP (1999) Short term streamflow forecasting using artificial neural networks. Journal of Hydrology 214:32–48

Chapter 10
Groundwater Table Estimation Using MODFLOW and Artificial Neural Networks

K. Mohammadi

Abstract The use of numerical models to simulate groundwater flow has been addressed in many research studies during the past decade. The main drawback with these models is their enormous and generally difficult or costly data requirements. On the other hand, artificial neural networks (ANNs) are offering a simple but precise solution to many simulation problems. In this chapter, the applicability of ANN models in simulating groundwater levels has been investigated. In order to be able to use ANN models for aquifers with limited data, MODFLOW was used to simulate the groundwater flow and the calibrated model was then applied to generate hundreds of data sets for the training of the ANN model. Another purpose of this chapter is to identify ANN models that can capture the complex dynamics of water table fluctuations, even with relatively short lengths of training data. MODFLOW outputs and measured water table elevations were used to compare the performance of the ANN models. The average regression coefficients for multi-layer perceptrons and time lag recurrent neural networks were 0.865 and 0.958, respectively.

Keywords Groundwater modeling · artificial neural network · MODFLOW

10.1 Introduction

Groundwater models provide a scientific and predictive tool for determining appropriate solutions to water allocation, surface water–groundwater interaction, landscape management, or impact of new development scenarios. For many practical problems of groundwater hydrology, such as aquifer development, contaminated aquifer remediation, or performance assessment of planned water supply projects, it is necessary to predict the water table and its fluctuations during the year. The use of numerical models to simulate groundwater flow has been addressed in many

K. Mohammadi
Associate Professor, Tarbiat Modares University, Tehran, Iran,
e-mail: kouroshm@modares.ac.ir

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

research studies over the past few decades. The main drawback with these models is their enormous and generally difficult or costly data requirements. On the other hand, artificial neural networks (ANNs) are offering a simple but precise solution to many simulation problems. ANNs are computational modeling tools that have recently emerged and found extensive acceptance in many disciplines for modeling complex real-world problems. ANNs may be defined as structures comprised of densely interconnected adaptive simple processing elements (called artificial neurons or nodes) that are capable of performing massively parallel computations for data processing and knowledge representation (Basheer and Hajmeer, 2000).

Neurocomputing does not require algorithm or rule development, a feature that often significantly reduces the quantity of software that must be developed. Consequently, the last decade has seen an uncharacteristically rapid growth of applications in virtually all disciplines, including sensor processing, pattern recognition, data analysis, and civil engineering. The development of the ANN has provided a powerful tool for nonlinear approximations. It offers a black-box approach in which a multi-layered ANN with enough neurons can approximate almost any nonlinear input–output mapping at any required accuracy. The ANN develops a solution by training on examples given to it. An ANN learns to solve a problem by developing a memory capable of associating a large number of input patterns with a resulting set of outputs or effects (Hsu et al., 1995). The only problem with these models is their dependency on data for training. Training is the process of updating the internal representation of the ANN model in response to external stimuli so that the ANN can perform a specific task. This includes modifying the network architecture, which involves adjusting the weights of the links, pruning or creating some connection links, and/or changing the firing rules of the individual neurons (Schalkoff, 1997). ANN learning is performed iteratively as the network is presented with training examples, similar to the way we learn from experience.

In the aquifer system modeling context, the ANN approach was first used to provide maps of conductivity or transmissivity values (Rizzo and Dougherty, 1994; Ranjithan et al., 1993) and to predict water retention curves of sandy soils (Schaap and Bouten, 1996). More recently, ANNs have been applied to perform inverse groundwater modeling for the estimation of different parameters (Morshed and Kaluarachchi, 1998; Lebron et al., 1999). Coulibaly et al. (2001) applied ANNs to predict water table depth fluctuations using past input delay neural networks (IDNN), recurrent neural networks (RNN), and radial basis function (RBF) neural networks. They concluded that RBF neural networks could not simulate depth to water table very well, but the other two methods produced reasonable results.

MODFLOW (McDonald and Harbaugh, 1989) is one of the most popular groundwater modeling programs in existence. Some reasons for this popularity may be (1) the program is applicable to most types of groundwater modeling problems; (2) the original packages in the program are well structured and documented; (3) the source code is in the public domain and thus can be checked for errors and modified by anyone with the necessary mathematical and programming skills; (4) the program is accepted by regulatory agencies and in litigation; and (5) ongoing modifications of the program continue to increase its capabilities (Winston, 1999).

In this chapter, the applicability of ANN models in simulating groundwater level has been investigated. In order to be able to use an ANN model for aquifers with limited data, MODFLOW was used to simulate the groundwater flow and then the calibrated model was applied to generate hundreds of data points for the ANN model. Using the generated data, the ANN model was trained. Another purpose of this chapter is to identify ANN models that can capture the complex dynamics of water table fluctuations, even with relatively short lengths of training data. MODFLOW outputs and measured water table elevations were used to compare the performance of the ANN model.

10.2 Materials and Methods

10.2.1 Study Area

Chamchamal plain with an area of 195.46 km^2 located in the west of Iran is selected for this study. Chamchamal plain is surrounded by a karstic formation, where the recharge is a major source of groundwater, as well as rainfall, infiltration from two major rivers in the area, and return flow from irrigation. The average annual rainfall is 598.25 mm. Gamasiab and Dinavar rivers are flowing east–west and north–south with an average of 32.8 and 11 cubic meters per second, respectively. Return flow from irrigation is about 40.16 million cubic meters per year which is calculated from a balance equation. The bedrock is mostly karst formation. There are 10 observation wells in the area where depth to the water table is measured monthly. Chamchamal has an unconfined aquifer with transmissivity between 230 and 3000 m^2 per day and with average specific yield about 3.5%. The main discharge parameters in the area are withdrawal wells, evaporation, and drainage canals.

10.2.2 MODFLOW Simulation Model

A grid consisting of 276 rectangles with dimensions of 0.5×0.5, 1×1, and 0.5×1 km was selected for the study area. Figure 10.1 shows the aquifer boundary, model grids, and the location of observation wells. The center of the cell was the representative point of each cell, and all data and information were discretized at these points. The data were interpolated into the desired cells using ArcGIS 8.1, and PMWIN 5.3 was used to simulate the water table fluctuations.

MODFLOW was calibrated on 1 year of available data. The average values of input data for September 1986 were used to calibrate the model in the steady-state condition. The resulting calibrated values were then used as initial conditions for the unsteady state model for the period of 1 year.

From 1986 to 1998, water surface elevations were measured monthly but there was no information regarding recharges and discharges. Data and information in the

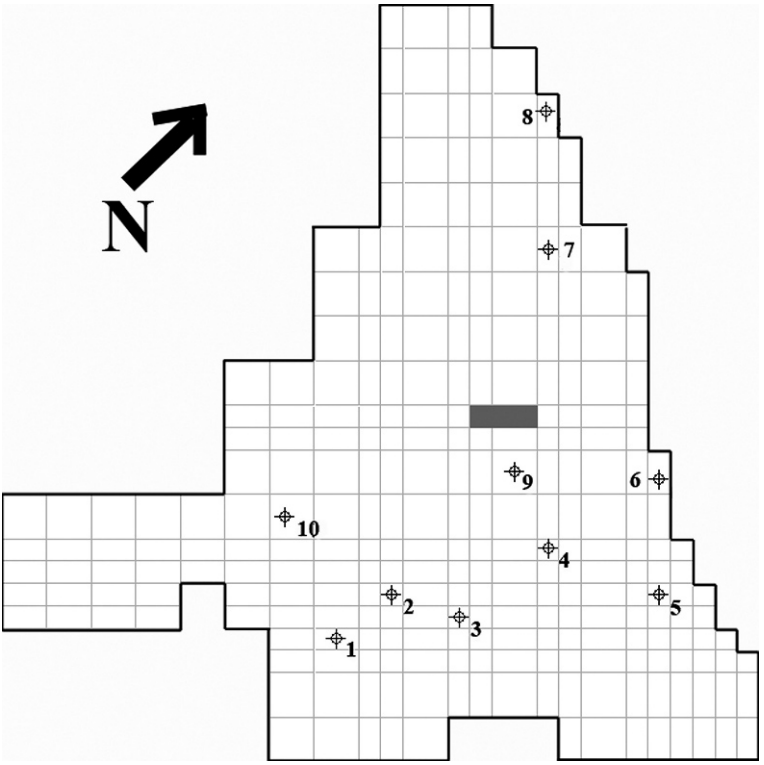


Fig. 10.1 Chamchamal aquifer and location of observation wells

year 1998 were used to verify the calibrated model. Figure 10.2 shows the comparison between observed and calculated water surface levels (WSL) in September 1986. In addition, Fig. 10.3 shows a comparison between observed and calculated WSL in one of the observation wells in the year 1998 as an example.

The calibrated model was run 144 times to produce the necessary input and output data for the ANN model. The input data were precipitation, river flow, irrigation, return flow from wells, well discharge, evaporation, recharge from karstic bedrock, drainage, underground water inflow and outflow, and the output was water table elevation in observation wells. Table 10.1 shows the different combinations of the input data for different ANN models.

10.2.3 ANNs: An Overview

In general, relationships between precipitation, the nearby surface water, well discharges, evaporation, and return flow from irrigation, etc., are likely to be nonlinear rather than linear. However, owing to the difficulties of identifying nonlinear model structure and estimating the associated parameters, only very few nonlinear

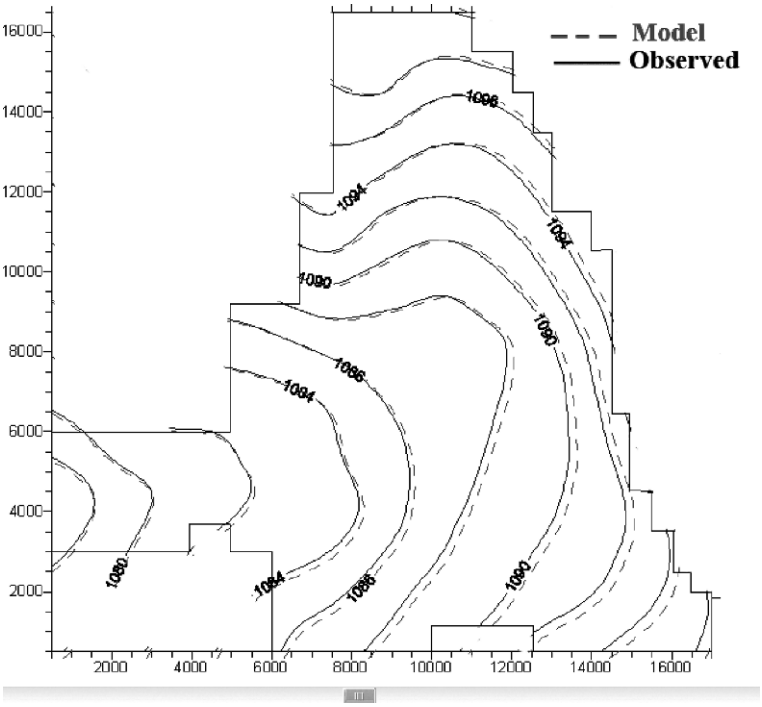


Fig. 10.2 Comparison between observed and calculated water surface level using PMWIN in September 1998

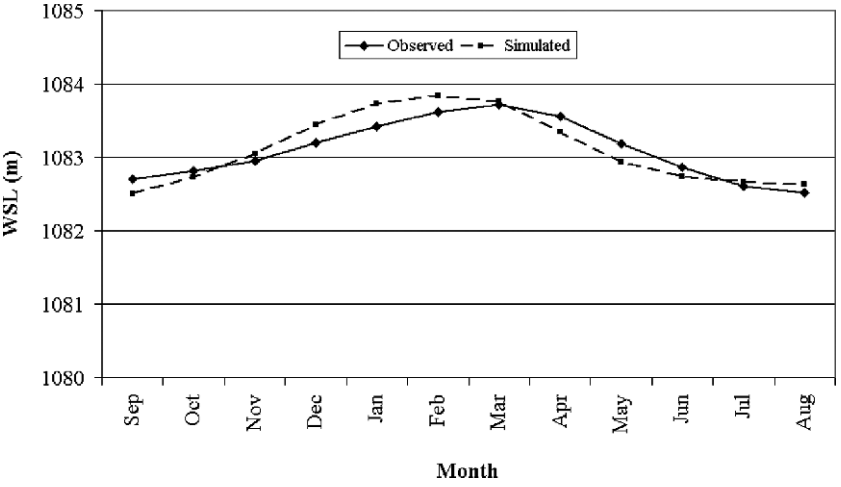


Fig. 10.3 Comparison between observed and MODFLOW simulated water surface level (WSL) in well no. 1 during the verification period

Table 10.1 Input data combination for different ANN models

Input parameter	Model number						
	1	2	3	4	5	6	7
Precipitation	+	+	+	+	+	+	+
Rivers flow	+	+	+	+	+	+	
Irrigation	+	+	+	+	+	+	+
Return flow from wells	+	+	+	+			
Recharge from bedrock	+		+				
Evaporation	+	+	+	+	+	+	+
Drainage	+	+	+	+	+		
Underground inflow	+	+					
Underground outflow	+	+					
Well discharge	+	+	+	+	+	+	+

empirical models, such as stochastic differential equations and threshold autoregressive self-exciting open-loop models, have been recently proposed for shallow water table modeling (Coulibaly et al., 2001). Therefore, a dynamic predictive model that can cope with a persistent trend and the time-varying behavior of a semiarid aquifer system is still very desirable for improving water resource management and reliable water supply planning.

Recent literature reviews reveal that artificial neural networks, specifically feedforward networks, have been successfully used for water resource modeling and prediction (Coulibaly et al., 2001; Maier and Dandy, 2000). In this study, two ANN approaches namely a multi-layer perceptron (MLP) trained with backpropagation (BP) and time lag recurrent neural networks (TLRN) will be presented for the Chamchamal aquifer.

10.2.3.1 A Multi-layer Perceptron (MLP) Trained with Backpropagation

A MLP uses a set of input and output patterns. An input pattern is used by the system to produce an output which then is compared with the target output. If there is no difference, then no learning takes place. Otherwise the difference "propagates" backwards through the network and the weights are changed to reduce the difference. The objective is to minimize the overall network error for all input patterns in the training set.

10.2.3.2 Time-Lagged Recurrent Neural Networks (TLRN)

The goal in this type of ANN is to forecast a multivariate time series using past values and available covariates. The first obvious approach to modeling these data is to fit standard statistical models (i.e., multivariate ARMA models, Kalman filtering, vector autoregression, smoothing). All of these models, however, assume regular dependence structures, which are locally smooth and linear.

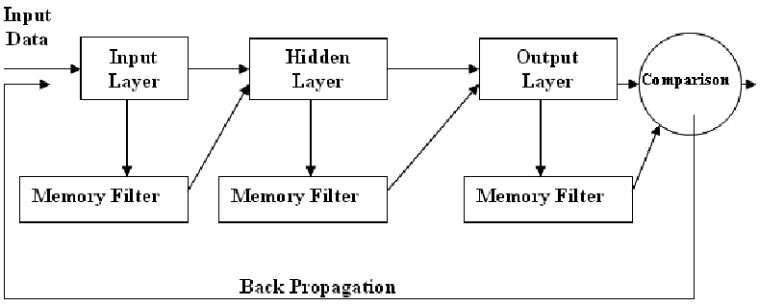


Fig. 10.4 Structure of a TLRN model

For groundwater level fluctuation data, the structural relationships and the dependence structures are likely to be nonlinear. In a regular approach, static feedforward ANNs were employed to model nonlinear relationships in water table level forecasting. The approach in TLRNs differs in that the temporal nature of the data is taken into account. In TLRNs, one adds cycles to the feedforward network structure to produce ANNs with a sort of memory.

The most studied TLRN network is the gamma model (Lefebvre and Principe, 1998). The gamma model is characterized by a memory structure that is a cascade of leaky integrators, i.e., an extension of the context layer of the Jordan and Elman nets (Fig. 10.4). The signal at the taps of the gamma memory can be represented by

$$x_0(n) = u(n) \tag{10.1}$$

$$x_k(n) = (1-\mu)\,x_k(n-1) + \mu x_{k-1}(n-1) \quad k = 1,\ldots,K \tag{10.2}$$

where x_k is the data point, n is number of signals, and k is the memory depth. Note that the signal at tap k is a smoothed version of the input, which holds the value of a past event, creating a memory. The point in time where the response has a peak is approximately given by k/μ, where μ is the feedback parameter. This means that the neural network can control the depth of the memory by changing the value of the feedback parameter, instead of changing the number of inputs. The parameter μ can be adapted using gradient descent procedures just like the other parameters in the neural network. However, since this parameter is recursive, a more powerful learning rule needs to be applied. A backpropagation through time (BPTT) algorithm was used to do this adaptation. For more information regarding all algorithms used in Table 10.2, see Lefebvre and Principe (1998).

10.2.4 Methodology

In this study, MATLAB 5.3 and NeuroSolution softwares were used to develop the multi-layer perceptron (MLP) and TLRN models, respectively. The proposed

Table 10.2 Multi-layer perceptron networks for simulating observation wells individually

Well no.	ANN model	Epoch	Topology	Learning rule	Transfer function		MSE	NMSE	R^2
1	MLP	1000	7-4-4-1	Momentum	Tanh	Learning	0.010	0.042	0.956
						Testing	0.174	0.503	0.874
2	MLP	276	8-5-1	Momentum	Tanh	Learning	0.0099	0.038	0.960
						Testing	0.042	0.121	0.922
3	MLP	1000	7-4-4-1	Delta Bar Delta	Tanh	Learning	0.020	0.100	0.898
						Testing	0.196	0.857	0.537
4	MLP	94	7-4-1	Momentum	Tanh	Learning	0.010	0.040	0.960
						Testing	0.028	0.092	0.943
5	MLP	1000	6-8-7-1	Conjugate Gradient	Tanh	Learning	0.021	0.131	0.868
						Testing	0.160	0.704	0.701
6	MLP	648	8-8-5-1	Momentum	Tanh	Learning	0.010	0.043	0.956
						Testing	0.049	0.141	0.927
7	MLP	1000	7-4-4-1	Momentum	Tanh	Learning	0.013	0.069	0.931
						Testing	0.145	0.717	0.374
8	MLP	1000	7-7-8-1	Momentum	Tanh, Linear	Learning	0.013	0.082	0.916
						Testing	0.045	0.228	0.791
9	MLP	1000	9-7-8-1	Momentum	Tanh, Linear	Learning	0.013	0.056	0.943
						Testing	0.029	0.116	0.915
10	MLP	155	8-4-1	Momentum	Tanh	Learning	0.010	0.035	0.964
						Testing	0.010	0.034	0.964

networks could not substitute the MODFLOW model as one single model, but for every individual observation well, the designed model simulated the water table fluctuation very well. ANN topology is problem dependent and whatever type of ANN model is used, it is important to determine the appropriate network architecture in order to obtain satisfactory generalization capability. An understanding of the topology as a whole is needed before the number of hidden layers and the number of processing elements (PEs) in each layer can be estimated. In order to find the best structure, different numbers of hidden layers were tested. In addition, various transfer functions and memory models were trained and tested to obtain an optimum network. The chosen network's topology for each case will be discussed in the next section. Three selection criteria were used to compare the networks which were the mean squared error (MSE), the normalized mean squared error (NMSE), and the coefficient of determination (r^2).

In order to overcome the difficulties with the first algorithm, TLRN models were tested to simulate the whole groundwater system with one ANN model. Different input parameters, as shown in Table 10.1, were tested to find the minimum effective input set.

10.3 Results and Discussion

An MLP network was used to estimate the water table in each observation well. MODFLOW outputs were used for training and testing the ANN model. In each model, the effective parameters were selected using a trial and error method. Seventy-five percent of the 144 data points were used for training and the rest for testing the model. After testing different network topologies, the best network for each well is shown in Table 10.2. As can be seen in most cases, the ANN could predict the water table very well and the r^2 values are high.

There are, however, two problems with these networks. First, there is no single model with similar input parameters that can predict water tables in all observation wells. Second, these models are static and do not consider inputs and outputs from previous time steps (unless these are introduced explicitly). Therefore, as shown in Fig. 10.5 for instance, at some points the difference between the observed and calculated output is too high.

TLRN models do take into account the lagged inputs and in this respect compare favorably with MLP networks. Figures 10.6 and 10.7 show the water surface level in observation well no. 1 to show the comparison between the calculated data from MODFLOW and TLRN model outputs with 10 and 4 input parameters, respectively.

One year of available data was used to calibrate MODFLOW and it was then used to produce 144 data points to train the ANN models. A multi-layer perceptron trained with the backpropagation algorithm was used to estimate the water table in every observation well. In addition, two TLRN models were constructed, one with all available input parameters and one with minimum effective input parameters, to estimate water tables in all 10 observation wells. Considering computational costs and data requirements, the results shown in Table 10.3 indicate that a TLRN model can be effectively used in the field of groundwater simulation.

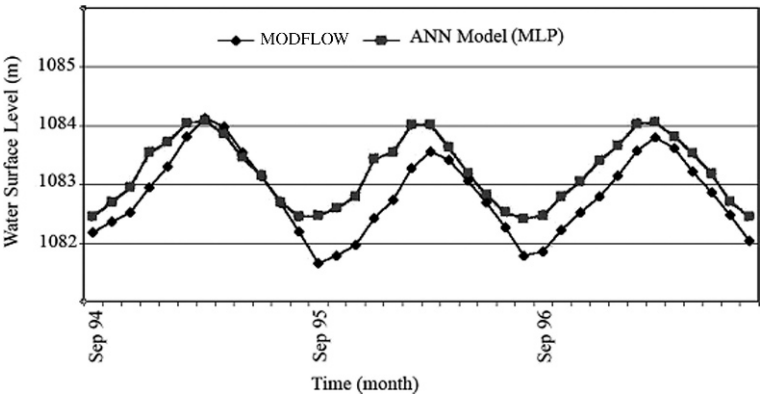


Fig. 10.5 Comparison of MODFLOW and MLP model results for the testing data set – well no. 1 (Table 10.1)

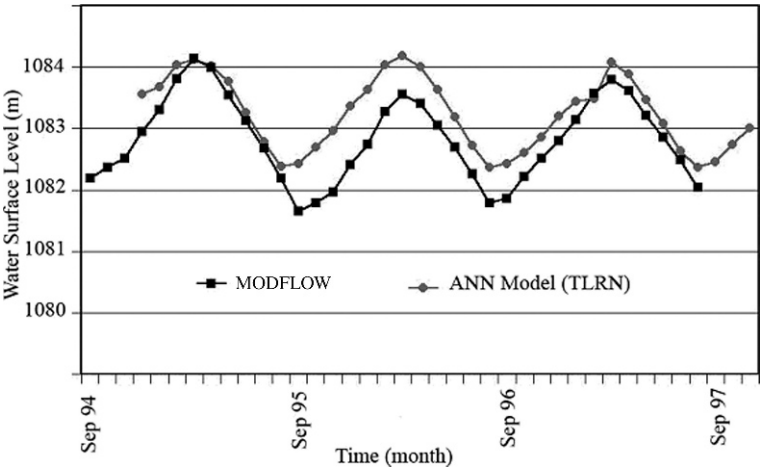


Fig. 10.6 Comparison of MODFLOW and TLRN with 10 input variables for the testing data set – well no. 1 and model no. 6 (Table 10.2)

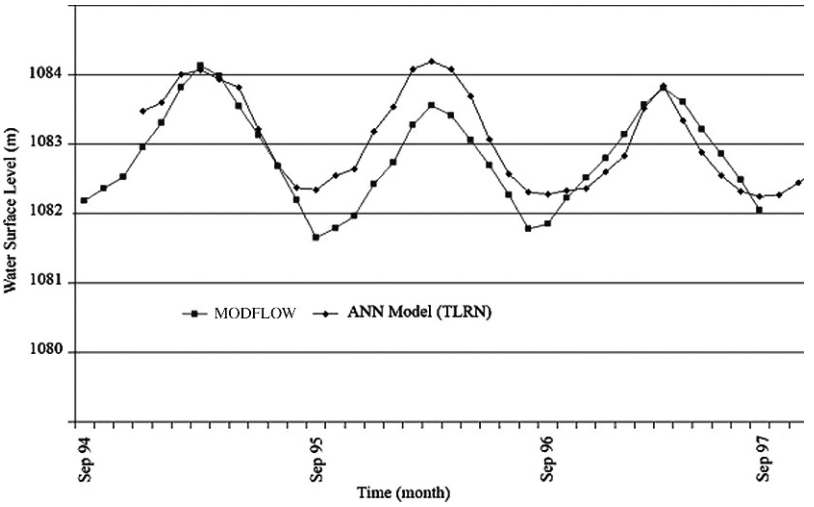


Fig. 10.7 Comparison of MODFLOW and TLRN with four input variables for testing data set – well no. 1 and TLRN network (model no. 5, see Table 10.2)

Table 10.3 TLRN networks for simulating observation wells simultaneously

Test no.	ANN model	Epoch	Topology	Learning rule	Transfer function		MSE	NMSE	R^2
1	Fully recurrent	1859	4-10-10-10	Conjugate Gradient	Tanh- Sigmoid	Learning	0.016	0.294	0.846
						Testing	0.019	0.312	0.823
2	Fully recurrent	1446	4-10-10-10	Delta Bar Delta	Tanh- Sigmoid	Learning	0.013	0.246	0.878
						Testing	0.017	0.279	0.853
3	Fully recurrent	451	10-10-10-10	Conjugate Gradient	Tanh- Sigmoid	Learning	0.017	0.302	0.841
						Testing	0.020	0.322	0.833
4	Fully recurrent	429	10-10-10-10	Delta Bar Delta	Tanh- Sigmoid	Learning	0.014	0.249	0.892
						Testing	0.016	0.273	0.880
5	Time lag recurrent	2000	4-10-10-10	Conjugate Gradient	Tanh- Sigmoid	Learning	0.005	0.088	0.951
						Testing	0.009	0.151	0.923
6	Time lag recurrent	1000	10-10-10-10	Conjugate Gradient	Tanh- Sigmoid	Learning	0.005	0.086	0.958
						Testing	0.007	0.110	0.945
7	Time lag recurrent	5000	4-10-10-10	Momentum Online	Tanh- Sigmoid	Learning	0.006	0.109	0.944
						Testing	0.009	0.153	0.922
8	Time lag recurrent	2741	10-10-10-10	Momentum Online	Tanh- Sigmoid	Learning	0.006	0.110	0.942
						Testing	0.009	0.142	0.925
9	Time lag recurrent	11000	4-10-10-10	Momentum Batch	Tanh- Sigmoid	Learning	0.00803	0.1462	0.918
						Testing	0.01209	0.1968	0.897

10.4 Conclusions

In this research, several mathematical models were tested to simulate the water surface level of Chamchamal plain. A commonly used simulation model, MODFLOW, and two different artificial neural network algorithms were used in this study. All models could estimate the water table with reasonable accuracy but the ANN models needed less input data and took less time to run, indicating advantages of ANNs over other common numerical models. The only drawback with ANN models is related to their dependency in having enough input and output data to train them.

References

Basheer I.A. and Hajmeer M. (2000) Artificial neural networks: Fundamentals, computing, design and application. Journal of Microbiological Methods, 43, 3–31.

Coulibaly, P., Anctil, F., Aravena, R. and Bobee, B. (2001) Artificial neural network modeling of water table depth fluctuations. Water Resources Research, 37(7), 885–896.

Hsu, K., Gupta, H.V. and Sorooshian, S. (1995) Artificial neural network modeling of the rainfall-runoff process. Water Resources Research, 31(10), 2517–2530, 1995.

Lebron, I., Schaap, M.G. and Suarez, D.L. (1999) Saturated hydraulic conductivity prediction from microscopic pore geometry measurements and neural network analysis. Water Resources Research, 35, 3149–3158.

Lefebvre, C. and Principe, J. (1998) NeuroSolotions User's Guide, Neurodimension Inc., Gainesville, FL., USA, 786p.

McDonald, M.G. and Harbaugh, A.W. (1989). A modular three-dimensional finite-difference ground-water flow model. USGS Techniques of Water Resources Investigations, Book 6, Chapter A1. Washington DC.

Maier. H.R. and Dandy, G.C. (2000) Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. Environmental Modelling and Software, 15, 101–124.

Morshed, J. and Kaluarachchi, J. J. (1998) Parameter estimation using artificial neural network and genetic algorithm for free-product migration and recovery, Water Resources Research, 34(5), 1101–1114.

Ranjithan, S., Eheart, J.W. and Garrett, Jr., J.H. (1993) Neural net work-based screening for groundwater reclamation under uncertainty, Water Resources Research, 29(3), 563–574.

Rizzo, D.M. and Dougherty, D.E. (1994) Characterization of aquifer properties using artificial neural networks: Neural Kriging, Water Resources Research, 30(2), 483–497.

Schaap, M.G. and Bouten, W. (1996) Modelling water retention curves of sandy soils using neural networks, Water Resources Research, 32(10), 3033–3040.

Schalkoff, R.J. (1997) Artificial Neural Networks, McGraw-Hill Inc., New York.

Winston, R.B. (1999) Upgrade to MODFLOW-GUI: Addition of MODPATH, ZONEBDGT, and additional MODFLOW packages to the U.S. Geological Survey MODFLOW-96 Graphical-User Interface: U.S. Geological Survey Open-File Report 99–184, 63p.

Chapter 11
Neural Network Estimation of Suspended Sediment: Potential Pitfalls and Future Directions

R.J. Abrahart, L.M. See, A.J. Heppenstall and S.M. White

Abstract This chapter examines two neural network approaches for modelling suspended sediment concentration at different temporal scales: daily-record and flood-event. Four daily-record models are developed for the USGS gauging station at Quebrada Blanca near Jagual in Puerto Rico previously used by Kisi (2005) for estimating suspended sediment concentration: comparisons with that earlier investigation are presented. The flood-event approach is trialled on records for the EA gauging station at Low Moor on the River Tees in northern England. The power of neural networks to perform different types of modelling operation and to develop reasonable results in the two test cases is highlighted. Event-based modelling of mean suspended sediment concentration is a novel concept that warrants further trialling and testing on different international catchments or data sets.

Keywords Sediment modelling · neural network · hysteresis

11.1 Introduction

The timing and pattern of sediment movement in response to rainfall is of interest to many different types of stakeholder and manager. The range of interested parties is vast and spans a broad set of interrelated fields. Major participants would include:

R.J. Abrahart
School of Geography, University of Nottingham, University Park, Nottingham, NG7 2RD, UK

L.M. See
School of Geography, University of Leeds, Woodhouse Lane, Leeds, LS2 9JT, UK

A.J. Heppenstall
School of Geography, University of Leeds, Woodhouse Lane, Leeds, LS2 9JT, UK

S.M. White
Integrated Environmental Systems Institute, Cranfield University, College Road, Cranfield, Bedfordshire, MK43 OAL, UK

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

- scientists trying to understand catchment hydrology, contaminant transport, water-quality trends or the harmful effects of pollutants on fisheries, waterfowl habitats and stream ecologies (Milner et al., 1981; Ottaway et al., 1981; De Vries and Klavers, 1994; Horowitz, 1995; Newcombe and Jensen, 1996; Marks and Rutt, 1997);
- policy-makers and concerned citizens worried about potential environmental impacts related to: establishing forests and the harvesting of timber (Stott and Mount, 2004); road construction (Fransen et al., 2001; MacDonald et al., 2001) or use of snow-fencing to trap drifting snow and so produce increased river discharge volumes (Sturges, 1992);
- resource managers interested in the control of soil erosion and loss (Walling, 1999); or the effects of that process on the infilling of lakes and reservoirs – reducing drinking water storage capacities and causing operational problems for scour valve releases (Labadz et al., 1995);
- commercial organisations involved in generating hydroelectric power from glacial meltwater streams (Østrem, 1975; Fenn et al., 1985).

The recent surge of interest in suspended sediment estimation is also related to the current requirement for efficient and effective models that can be used to assess the potential impact of environmental change: modelling past, present and future sediment movement is an important component in such undertakings. Land use changes that lead to flood events of increased magnitude and duration can be very instrumental, especially under unstable channel conditions, for the production of increased sediment concentrations. Examples of elevated sediment deliveries related to land use changes include clearance of natural vegetation to provide land for cultivation (Morgan, 1986); channelisation for land drainage and flood control (Simon and Hupp, 1986); plus silvicultural or reservoir impacts (US Environmental Protection Agency, 1980; Lopes et al., 2001; Rosgen, 1996; Rosgen, 2001). The opposite situation holds for programmes that attempt to restore natural vegetation, such as the conversion of farmland into forestland and grassland, which can be used to control gully erosion-induced sediment yield (Hao and Qiangguo, 2006). The potential impact of changing climatic conditions related to global warming and its effect on the processes of sediment production and transportation is another major driver. Walling and Fang (2003) report that it is difficult to disentangle the influences of climate change from that of other factors that affect catchment conditions: however, they offer clear evidence that sediment loads in some rivers are changing; whilst others show little sign of significant temporal trends. More frequent storms or more violent weather will nevertheless produce a larger number of extreme events, i.e. the type of event that is most influential in controlling erosion and sediment transport, especially in semi-arid regions (Coppus and Imeson, 2002).

The mechanism of sediment transport in natural streams and rivers is a complex process that is difficult to model in a traditional or conventional manner. No direct or indirect empirical model of sediment transport has received universal acceptance and the challenge to discover a superior solution continues to thwart scientists and practitioners. Typically such estimates are based on a simple relationship that is established between sediment and discharge: no allowance is made for climatic factors,

or catchment characteristics and conditions, in terms of differing temporal or spatial variability. To model an exact rating relationship under such circumstances is quite difficult, however, due to (i) the broad scatter of points that is created and (ii) the presence of hysteresis. The sediment concentrations for a given level of discharge on the rising limb of the hydrograph, can be higher or lower than the sediment concentrations for a given level of discharge on the falling limb, depending on the relative position of the sediment source to the point of measurement and on the amount of sediment that is available for transportation. This often produces large levels of scatter, in complex data sets that possess only partially significant correlations between sediment and discharge (Chikita et al., 2002). Figure 11.1 illustrates modelling difficulties related to the production of a sediment rating curve for manual samples collected at 19 points spread across 11 rivers in northern England (Aire, Calder, Derwent, Don, Nidd, Ouse, Swale, Trent, Tweed, Ure and Wharfe). The monitoring period spanned 1993–1997; samples were collected at irregular intervals and sought to encompass a broad spread of different event types. This regional model demonstrates several important sediment modelling issues: (i) that simple models can be developed on intermittent sampling; (ii) that broad numerical approximations can be developed on spatial and temporal aggregations; and (iii) that low accuracies can be expected due to the natural scatter of points.

The most common form of hysteresis is the clockwise loop caused by early sediment depletion (Lenzi and Marchi, 2000). This implies that sediment concentration peaks before river discharge, as sufficient sediment is not available to meet transport capacity at higher discharge rates. An anticlockwise loop (discharge leading sediment) is thought to demonstrate activation of more distant sediment supplies. The complexities of modelling such relationships will also be influenced by two other important factors: (i) some rivers will exhibit different behaviours at different points in time and (ii) most sediment–discharge relationships are heteroscedastic, i.e. greater variability occurs at higher levels of discharge; lower levels exhibit more regular relationships. This challenging situation has resulted in a long track record

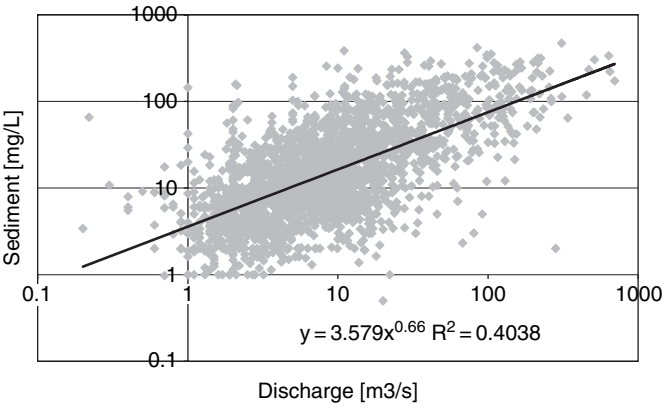


Fig. 11.1 Sediment rating curve: traditional model of a regional relationship developed on 19 sampling sites spread over 11 rivers in northern England (1993–1997)

of model outputs that are highly inaccurate at peak discharge rates. This is due in part to the use of aggregated temporal measurements that present a 'dampened response'. However, from a pragmatic viewpoint, such tools can nevertheless provide accurate and reliable annual and longer term estimates of load since the errors are averaged out (Horowitz, 2003; Crowder et al., 2007). The demand for better sediment modelling methodologies that can support shorter period daily, weekly, monthly or quarterly requirements has in consequence escalated (Horowitz, 2003; White, 2005).

The two sediment-related attributes that are typically estimated for a river and perhaps equate to separate modelling problems are (i) relative sediment concentration and (ii) absolute sediment load. The two measurements are of course interchangeable and conversion is performed with the aid of discharge data sets. Each item could be modelled at different temporal resolutions ranging from instantaneous records or daily averages to seasonal budgets or annual totals. The main benefit of predicting sediment concentration records, however, occurs at shorter intervals where it more useful; whereas the main benefit of computing total sediment loads is over longer periods due to lower error. It is also possible to model a continuum of temporal intermediaries but with strategic tradeoffs occurring in terms of (i) regulatory requirements; (ii) sampling capabilities; and (iii) acceptable error.

11.1.1 Estimation of Short-Term Dynamic Time Series Records

Individual records, at one end of this scale, can be used to predict time series concentrations for pollution monitoring purposes, to evaluate ecological thresholds or to perform ecosystem health appraisal of rivers and estuaries. Model outputs could be concurrent predictions or short period forecasts of suspended sediment. There is, however, often no clear relationship at this scale of measurement between instantaneous suspended sediment and concurrent river discharge records due to the activation or exhaustion of sediment supplies. It produces a scatter of points that contains a strong hysteresis element, expressed in different relationships related to the rising and falling limbs of each flood hydrograph – something that is difficult to encapsulate. The dynamic role of sediment in fluvial ecosystems and estuarine wetlands is nevertheless of great importance with respect to the environmental aspects of sediment transport. It impacts on both river systems and at sink points – such as natural or artificial bodies for water and sediment storage. Nutrient fluxes are of particular concern since nutrients perform a critical role in determining the ecology of rivers, lakes and estuaries. Transported sediments can also have contaminants (including nutrients) bound to them such that an assessment of contaminant flux and delivery will require prior estimates to be made of sediment input or throughput. The need to produce estimations at high(er) temporal resolutions for unsampled sites is paramount. For example, to deal with problems related to excess sediment or its presumptive biological impact the US Clean Water Act of 1972 requires estimated total maximum daily loads (Keyes and Radcliffe, 2002).

11.1.2 Estimation of Long-Term Throughputs or Delivered Loads

The use of aggregated records, at the other end of the scale, supports lower temporal resolution modelling of total throughput (flux) or delivered load (input) computed over long(er) periods, e.g. solutions developed on mean annual data sets that might span several decades. There is a much clearer relationship at this scale of measurement between suspended sediment and river discharge records due to the averaging effect. It is easier to model such relationships using linear, concave or convex solutions. However, the exact nature of that modelled relationship might change over time, thus creating problems for longer term forecasts of sediment load (Crowder et al., 2007). Model outputs could be used to support the assessment of potential loads on engineering structures, or related to environmental damage that might results from the impact of accumulated pollutants on biological communities, such as local marine life and coral reefs (Bellwood et al., 2004; Warne et al., 2005). The sedimentation of reservoirs and channels is a relevant international issue; so too is the need for planning and management of dredging activities, to meet either downstream navigation requirements (Hansen et al., 2002) or for flood protection purposes, e.g. the regular maintenance of smaller streams and ditches. Total load estimates can be used to manage material resources, such as impounded water bodies, so as to minimise sediment trapping or water-quality problems. Effective modelling could maximise the potential benefits and economic life of major structures and river regulation schemes. It is also important in the design stage since reservoirs are required to accommodate their incoming sediment, i.e. to provide what is termed 'dead storage' – excess capacity that will be filled over a specified period. The use of inadequate models during the specification process can have serious repercussions: insufficient storage capacities will result in inadequate resources (underestimation); whereas the provision of unused storage capacities will entail wasted resources (overestimation). The need to develop reliable annual estimates for pollution monitoring and control purposes also looks destined to expand. For example, annual mean concentration is likely to be used as a target under the EEC Water Framework Directive and is already used under the EEC Fisheries Directive (target level set at 25 mg l^{-1}).

11.1.3 Estimation of Mean Sediment Concentrations per Flood Event

Somewhere towards the middle of this continuum lies the potential estimation of mean sediment concentration for individual flood events, i.e. for a rainfall-related sequence of individual temporal records. For most rivers in Europe, between 70% and 90% of sediment load is moved in the top 20% of discharge magnitudes, i.e. during high and peak flood events. The sediment–discharge relationship is highly skewed, but through an understanding of sediment flux and the manner in which it varies over different events, it should be possible to improve our estimates of both

short-term sediment concentration and long-term sediment load. The flood-event approach that is proposed at the end of this chapter is quite demanding since it requires continuous recording of instantaneous discharge and sediment concentration measurements throughout each flood event. However, using an event based approach allows the hysteresis problem to be sidestepped, and might provide a route for the identification of different event types that possess a distinct hysteretic behaviour. It is also anticipated that subsequent investigations will produce models that require less detailed data sets and offer better predictive capabilities.

It is important to recognise that different issues at different scales could be resolved using different methods. Merritt et al. (2003) reviewed numerous models and concluded that empirical and conceptual methodologies might perhaps be combined in a constructive manner to help overcome the less well suited approaches of more complex theoretical or physics-based solutions. The extent to which complex distributed models offer substantial advantages over simple lumped models is also doubtful in cases where the real spatial and temporal distributions are not considered. The use of empirical and conceptual methodologies will instead offer practical operational benefits, e.g. simple transparent solutions, that are easier to understand, and can be implemented in a fast and efficient manner. Moreover, for practical reasons, the rapid assessment of erosion rates or sediment movement must in most cases be made without recourse to detailed and expensive field monitoring programmes, thus forcing modellers to accept less demanding methods. The associated development of lower cost methods requiring fewer samples has in recent times been the main focus of interest: the motivation behind such methods is basic economics, e.g. development of improved rating curve estimators and statistical sampling procedures that will reduce the number of samples required to obtain acceptable estimates (Thomas, 1985, 1991; Thomas and Lewis, 1993, 1995). The need to develop minimalist empirical approaches that are sensitive to catchment characteristics, anticedent conditions and climatic control, and will also will permit the investigation of sediment source strengths, remains a long-term goal.

The hydrologist has as a result at his/her disposal a recognised set of numerical procedures that can be used to estimate various parts of the sediment supply and sediment transfer system. More than 20 such techniques for relating suspended sediment to discharge have in fact been reported. The most common method is to develop a simple linear regression model (or its power function equivalent) that relates log suspended sediment concentration (independent variable) to log water discharge (dependent variable). The resultant sediment rating curve can thereafter be expressed in one of two standard formats – as a linear regression model:

$$\text{Log } S_t = \text{Log } a + b \text{ Log } Q_t \tag{11.1}$$

or as a power function model:

$$S_t = aQ_t^b \tag{11.2}$$

The shortfalls of such methods are well documented and comparisons of actual and predicted sediment concentrations indicate that substantial under-prediction could result, e.g. see Ferguson (1986), Walling and Webb (1988) and Asselman

(2000). Numerous methodological modifications have been proposed and applied to compensate for such errors. These include the development of compound models fitted to distinct seasonal or hydrological groupings, the application of mathematical correction factors, the use of non-linear regression equations (Duan, 1983, Ferguson, 1986; Walling and Webb, 1988; De Vries and Klavers, 1994; Phillips et al., 1999; Asselman, 2000; Holtschlag, 2001; Crowder et al., 2007) and the implementation of robust iterative local regression techniques (Krishnaswamy et al., 2001). More recent studies have investigated the potential advantages of using machine learning algorithms, e.g. neural network (NN) and M5 model tree (M5MT) approaches; several different types of NN have in fact been used to conduct sediment modelling experiments (Abrahart and White, 2001; Jain, 2001; Cigizoglu, 2002a,b; Cigizoglu, 2004; Nagy et al., 2002; Agarwal et al., 2005; Kisi, 2004; Kisi, 2005; Cigizoglu and Kisi, 2006).

NN modelling has introduced a major shift in focus: numerous different modelling opportunities can be realised. The sediment rating solutions that are being trialled and tested no longer equate to the production of a simple mathematical relationship, that is developed between concurrent sediment and discharge records, in the manner of a traditional log–log linear regression or power function model. It is of course possible to model a single-input single-output relationship using such methods and to develop the required curvilinear solutions. NN might also be capable of avoiding difficult issues surrounding the requirement to perform a logarithmic transformation in the case of a zero recorded sediment measurement. The neural solution might contain strong non-linear properties, near linear properties or some mixed grouping of both types of relationship. It would not, however, address or resolve the two major modelling difficulties that confront traditional methods: a broad scatter of points and the hysteresis effect. To help overcome such issues some experimenters have resorted to using different sets of predictors for the estimation of current sediment, combining their current discharge input with some mix of lagged discharge and lagged sediment records. Cigizoglu (2002b) for instance used combinations of current and lagged discharge inputs to predict suspended sediment concentration and was to some extent successful in capturing certain aspects of the hysteresis loop. Figures 11.2 and 11.3 show that different curves were approximated but the end result is nevertheless an averaging mechanism; the major trajectories are not well captured and everything is reduced to a single loop solution. Cigizoglu and Kisi (2006) instead experimented with a mixed set of different input drivers for the prediction of current suspended sediment load: current discharge, past discharge and past sediment records were considered. The justification for selecting appropriate input drivers must nevertheless be twofold: logic on the one hand dictates the requirement for a causal relationship between predictor and predictand; numerical testing on the other can be used to reveal the strength and direction of some plausible ‘indirect association’. The later option for selecting and using past sediment records as inputs to a data-driven model, irrespective of potential benefits, has, however, generally led to a poorer result. Moreover, it is important to stress that the use of past sediment records as an input to the modelling process makes no operational sense: each solution requires one or more past sediment inputs such that a

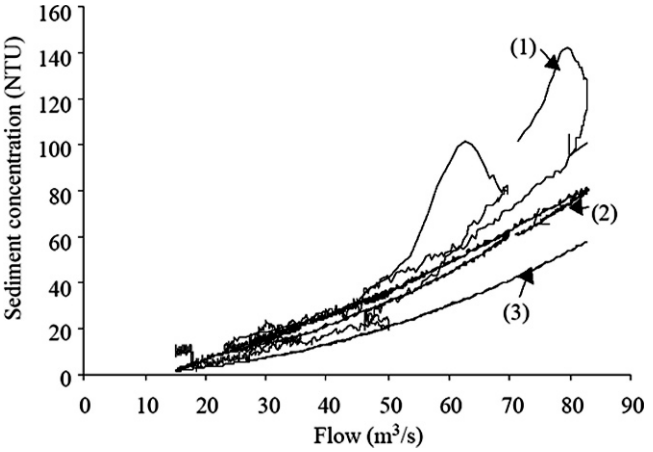


Fig. 11.2 Sediment modelling outputs for the River Tees at Low Moor, 26 February–17 March 2000: (1) observed hysteresis; (2) neural network; (3) sediment rating equation (Cigizoglu, 2002b; reproduced with permission from the *Turkish Journal of Engineering and Environmental Sciences*)

full set of observed measurements would be required to run the model. However, good sediment data sets are seldom available, and the short-term need for sediment concentrations versus the long-term need for total loads must also be considered. If a short-term estimation was required for pollution control purposes the need for a current discharge input would mean that the related environmental damage from suspended sediment had already occurred. If the requirement was related to some long-term purpose the need to possess a full set of existing records would mean that such items could of course be summed to give a total. Two other issues are also important: extended temporal forecasts beyond the present are not supported since

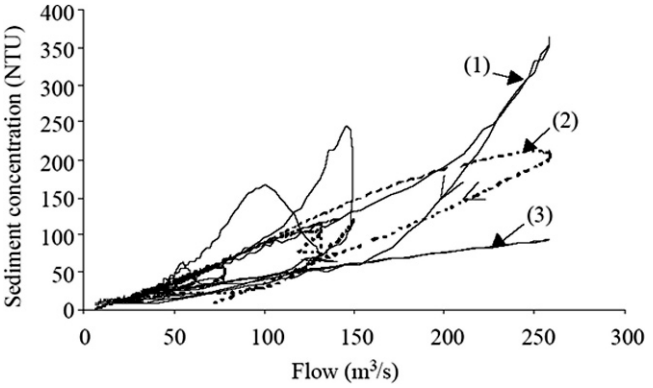


Fig. 11.3 Sediment modelling outputs for the River Swale at Thornton Manor, 14–25 December 1994: (1) observed hysteresis; (2) neural network; (3) sediment rating equation (Cigizoglu, 2002b; reproduced with permission from the *Turkish Journal of Engineering and Environmental Sciences*)

no relevant future sediment values exist; solutions cannot be transferred to similar rivers or to different periods, if no observed suspended sediment records are available for use as inputs to the sediment prediction model.

The rest of this chapter considers two different examples of sediment modelling: daily-record modelling of standard data sets, to highlight various issues in developing and comparing a set of simple curve fitting procedures, and event-based modelling that overcomes the hysteresis problem by using selected inputs that drive the sediment supply process. The two different methods are contrasted in terms of operational issues and demands.

11.2 Daily-Record Scale Modelling

Four daily-record suspended sediment models were developed for a small catchment on the island of Puerto Rico, located in the north-eastern corner of the Caribbean Sea. This island has a tropical marine climate, experiencing uniform temperature, high humidity and variable rainfall. Mountains and adjacent foothills cover most of the land. The humid tropical environment and mountainous terrain are conducive to high rates of sedimentation (Warne et al., 2005). Local variation is related to the interaction of topographic factors and the prevailing trade winds; hurricanes are both frequent and severe (Boose et al., 2004). Daily-record modelling was applied to sediment and discharge data sets for a United States Geological Survey (USGS) gauging station in Puerto Rico: Quebrada Blanca near Jagual (QB; Fig. 11.4; USGS Station No. 50051150; latitude 18°09′40″N; longitude 65°58′58″W). Figure 11.5 contains a sediment–discharge plot of the calibration data set; the SRC model that is fitted to this scatter of points provides a typical example of excess demands being placed on a simple power law function (Hicks and Gomez, 2003). The model offers a reasonable fit to the mass of lower level records, but a poor fit to the most important higher magnitude discharges, the latter frequently comprising a short tail of sparse points on the right-hand side of a log–log plot. The reported experiments mimic one of two earlier reported modelling applications (Kisi, 2005); modelling input data sets and architectural configurations identified as optimal in that earlier paper were used in the current studies to permit a direct comparison of modelling methodologies and related outputs.

This follow-on modelling exercise sought to extend the earlier reported studies in two ways: (i) through the inclusion of a superior learning algorithm and (ii) by means of a reinterpretation of modelling outcomes. Further details on the relevant data sets and original selection procedures can be found in that earlier paper. The gauging station, at an elevation of 130 m, monitors a small upstream drainage area ($8.42\,km^2/3.25$ square miles). Daily river discharge ($m^3\,s^{-1}$) and suspended sediment concentration ($mg\,l^{-1}$) records were downloaded from the USGS: http://webserver.cr.usgs.gov/sediment. Models were developed on data for the 1994 water year (365 daily values from October 1993 to September 1994); testing was performed on data for the 1995 water year (October 1994–September

Fig. 11.4 USGS gauging station at Quebrada Blanca on the island of Puerto Rico

1995). Table 11.1 lists four different input combinations that were used to predict the suspended sediment concentration records S_t, i.e. suspended sediment in mg l^{-1} on day t. D1 and D2 use different discharge inputs to predict suspended sediment; whereas D1S and D2S include past sediment records. The last two models in this list offer diagnostic support but are otherwise less useful since (i) they rely on

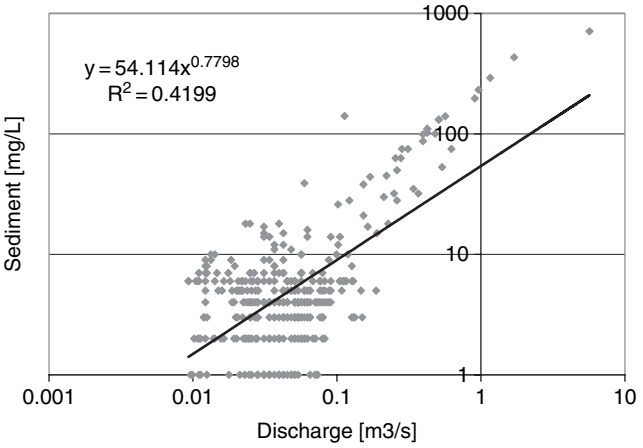


Fig. 11.5 SRC model of sediment–discharge relationship developed on training data set for Quebrada Blanca (October 1993–September 1994)

Table 11.1 NN inputs and architectures (Kisi, 2005)

Experiment	Model inputs	Neural network architecture (input:hidden:output)
D1	Q_t	1:2:1
D2	Q_t and Q_{t-1}	2:2:1
D1S	Q_t and S_{t-1}	2:2:1
D2S	Q_t, Q_{t-1} and S_{t-1}	3:2:1

continuous sediment records at the site and (ii) no operational requirement for this type of solution has so far emerged.

NN solutions were developed on the four different input combinations using symbiotic adaptive neuro-evolution (SANE: Moriarty and Miikkulainen, 1998). Neuroevolution is the use of genetic algorithms to train artificial neural networks. For each set of inputs a population of candidate solutions is evolved and a winner selected. Model development is based on the concept of 'cooperative coevolution' (Horn et al., 1994; Potter, 1997). This approach advocates the evolution of a population of hidden neurons in contrast to the more traditional approach of evolving a population of functional networks. Each individual member in the hidden neuron population can be viewed as a partial solution. To create a complete and optimal solution, each individual must form connections with other hidden neurons in that population. Individuals are thus organised into groups that optimise different parts of the solution space, doing so in cooperation with other groups. Established genetic operators such as mutation and crossover are used to evolve different types of hidden neuron specialisation. This approach has several advantages over the use of traditional methodologies: cooperation results in a faster and more aggressive search of the solution space; evolution of neuron specialisation preserves diversity in the candidate population; and, since convergence does not tend towards a particular peak, the resultant sub-optimal solutions can readily adapt to changes. The cooperative coevolution algorithm is incorporated into two software packages: SANE-C (research code) and JavaSANE (platform-independent code that requires minimum effort to implement novel applications in fresh domains).[1] Further examples of related applications in the water sector include Heppenstall et al. (2008), Abrahart et al. (2007) and Dawson et al. (2006).

JavaSANE neural network (JSNN) models were developed on root mean squared error (RMSE). Numerous candidate solutions were evolved and for each input scenario a best model selected, the criterion for selection being RMSE calculated on its training data set. That model was then used to make predictions for the test data set. Kisi (2005) provides a set of comparative statistics derived for a neuro-fuzzy system (NFNN: Jang, 1993; hybrid combination of artificial neural network and fuzzy logic); a backpropagation neural network (BPNN: Rumelhart et al., 1986; training was stopped after 10,000 epochs); a traditional sediment rating curve (SRC; (11.1) and (11.2)); and a multi-linear regression model (MLR). JSNN and BPNN models

[1] http://nn.cs.utexas.edu/pages/software/software.html

used identical structures; reported output differences are thus a direct reflection of the different training procedures and learning algorithms that were used. Neuroevolution modelling is acknowledged to produce sub-optimal populations in which the potential dangers of overfitting are substantially reduced. Thus no 'early stopping procedure' or 'cross-validation data set' is required for model development purposes (Giustolisi and Laucelli, 2005); BPNN models developed with no strong regard for such issues should in contrast be questioned and tested since their solutions will have a high potential to overfit.

The power of each method to model specific combinations of inputs was evaluated using the test data set and compared to the reported findings of Kisi (2005); Tables 11.2–11.4 provide RMSE, r-squared (R-Sqd) and relative error in total sediment (RETS) statistics. The neural solutions on most occasions tended to provide similar levels of fit which is to be expected since the demands of the reported experiment were somewhat limited; the question of minor numerical differences being meaningful or not is an important issue in such cases. It is indeed possible that the final ordering of such tight outcomes might be some facet of peculiarities related to the manner in which the material was presented, or to random runtime

Table 11.2 RMSE for test period (water year 1995)

Exp	NFNN*	BPNN*	JSNN	SRC*	MLR*
D1	**17.96**	27.32	18.36	53.11	29.99
D2	21.40	21.40	**18.14**	–	29.99
D1S	19.68	36.30	**17.72**	–	29.80
D2S	21.97	21.59	**21.20**	–	29.80

* From Kisi (2005). Best score for each data set is in bold.

Table 11.3 R-Sqd for test period (water year 1995)

Exp	NFNN*	BPNN*	JSNN	SRC*	MLR*
D1	**0.929**	0.821	0.920	0.816	0.894
D2	0.887	0.865	**0.934**	–	0.894
D1S	0.907	0.754	**0.925**	–	0.891
D2S	0.883	0.888	**0.930**	–	0.890

* From Kisi (2005). Best score for each data set is in bold.

Table 11.4 RETS for test period (water year 1995)

Exp	NFNN*	BPNN*	JSNN	SRC*	MLR**
D1	10.9	#	**7.9**	−83.0	#
D2	#	**9.2**	−11.3	#	#
D1S	#	#	**7.3**	#	−31.0
D2S	#	#	**−17.4**	#	−31.0

* From Kisi (2005); ** best model not identified; # not provided. Best score for each data set is in bold. Numbers expressed in percentage format.

processes, arising out of program mechanics. It must nevertheless be stressed that this modelling exercise is not about implementing a set of simple curve fitting exercises or about selecting an overall winner with the best statistics. It is about investigating the different toolkits on offer for the purposes of learning from such experiences; about their similarities and differences in terms of gains or losses; and about the specific nature of individual solutions developed under controlled conditions. The results in this context can be summarised as follows:

D1: NFNN was the best performing sediment rating curve model in terms of RMSE and R-Sqd; JSNN was a close second. BPNN was a fair bit poorer. NFNN, with its 'divide and conquer' four-rule mechanics, is found to provide a better approximation for the scatter of points. JSNN equated to a better parameterised version of BPNN.

D2: JSNN was the best performing two-input discharge model in terms of RMSE and R-Sqd; it was also the best overall performing model in terms of R-Sqd. JSNN was also a clear winner; NFNN and BPNN produced a set of similar but much weaker numerical statistics. JSNN with its superior calibration procedures is thus found to accommodate the additional discharge input in a more meaningful manner. The inclusion of a past discharge input had mixed effects on the other two neural models; NFNN statistics worsened, whilst BPNN statistics improved. Potential gains related to a traditional architecture could perhaps be explained in terms of stronger pattern detection and knowledge extraction capabilities or superior non-linear modelling properties.

D1S: JSNN was the best performing mixed input model in terms of RMSE and R-Sqd; it was also the best overall performing model in terms of RMSE. JSNN with its superior development procedures was once again able to accommodate the additional material in a more meaningful manner. NFNN and BPNN were both much poorer vis-à-vis D1; showing that the additional input was confusing matters. NFNN performed better vis-à-vis D2; BPNN performed worse vis-à-vis D2. NFNN and its rule-based components seemed better able to counter the input noise related to the incorporation of past sediment records; BPNN was less able to deal with sediment-related noise, but more able to incorporate the hydrological relationships contained in past discharge records. JSNN model outputs for the D1S test data set are provided in Fig. 11.6.

D2S: JSNN was once again the best performing mixed input model in terms of RMSE and R-Sqd, but with RMSE being much higher than in D1S. This suggests that the use of a second discharge input did not improve the model, contrary to what happened in the case of moving from D1 to D2. NFNN and BPNN are close seconds providing near-identical measures of performance on both statistics, similar to what happened in the case of D2. NFNN lost ground vis-à-vis D1S; BPNN gained ground. The use of additional input variables does not in such cases appear to have had a consistent impact; such findings are perhaps indicative of previous development procedures having focused on dissimilar aspects of the captured relationship.

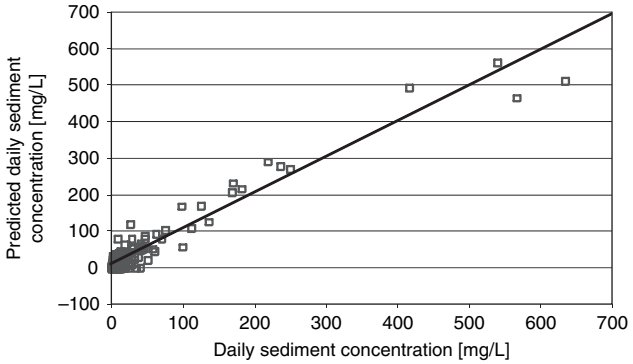


Fig. 11.6 JSNN daily-record model outputs for DS1 test data set. Note: model tends to underestimate some of the larger peaks, a problem also encountered by Kisi (2005). This can be attributed to: (i) having a small number of peak events in the training data set and (ii) the use of a global model. It is possible to address both issues with computational procedures such as boosting or the use of local modelling techniques (Anctil and Lauzon, 2004)

The statistical closeness of the final models is of particular interest since it is possible to speculate that using more inputs places the greatest demands on each model; thus modelling more complex situations has perhaps forced the solutions to address similar aspects of that relationship and to converge towards some sort of 'common ground'.

RETS: Table 11.4 has several missing numbers: Kisi (2005) provided total sediment assessments for the best model of each type but not for all contenders. JSNN was the best performing model overall; the best result was obtained for JSNN on D1S; the second best result for JSNN was on D1. The two values are rather similar and much better than either the best BPNN (D2) or best NFNN (D1). BPNN (D2) had poorer RMSE and R-Sqd statistics than NFNN (D1) but nevertheless produced a better overall total. The best models also produced an overestimated total; whereas JSNN for D2 and D2S provided underestimations and perhaps such results can be linked to the inclusion of a past discharge input which appears to have reduced the overall level of sediment output.

MLR: Abrahart and See (2007a,b) recommend the construction of a linear model to provide a standard against which the requirement for a non-linear modelling solution can be tested. If a linear model offers an acceptable solution it should be operationalised. However, it is also the case that non-linear tools could thereafter be used for the identification of neglected non-linearities (Curry and Morgan, 2003). The result for all MLR models was near identical; the addition of a second discharge input made no difference at all; the addition of a past sediment input produced a minor drop in the reported metrics. This result suggests that limited notice is being taken of anything other than present discharge. However, it should be noted that MLR produced a similar result to BPNN in terms of RMSE for D1and

a better result in terms of R-Sqd for D1, D2 and D2S. It was also better than NFNN in terms of R-Sqd for D2. The bearing of particular models on different input scenarios is thus questioned.

11.3 Flood-Event Scale Modelling

In European rivers the sediment transfer process is normally constrained by supply rather than by the ability of the river to transport material. The spatial and temporal impact of rainfall drivers on such factors produces a non-linear hydrological relationship between suspended sediment concentration and river discharge, relationships that often demonstrate hysteresis at the event level. From such understandings arises a simple proposition: if hysteresis and related scatter, are considered to be event scale properties, perhaps this scale of hydrological activities is the one at which sediment transport modelling should occur. For management purposes it is axiomatic that sediment models must be able to predict into the future; practical considerations will also require model inputs that are known or easily calculated. Two potential modelling options for operational event-level sediment estimation can be identified. The modeller could for instance attempt:

1. to perform a classification of individual flood events according to type and duration; thereafter producing and assigning to the different class groups some estimate of average sediment throughput per unit time or total sediment delivered; or
2. to develop an event-based model in which important characteristics that drive the sediment supply process are used to predict some measure of average sediment throughput per unit time or total sediment delivered. The modelling inputs in such cases could be simple variables that were selected to act as proxies for different factors that are understood to control sediment supply; the output could be mean sediment concentration per unit period. This is a simplification of the sediment supply and delivery process: if inputs are not instantaneous measurements of precipitation or discharge, and outputs are not instantaneous measurements of suspended sediment concentration, it is possible that the resultant model would be missing the highest peaks that are more challenging to predict. However, the main purpose of such activities is not process modelling per se, it is the production of an operational management solution – so this simplification can be justified.

'Option 2' is explored in this chapter and some initial experiments to develop an event-based model are presented. The flood event model was constructed on observed data sets for the Environment Agency (EA) gauging station at Low Moor on the River Tees in northern England (Fig. 11.7; EA Station No. 25009; OS Grid Ref NZ365105; latitude 54°29′21″N; longitude 1°26′31″W). It is the lowest downstream recording point on that river. The Tees Barrage was completed in 1994 and this gauge is situated at the former tidal limit. The Tees is a major river with an upstream drainage area of 1,264 km^2. Headwaters in the west cover a steep upland

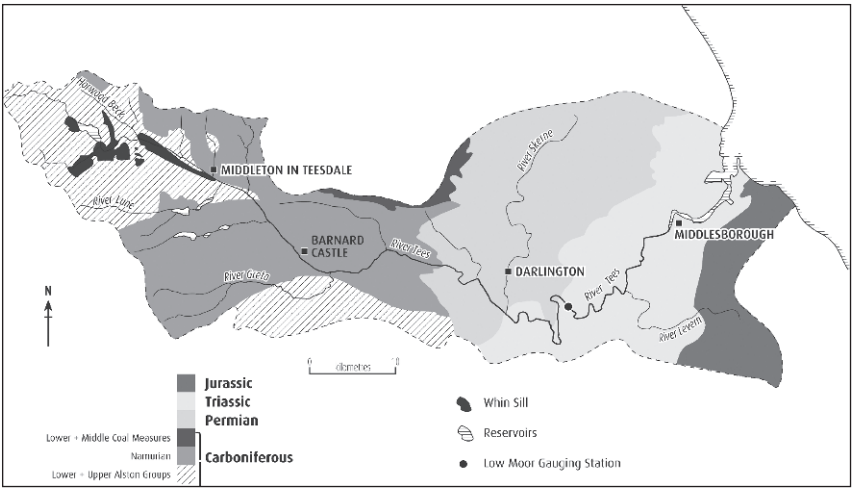


Fig. 11.7 EA gauging station at Low Moor on the River Tees in Northern England

area that is underlain by Carboniferous geology; lowlands in the east are underlain by Permian and Triassic rocks. There is a strong west–east precipitation gradient, rising to some 2,000 mm pa or more in the highlands and decreasing to 600 mm pa in the lowlands. High drainage densities in the extensive uplands, coupled with high precipitation rates, lead to a runoff-dominated regime that exhibits large and very rapid variations in discharge. Large events are responsible for much of the sediment that is delivered. Each item or feature in the above list is a major factor that, in combination with others, would be expected to exert a control on suspended sediment concentration. These factors are also the ones that are most likely to change in the future under global warming. The selection of appropriate drivers for an event-based model must be drawn from that list.

Fifteen-minute turbidity measurements were collected over a period of more than 3.5 years which preceded, included and continued after the major flood events of winter 2000–2001. Two periods were involved: (i) December 1999–April 2001 and (ii) December 2001–February 2003. The turbidity measurements were augmented by automatic pumped-sampling, during high discharge events, with samples being analysed for suspended sediment. The resultant concentration data set was used to calibrate the turbidity sensor allowing a long-term 15-minute suspended sediment concentration data set to be established. Concurrent river discharge measurements allowed sediment loads to be estimated. Prior to the winter 2000–2001 floods the river appeared to be in equilibrium with a well-constrained relationship existing between suspended sediment concentration and discharge in individual events. However, during the winter 2000–2001 floods, the relationship between suspended sediment concentration and discharge was realigned. The revised relationship exhibited much higher concentrations at the same levels of discharge – in contrast to previous observed measurements. It was hypothesised that this modification was due to the activation of fresh sediment sources, situated close to the river network, and

justified by observation of a number of river bank collapses. Post-flood measurements also indicate that the relationship drifted back towards its previous position as the new sediment sources were exhausted. Two types of hysteresis behaviour during that period can be observed in the Tees: clockwise and anticlockwise rotations. There are also several different groups of behaviour occurring within the data set. Prior to the very wet winter of 2000–2001, sediment peaks lagged discharge peaks; after that period sediment peaks lead discharge peaks; their relationship then tends back towards that occurring in pre-flood events.

For the purposes of the reported experiment, an event is defined as starting at the beginning of the day on which the rate of rise in the hydrograph exceeded $0.75\,m^3\,s^{-1}$. Mixed combinations of different catchment variables that could act as proxies for factors that are understood to control the supply of sediment were analysed. Ten inputs were thereafter selected and used to predict event mean sediment concentration ($mg\,l^{-1}$):

- Event duration in hours
- Peak discharge in $m^3\,s^{-1}$
- Total discharge in $m^3\,s^{-1}$
- Baseflow Index
- Total flow for 7 and 30 days before the event (mm; units selected so as to be consistent with catchment precipitation inputs)
- Precipitation duration in hours
- Precipitation at three rain gauges across the catchment in mm: Cow Green, Dartington and Lartington.

This selected mix of different inputs was considered to provide a good starting point. The model attempts to integrate several different types of measurement: recognised scientific drivers, calculated hydrological indices and other relevant material that is contained in raw measurement data sets. Like its counterparts this model uses simple and readily available inputs that are well understood. It also maintains the ability to differentiate between different hydrological conditions and catchment properties, which affect the supply and delivery of sediment, and offers a level of detail that could be useful for monitoring and evaluation purposes. The inclusion of a date input, expressed as a fraction of the year, was considered – but not implemented – in this initial trial. It is anticipated that different variables or different combinations could be used to develop an improved model but for the moment the main interest is in ‘proof of concept’.

TNNS (Trajan Neural Network Simulator[2]) was used to develop and implement a standard BPNN. Following removal of flood events with missing variables the data set was reduced to 86 cases; 28 occurred before the extreme floods of autumn 2000; 4 occurred during the flooding period; and a further 54 occurred subsequent to that catastrophic sequence of events. Initial testing was performed on a simple model: 10:5:1 architecture; each processing unit was connected to all processing units in each adjacent layers; a full set of initial connections was maintained throughout.

[2] http://www.trajan-software.demon.co.uk/

No transfer function is applied in the input layer and unaltered inputs are passed as direct outputs to units in the next layer. Each processing unit in the hidden layer and output layer contained a logistic transfer function that had a sigmoid operator. The data set was standardised to a common scale: 0.0–0.8. This was designed to permit extrapolation and remove possible upper-end saturation of the sigmoid transfer function situated inside the output unit.

The BPNN was thereafter trained on 30 events (28 preceding major floods/plus first two major floods) with cross-validation and early stopping operations being performed using 56 subsequent events (next two major floods plus 54 post-flood events). The network was trained for 12,000 iterations using low levels of learning rate (0.2) and momentum (0.1). Figure 11.8 shows the training and cross-validation error plots. The optimum solution was considered to have occurred at 4,000 epochs; no independent test data set was available so a full set of results is instead provided for pre-flood and post-flood data sets. Figure 11.9 provides a combined scatter plot of the training and cross-validation data sets for the 4,000-epoch model. The error between the observed and predicted values indicates that the model is better at predicting mid-range event mean concentrations, as opposed to predicting outputs at either end of the observed range, but especially in the case of two cross-validation data set higher magnitude events that were well off target. The R^2 of the training data set is 0.88; for the cross-validation data set it is 0.33. However, if the two outlier events that were so poorly predicted are not included, then this second statistic is increased to 0.66.

The two events that were not well predicted contained the highest sediment concentrations of the whole data set; whereas their concurrent discharge measurements were both situated towards the lower end of the recorded values. It is important at this stage to remember that suspended sediment concentration in the recorded events was not a direct measurement; it was estimated from a turbidity data set using

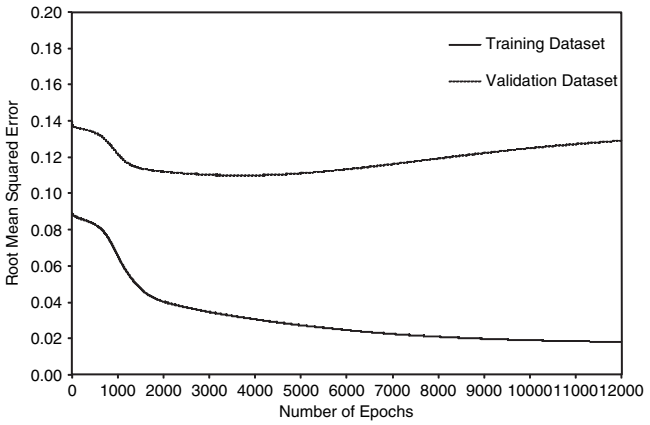


Fig. 11.8 BPNN training and cross-validation error for event scale model of Low Moor. Model selected at start of rise in error computed on cross-validation data set, i.e. epochs = 4000

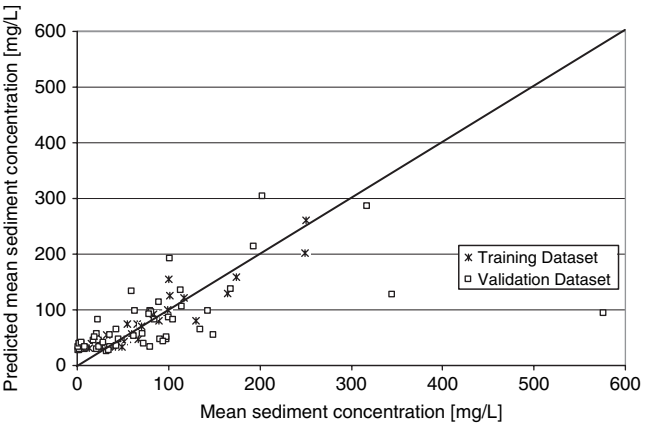


Fig. 11.9 BPNN outputs for event scale model of Low Moor

turbidity–concentration rating curves that were themselves developed on measurements related to a previous event. The concentration values are as a result somewhat less reliable in comparison to the discharge records: the two instances in question also make little or no real physical sense. The model nevertheless attempted a reasonable extrapolation to cover major events in the cross-validation data set and was able to accommodate the two major outliers in a not too unreasonable manner. It should likewise be noted that this modelling exercise provided a very severe test of neural modelling capabilities since:

- the network was trained on events of only one behaviour type; but was used to predict two different types of event, i.e. pre- and post-major flood situations;
- the number of samples in the training data set was very small;
- the model used raw material – no data cleansing operations were used to exclude events of questionable data quality; and
- inclusion of exceptional magnitudes associated with major outliers in the standardisation procedure will have handicapped the production of a decent spread of suspended sediment output records.

11.4 Final Thoughts

NN solutions can be used to model suspended sediment concentration records and it is clear that different types of solution can be developed to meet different operational requirements. The selection of an appropriate method boils down to a simple matter of selecting the right approach to support one or more particular demands. It is suggested that the future of sediment modelling for operational purposes requires a shift in mindset from the way things are currently done. The established method of producing sediment rating curves revolves around a simple generalisation that can serve many

purposes, but in the face of unprecedented climate and land use changes, it is quite unreasonable to expect such a mechanism to cope with the demands that are being placed upon it, i.e. relationships will change and are likely to be highly non-linear. It is thus argued that simple linear or non-linear generalisations will not be the best way forward. NN 'mimics' of simple linear or non-linear solutions will likewise suffer the same fate, or worse, since their power to extrapolate and venture into unfamiliar territories is poor. Perhaps this approach could support the modelling of historical annual averages for larger rivers or be applied to small basins that possess no strong hysteresis loops, or loops of different types, related to inconsistencies in sediment sourcing.

Event-based modelling offers one potential route forward. It incorporates event scale features and characteristics. It sidesteps the problems of hysteresis and scatter. The discussion that is presented in this chapter also suggests that neither discharge records nor discharge and sediment data sets will be sufficient to detect and model event behaviours. Including a range of proxy factors in an innovative neural model – appears to perform well – even predicting event mean sediment concentration for larger events with different internal behaviour. The next step in our research is to perform a scientific prototyping of events using such proxy factors, that is to test out 'Option 1'. However, perhaps events could be better modelled under situations of climate change using modelling inputs that account for climatic variability. It is also the case that other variables might be needed to account for natural or anthropogenic induced land use changes. There is much to be explored.

In overall terms the hydrological scientist or practitioner still needs to accept that neural solutions have a lot to offer; to understand the tasks that such tools are good at; and to engage in the process of discovering how to exploit the untapped opportunities that exist to do something different. Lots of modelling potential exists for mixing and matching different types of input; for building models of situations that are poorly defined; for resolving difficult situations in cases where an exact mathematical relationship is not known, etc. Numerous papers have demonstrated the development of reasonable time series models: but do such models offer realistic operational solutions? There is a clear use for simple rating curve type functions but that should not be the end of the road. There is also a clear need for more 'thinking outside the box'. The reported modelling of mean sediment concentration per event is but one example of the endless possibilities that could be explored. It would also be great to test this novel procedure on different river systems – but good event-based data sets to support such investigations are for the moment still few and far between!

References

Abrahart RJ, Heppenstall, A.J, See LM (2007) Timing error correction procedure applied to neural network rainfall-runoff modelling. *Hydrological Sciences Journal* 52(3): 1–12.

Abrahart RJ, See LM (2007a) Neural network emulation of a rainfall-runoff model. *Hydrology and Earth System Sciences Discussions* 4: 287–326.

Abrahart RJ, See LM (2007b) Neural network modelling of non-linear hydrological relationships. *Hydrology and Earth System Sciences* 11(5): 1563–1579.

Abrahart RJ, White SM (2001) Modelling Sediment Transfer in Malawi. Comparing Backpropagation Neural Network Solutions Against a Multiple Linear Regression Benchmark Using Small Data Sets. *Physics and. Chemistry of the Earth (B)* 26(1): 19–24.

Agarwal A, Singh RD, Mishra SK, Bhunya PK (2005) ANN-based sediment yield models for Vamsadhara river basin (India). *Water SA* 31(1): 95–100.

Asselman NEM (2000) Fitting and interpretation of sediment rating curves. *Journal of Hydrology* 234(3–4): 228–248.

Bellwood DR, Hughes TP, Folke C, Nyström M (2004) Confronting the coral reef crisis. *Nature* 429(24): 827–833.

Boose ER, Serrano MI, Foster DR. (2004) Landscape and Regional Impacts of Hurricanes in Puerto Rico, *Ecological Monographs* 74(2): 335–352.

Chikita KH, Kemnitz R, Kumai R (2002) Characteristics of sediment discharge in the subarctic Yukon River Alaska. *Catena* 48(4): 235–253.

Cigizoglu HK (2002a) Suspended sediment estimation and forecasting using artificial neural networks. *Turkish Journal of Engineering and Environmental Sciences* 26(1): 16–26.

Cigizoglu HK (2002b) Suspended sediment estimation for rivers using artificial neural networks and sediment rating curves. *Turkish Journal of Engineering and Environmental Sciences* 26(1): 27–36.

Cigizoglu HK (2004) Estimation and forecasting of daily suspended sediment data by multi-layer perceptrons. *Advances in Water Resources* 27(2): 185–195.

Cigizoglu HK, Kisi O (2006) Methods to improve the neural network performance in suspended sediment estimation. *Journal of Hydrology* 317(3–4): 221–238.

Coppus R, Imeson AC (2002) Extreme events controlling erosion and sediment transport in a semiarid sub-Andean Valley. *Earth Surface Processes and Landforms* 27(13): 1365–1375.

Crowder DW, Demissie M, Markus M (2007) The accuracy of sediment loads when log-transformation produces nonlinear sediment load-discharge relationships. *Journal of Hydrology* 336(3–4): 250–268.

Curry B, Morgan PH (2003) Neural networks, linear functions and neglected non-linearity. *Computational Management Science* 1(1): 15–29.

Dawson CW, See LM, Abrahart RJ, Heppenstall AJ (2006) Symbiotic adaptive neuro-evolution applied to rainfall-runoff modelling in northern England. *Neural Networks* 19(2): 236–247.

De Vries A, Klavers HC (1994) Riverine fluxes of pollutants: monitoring strategy first, calculation methods second. *European Journal of Water Pollution Control* 4(2): 12–17.

Duan N (1983) Smearing estimate: a nonparametric retransformation method. *Journal of the American Statistical Society* 78(383): 605–610.

Fenn CR, Gurnell AM, Beecroft IR (1985) An evaluation of the use of suspended sediment rating curves for the prediction of suspended sediment concentration in a proglacial stream. *Geografiska Annaler (A)* 67(1–2): 71–82.

Ferguson RI (1986) River loads underestimated by rating curves. *Water Resources Research* 22(1): 74–76.

Fransen PJB, Phillips CJ, Fahey BD (2001) Forest road erosion in New Zealand: overview. *Earth Surface Processes and Landforms* 26(2): 165–174.

Giustolisi O, Laucelli D (2005) Increasing generalisation of input-output artificial neural networks in rainfall-runoff modelling. *Hydrological Sciences Journal* 50(3): 439–457.

Hansen NC, Daniel TC, Sharpley AN, Lemunyon JL (2002) The fate and transport of phosphorus in agricultural systems. *Journal of Soil Water Conservation* 57(6): 408–417.

Hao C, Qiangguo C (2006) Impact of hillslope vegetation restoration on gully erosion induced sediment yield. *Science in China Series D: Earth Sciences* 49(2): 176–192.

Heppenstall AJ, See LM, Abrahart RJ, Dawson CW (2008) Neural Network Hydrological Modelling: An evolutionary approach. In: Abrahart RJ, See LM, Solomatine DP (eds) *Practical Hydroinformatics: computational intelligence and technological developments in water applications*. Springer-Verlag.

Hicks, DM, Gomez B (2003) Sediment Transport. In: Kondolf GM, Piégay H (eds) *Tools in Fluvial Geomorphology*. John Wiley & Sons Ltd., Chichester, UK, 425–461.

Holtschlag DJ (2001) Optimal estimation of suspended-sediment concentrations in streams. *Hydrological Processes* 15(7): 1133–1155.

Horn J, Goldberg DE, Deb K (1994) Implicit niching in a learning classifier system: Nature's way. *Evolutionary Computation* 2(1): 27–66.

Horowitz AJ (1995) *The Use of Suspended Sediment and Associated Trace Elements in Water Quality Studies*. IAHS Special Publication No. 4, IAHS Press: Wallingford, UK; 58 pp.

Horowitz AJ (2003) An evaluation of sediment rating curves for estimating suspended sediment concentrations for subsequent flux calculations. *Hydrological Processes* 17(17): 3387–3409.

Jain SK (2001) Development of integrated sediment rating curves using ANNs. *Journal of Hydraulic Engineering* 127(1): 30–37.

Jang J-SR (1993) ANFIS: Adaptive-Network-Based Fuzzy Inference System. *IEEE Transactions on Systems, Man, and Cybernetics* 23(3): 665–685.

Keyes AM, Radcliffe D (2002) *A Protocol for Establishing Sediment TMDLs*. The Georgia Conservancy: Atlanta, GA, USA; 31pp. http://www.georgiaconservancy.org/ WaterQuality/GA_CON%20QXD.pdf

Kisi O (2004) Multi-layer perceptrons with Levenberg-Marquardt training algorithm for suspended sediment concentration prediction and estimation. *Hydrological Sciences Journal* 49(6): 1025–1040.

Kisi O (2005) Suspended sediment estimation using neuro-fuzzy and neural network approaches. *Hydrological Sciences Journal* 50(4): 683–696.

Krishnaswamy J, Richter DD, Halpin PN, Hofmockel MS (2001) Spatial patterns of suspended sediment yields in a humid tropical watershed in Costa Rica. *Hydrological Processes* 15(12): 2237–2257.

Labadz JC, Butcher DP, Potter AWR, White P (1995) The delivery of sediment in upland reservoir systems. *Physics and Chemistry of the Earth* 20(2): 191–197.

Lenzi MA, Marchi L (2000) Suspended sediment load during floods in a small stream of the Dolomites (northeastern Italy). *Catena* 39(4): 267–282.

Lopes VL, Folliott PF, and Baker Jr MB (2001) Impacts of vegetative practices on suspended sediment from watersheds in Arizona. *Journal of Water Resources Planning and Management* 127(1): 41–47.

MacDonald LH, Sampson RW, Anderson DM (2001) Runoff and road erosion at the plot and road segment scales, St John, US Virgin Islands. *Earth Surface Processes and Landforms* 26(3): 251–272.

Marks SD, Rutt GP (1997) Fluvial sediment inputs to upland gravel bed rivers draining forested catchments: potential ecological impacts, *Hydrology and Earth System Sciences* 1(3): 499–508.

Merritt WS, Letcher RA, Jakeman AJ (2003) A review of erosion and sediment transport models. *Environmental Modelling & Software* 18(8–9): 761–799.

Morgan RPC (1986) *Soil Erosion and Conservation*. Longman, London.

Moriarty DE, Miikkulainen R (1998) Forming neural networks through efficient and adaptive coevolution. *Evolutionary Computation* 5(4): 373–399.

Milner NJ, Scullion J, Carling PA, Crisp DT (1981) A review of the effects of discharge on sediment dynamics and consequent effects on invertebrates and salmonids in upland rivers. *Advances in Applied Biology* 6: 153–220.

Nagy HM, Watanabe K, Hirano M (2002) Prediction of sediment load concentration in rivers using artificial neural network model. *Journal of Hydraulic Engineering* 128(6): 588–595.

Newcombe CP, Jensen JOT (1996) Channel suspended sediment and fisheries: a synthesis for quantitative assessment of risk and impact. *North American Fisheries Management* 16(4): 693–719.

Ottaway EM, Clarke A, Forrest DR (1981) *Some Observations on Washout of Brown Trout (Salmo Trutta L.) Eggs in Teesdale Streams*. Unpublished report, Freshwater Biological Association, Teesdale Unit, UK.

Østrem G (1975) Sediment transport in glacial meltwater streams. In: Jopling AV, McDonald BC (eds) *Glaciofluvial and Glaciolaustrine Sedimentation* 23: 101–122 Society of Economic Paleontologists and Mineralogists Special Publication; 101pp.

Phillips JM, Webb BW, Walling DE, Leeks GJL (1999) Estimating the suspended sediment loads of rivers in the LOIS study area using infrequent samples. *Hydrological Processes* 13(7): 1035–1050.

Potter MA (1997) *The Design and Analysis of a Computational Model of Cooperative Coevolution*. Unpublished PhD Thesis: George Mason University, Fairfax, Virginia, USA.

Rosgen DL (1996) *Applied River Morphology*. Wildland Hydrology Books, Pagosa Springs, Colorado, USA.

Rosgen DL (2001) A hierarchical river stability watershed-based sediment assessment methodology. In: *Proceedings of the Seventh Federal Interagency Sedimentation Conference*, Reno Nevada, 97–106.

Rumelhart DE, Hinton GE, Williams RJ (1986) Learning internal representations by error propagations. In: Rumelhart DE, McClelland JL (eds) *Parallel Distributed Processing: Explorations in the Microstructures of Cognition. Vol 1*. MIT Press, Cambridge, MA, USA, 318–362.

Simon A, Hupp CR (1986) Channel Evolution in Modified Tennessee Streams. In: *Proceedings of the Fourth Federal Interagency Sedimentation Conference, Las Vegas, Nevada*, 5.71–5.82

Stott T, Mount N (2004) Plantation forestry impacts on sediment yields and downstream channel dynamics in the UK: a review. *Progress in Physical Geography* 28(2): 197–240.

Sturges DL (1992) Streamflow and sediment transport responses to snow fencing a rangeland watershed. *Water Resources Research* 28(5): 1347–1356.

Thomas RB (1985) Estimating total suspended sediment yield with probability sampling. *Water Resources Research* 21(9): 1381–1388.

Thomas RB (1991) Systematic sampling for suspended sediment. In: *Proceedings of the Fifth Federal Interagency Sedimentation Conference, Advisory Committee on Water Data*, 2–17 to 2-24.

Thomas RB, Lewis J (1993) A comparison of selection at list time and time-stratified sampling for estimating suspended sediment loads. *Water Resources Research* 29(4): 1247–1256.

Thomas RB, Lewis J (1995) An evaluation of flow-stratified sampling for estimating suspended sediment loads. *Journal of Hydrology* 170(1): 27–45.

US Environmental Protection Agency (1980) *An Approach to Water Resource Evaluation of Non-point Silvicultural Sources*. EPA-600/8-80-012, Athens, GA, USA, [http://www.epa.gov/warsss/rrisc/handbook.htm]

Walling DE, Webb BW (1988) The reliability of rating curve estimates of suspended sediment yield; some further comments. In: Bordas MP, Walling DE (eds), *Sediment Budgets*. IAHS Publication No. 174, IAHS Press, Wallingford, UK, 337–350.

Walling DE (1999) Linking land use, erosion and sediment yields in river basins. *Hydrobiologia* 410(1): 223–240.

Walling DE, Fang D (2003) Recent trends in the suspended sediment loads of the world rivers. *Global Planetary Change* 39(1–2): 111–126.

Warne AG, Webb RMT, Larsen MC (2005) *Water, Sediment, and Nutrient Discharge Characteristics of Rivers in Puerto Rico, and their Potential Influence on Coral Reefs*: U.S. Geological Survey Scientific Investigations Report 2005-5206, 58pp.

White SM (2005) Sediment yield prediction and modelling. In: Anderson M. (ed) *Encyclopaedia of Hydrological Sciences*. John Wiley & Sons Ltd., Chichester, UK. doi:10.1002/0470848944.hsa089.

Part III
Models Based on Fuzzy Logic

Chapter 12
Fuzzy Logic-Based Approaches in Water Resource System Modelling

P.P. Mujumdar and S. Ghosh

Abstract Recent research in modelling uncertainty in water resource systems has highlighted the use of fuzzy logic-based approaches. A number of research contributions exist in the literature that deal with uncertainty in water resource systems including fuzziness, subjectivity, imprecision and lack of adequate data. This chapter presents a broad overview of the fuzzy logic-based approaches adopted in addressing uncertainty in water resource systems modelling. Applications of fuzzy rule-based systems and fuzzy optimisation are then discussed. Perspectives on the scope for further research are presented.

Keywords Fuzzy logic · fuzzy optimization · water resources systems

12.1 Introduction

Hydroinformatics is a cross-disciplinary field of study which includes applications of data mining, artificial intelligence (AI), expert systems, artificial neural networks (ANN), optimisation and evolutionary algorithms, fuzzy logic, grey systems theory, decision support systems (DSS), uncertainty and risk analysis to problems within the broad areas of hydraulics, hydrology and water resources. This chapter focuses on applications of fuzzy logic, a branch of hydroinformatics which deals with uncertainty due to imprecision.

The concept of "fuzzy sets" was introduced by Lotfi Zadeh in 1965 to address uncertainty due to imprecision, fuzziness or vagueness. In introducing the concept, Zadeh (1965) states, "The notion of a Fuzzy Set provides a convenient point of departure for the construction of a conceptual framework which parallels in many respects the framework used in the case of ordinary sets, but is more general than the

P.P. Mujumdar
Department of Civil Engineering, Indian Institute of Science, Bangalore, India,
e-mail: pradeep@civil.iisc.ernet.in

S. Ghosh
Department of Civil Engineering, Indian Institute of Science, Bangalore, India

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

latter and, potentially, may prove to have a much wider scope of applicability, particularly in the fields of pattern classification and information processing". Essentially, such a framework provides a natural way of dealing with problems in which the source of imprecision is the absence of sharply defined criteria of class membership rather than the presence of random variables. The possibilities of applying fuzzy set concepts in modelling water resource systems were discussed in an early paper on the subject by Hipel (1983). Fuzzy set-based models have since been developed to address imprecision in widely varying areas from atmospheric circulation (Bardossy et al., 1995) to ground water management (Guan and Aral, 2005).

A fuzzy set is a set of objects without clear boundaries or without well-defined characteristics. In contrast to a crisp set in which an element either fully belongs to the set or does not belong at all, a partial membership of an element is possible in a fuzzy set. A membership function (MF) of a fuzzy set is a function – normally represented by a geometric shape – that defines how each point in the input space is mapped to a membership value between 0 and 1. If X is the input space (e.g. possible concentrations of a water quality indicator) and its elements are denoted by x, then a fuzzy set A (e.g. a set of "high water quality") in X is defined as a set of ordered pairs:

$$A = \{x, \mu_A(x) | x \in X\} \tag{12.1}$$

where $\mu_A(x)$ is called the membership function of x in A. Thus, the membership function maps each element of X to a membership value between 0 and 1. A membership function can be of any valid geometric shape. Some commonly used membership functions are of a triangular, trapezoidal and bell shape (Kosko, 1996; Ross, 1995).

In the following sections, applications of fuzzy rule-based models and fuzzy optimisation are presented.

12.2 Fuzzy Rule-Based Modelling

A fuzzy rule system is defined as the set of rules which consists of sets of input variables or premises A, in the form of fuzzy sets with membership functions μ_A, and a set of consequences B, also in the form of fuzzy sets. Typically, a fuzzy if-then rule assumes the form:

$$\text{if } x \text{ is } A \text{ then } y \text{ is } B$$

where A and B are linguistic values defined by fuzzy sets on the variables X and Y, respectively. The "if" part of the rule "x is A" is called an antecedent or premise, and the "then" part of the rule "y is B" is called the consequence. In the case of binary or two-valued logic, if the premise is true then the consequence is also true. In a fuzzy rule, if the premise is true to some degree of membership, then the consequence is also true to that same degree. The premise and consequence of a rule can also have several parts, for example,

if x is A and y is B and z is C, then m is N and o is P, etc.

An important step in applying methods of fuzzy logic is the assessment of the membership function of a variable in various fuzzy sets. As an example, in reservoir

operation models, the membership functions required are typically those of inflow, storage, demand and release, and the corresponding fuzzy sets may be those of high inflow, low release, etc. When the standard deviation of a variable is not large, it is appropriate to use a simple membership function consisting of only straight lines, such as a triangular or a trapezoidal membership function. Kosko (1996) observed that the fuzzy controller attained its best performance when there is an overlapping in the adjacent membership functions. A good rule of thumb is that the adjacent fuzzy sets should overlap approximately 25%.

The fuzzy logic-based modelling of a reservoir operation (Panigrahi and Mujumdar 2000; Russel and Campbell, 1996; Shrestha et al., 1996) operates on an "if-then" principle, where "if" is a vector of fuzzy explanatory variables or premises such as the present reservoir pool elevation, the inflow, the demand and time of the year. The "then" is a fuzzy consequence such as release from the reservoir. In modelling the reservoir operation with fuzzy logic, the following distinct steps are followed: (a) fuzzification of inputs, where the crisp inputs such as the inflow, reservoir storage and release are transformed into fuzzy variables; (b) formulation of the fuzzy rule set, based on an expert knowledge base; (c) application of a fuzzy operator, to obtain one number representing the premise of each rule; (d) shaping of the consequence of the rule by implication; and (e) defuzzification.

Singh and Mujumdar (2002) further studied the sensitivity of reservoir operating policies derived with a fuzzy logic approach, to changes in the membership functions and to the methods of defuzzification. Considering the commonly used fuzzy membership functions such as the triangular, trapezoidal and the bell-shaped membership functions, and defuzzification methods such as the centroid, mean of maximum (MOM), largest of maximum (LOM) and the smallest of maximum (SOM), the fuzzy membership functions were tuned to reproduce as closely as possible, a long-term, steady-state operating policy derived from stochastic dynamic programming (SDP), for a case study in India. Figure 12.1 shows typical membership functions for such an application. More recently, Akter and Simonovic (2004) have dealt in detail with construction of fuzzy membership functions to represent uncertainties in penalty functions and release targets in a short-term reservoir operation problem.

Fuzzy rule-based systems are also applied in the areas of contaminant transport (Dou et al., 1999) and diffuse pollution (e.g. Binoy and Mujumdar, 2003). Dou et al. (1999) captured the underlying physical processes of solute transport by fuzzy rules. The rules are derived from a training set obtained from different test runs of the SWMS_2D (Simunek et al., 1994) model which simulates water flow and solute transport in two-dimensional variably saturated media. Fuzzy rules operate between two adjacent cells at each time step. Solute concentration of the upper cell and solute concentration difference between two adjacent cells are used as premises. For a given time step, the solute flux between the two cells is taken as the response, which is combined with the conservation of mass to update the new solute concentration for the new time step. A fuzzy inference system is used by Binoy and Mujumdar (2003) to obtain reasonable estimates of diffuse pollution from agricultural runoff using limited available information. Annual average use of herbicide per unit area, extent of herbicide applied area and herbicide application season are considered as fuzzy input sets and observed herbicide concentration at

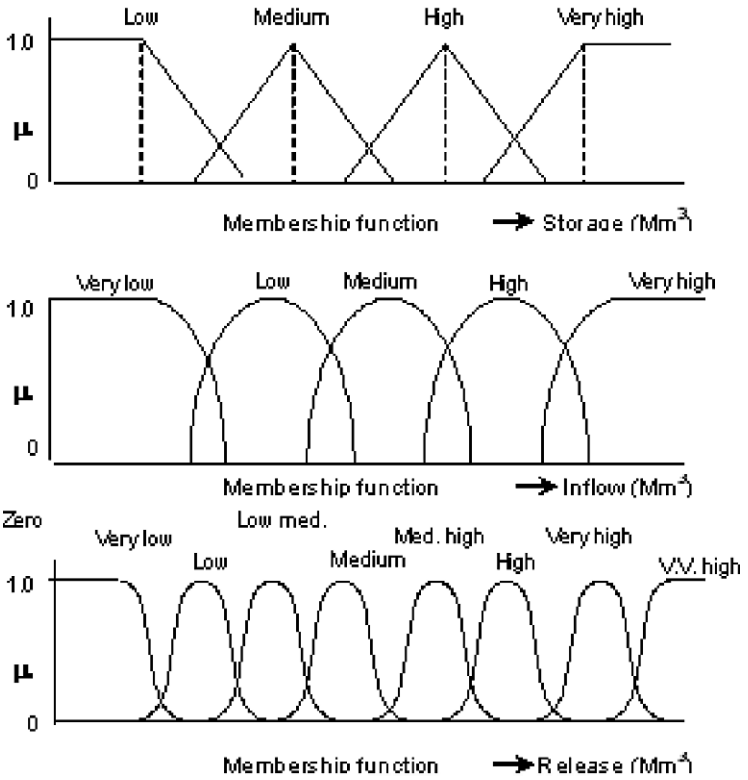


Fig. 12.1 Typical membership functions in a fuzzy rule-based model for reservoir operation

the basin outlet as the fuzzy output set. Fuzzy rules are generated from the available data sets from which combined rule bases are formed. These rules are then used for mapping the input space to the output space using a defuzzification procedure. This method learns from historical information. Other applications of fuzzy rule-based

Table 12.1 Some applications of fuzzy rule-based systems

Application	Issues addressed	Representative literature
Hydro-climatology	Classification of circulation patterns	Bardossy et al. (1995); Ozelkan et al. (1998)
Reservoir operation	Formulation of fuzzy rules for reservoir operation, implication and defuzzification	Russel and Campbell (1996); Shrestha et al. (1996); Panigrahi and Mujumdar (2000); Teegavarapu and Simonovic (1999); Fontane et al. (1997)
Water quality modelling	Diffuse pollution	Binoy and Mujumdar (2003)
	Contaminant transport	Dou et al. (1999)
Others	Ground water recharge	Coppola, Jr. et al. (2002)
	Infiltration estimation	Bardossy and Disse (1993)

systems include estimation of infiltration (Bardossy and Disse, 1993), real-time flood forecasting (Yu and Chen, 2005), classification of the atmospheric circulation pattern (Bardossy et al., 1995; Ozelkan et al., 1998) and estimation of ground water recharge (Coppola, Jr. et al., 2002). Table 12.1 provides a broad overview of applications of fuzzy rule-based systems.

12.3 Fuzzy Decisions and Fuzzy Optimisation

The concept of a fuzzy decision was first introduced by Bellman and Zadeh (1970). The imprecisely defined goals and constraints are represented as fuzzy sets in the space of alternatives. The confluence of fuzzy goals and fuzzy constraints is defined as the fuzzy decision. Considering a fuzzy goal, F, and a fuzzy constraint, C, the fuzzy decision, Z, is defined as the fuzzy set resulting from the intersection of F and C. Figure 12.2 shows the concept of a fuzzy decision. Mathematically,

$$Z = F \cap C \tag{12.2}$$

The membership function of the fuzzy decision Z is given by

$$\mu_Z(x) = \min[\mu_F(x), \mu_C(x)] \tag{12.3}$$

The solution x^*, corresponding to the maximum value of the membership function of the resulting decision Z, is the optimum solution. That is,

$$\mu_Z(x^*) = \lambda^* = \max_{x \in Z}[\mu_Z(x)] \tag{12.4}$$

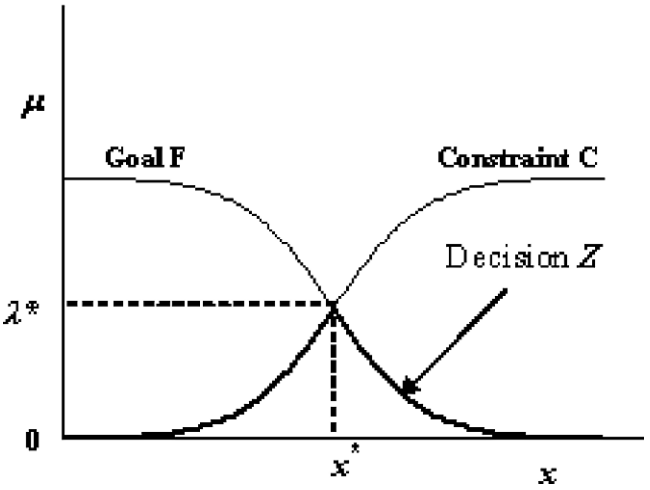


Fig. 12.2 Fuzzy decision

Goals and constraints are treated identically in fuzzy optimisation. Representing the fuzzy goals and fuzzy constraints by fuzzy sets $F_i, i = 1, 2, \ldots, n_F$ the resulting decision can be defined as

$$Z = \bigcap_{i=1}^{n_F} F_i \tag{12.5}$$

In terms of the corresponding membership functions, the resulting decision for the multi-objective problem is

$$\mu_Z(X) = \min_i [\mu_{F_i}(X)] \tag{12.6}$$

where X is the space of alternatives. The optimal solution X^* is given by

$$\mu_Z(X^*) = \lambda^* = \max_{X \in Z} [\mu_Z(X)] \tag{12.7}$$

The space of alternatives X (i.e. the decision space) is restricted by precisely defined constraints known as crisp constraints (e.g. mass balance of flows at a junction in a river network for a water allocation problem; minimum waste treatment level imposed on the dischargers by the pollution control agency for a waste load allocation problem). Incorporating these crisp constraints, $g_j(X) \leq 0$, $j = 1, 2, \ldots, n_G$, the crisp equivalent of the fuzzy multi-objective optimisation problem can be stated as follows (Zimmermann, 1978; Kindler, 1992):

$$\text{Max } \lambda \tag{12.8}$$

Subject to

$$\mu_{F_i}(X) \geq \lambda \qquad \forall i \tag{12.9}$$

$$g_j(x) \leq 0 \qquad \forall j \tag{12.10}$$

$$0 \leq \lambda \leq 1 \tag{12.11}$$

12.3.1 Application to a Water Quality Management Problem

Water quality management of a river system may be viewed as a multi-objective optimisation problem with conflicting goals of those who are responsible for maintaining the water quality of the river system (e.g. pollution control agencies), and those who make use of the assimilative capacity of the river system by discharging the waste to the water body (e.g. industries). The goal of the pollution control agency (PCA) is to ensure that the pollution is within an acceptable limit by imposing certain water quality and effluent standards. On the other hand, the dischargers prefer to make use of the assimilative capacity of the river system to minimise the waste treatment cost.

Concentration level of the water quality parameters i at the checkpoint l is denoted as C_{il}. The pollution control agency sets a desirable level, C_{il}^D, and a minimum

permissible level, C_{il}^L, for the water quality parameter i at the checkpoint l ($C_{il}^L < C_{il}^D$). The quantity of interest is the concentration level, C_{il}, of the water quality parameter, and the fraction removal level (treatment level), x_{imn}, of the pollutant. The quantities x_{imn} are the fraction removal levels of the pollutant n from the discharger m to control the water quality parameter i.

The fuzzy goal of the PCA (E_{il}) is to make the concentration level, C_{il}, of the water quality parameter i at the checkpoint l as close as possible to the desirable level, C_{il}^D so that the water quality at the checkpoint l is enhanced with respect to the water quality parameter i, for all i and l. The fuzzy goals of the dischargers F_{imn} is to make the fraction removal level x_{imn} as close as possible to the aspiration level x_{imn}^L for all i, m and n.

The membership function corresponding to the decision Z is given by

$$\mu_Z(X) = \min_{i,m,n} \left[\mu_{E_{il}}(C_{il}), \mu_{F_{imn}}(x_{imn})\right] \tag{12.12}$$

where X is the space of alternatives composed of C_{il} and x_{imn}. The corresponding optimal decision, X^*, is given by its membership function:

$$\mu_Z(X^*) = \lambda^* = \max[\mu_Z(X)] \tag{12.13}$$

The membership function for the fuzzy goal E_{il} is constructed as follows. The desirable level, C_{il}^D, for the water quality parameter i at checkpoint l is assigned a membership value of 1. The minimum permissible level, C_{il}^L, is assigned a membership value of zero. The membership function for the fuzzy goal E_{il} is expressed as

$$\mu_{E_{il}}(C_{il}) = \begin{cases} 0 & C_{il} \le C_{il}^L \\ \left[\dfrac{C_{il} - C_{il}^L}{C_{il}^D - C_{il}^L}\right]^{\alpha_{il}} & C_{il}^L \le C_{il} \le C_{il}^D \\ 1 & C_{il} \ge C_{il}^D \end{cases} \tag{12.14}$$

Using a similar argument, the membership function for the goal F_{imn} is written as

$$\mu_{F_{imn}}(x_{imn}) = \begin{cases} 0 & x_{imn} \le x_{imn}^L \\ \left[\dfrac{x_{imn}^M - x_{imn}}{x_{imn}^M - x_{imn}^L}\right]^{\beta_{imn}} & x_{imn}^L \le x_{imn} \le x_{imn}^M \\ 1 & x_{imn} \ge x_{imn}^M \end{cases} \tag{12.15}$$

These membership functions may be interpreted as the variation of satisfaction levels of the PCA and the dischargers. The indices α_{il} and β_{imn} determine the shape of the membership functions. $\alpha_{il} = \beta_{imn} = 1$ would result in linear membership functions.

The optimisation model is formulated to maximise the minimum satisfaction level, λ, in the system. The model is expressed as

$$\text{Max } \lambda \tag{12.16}$$

Subject to

$$\mu_{E_{il}}(C_{il}) \geq \lambda \qquad \forall i,l \tag{12.17}$$

$$\mu_{F_{imn}}(x_{imn}) \geq \lambda \qquad \forall i,m,n \tag{12.18}$$

$$C_{il}^{L} \leq C_{il} \leq C_{il}^{D} \qquad \forall i,l \tag{12.19}$$

$$x_{imn}^{L} \leq x_{imn} \leq x_{imn}^{M} \qquad \forall i,m,n \tag{12.20}$$

$$x_{imn}^{\mathrm{MIN}} \leq x_{imn} \leq x_{imn}^{\mathrm{MAX}} \qquad \forall i,m,n \tag{12.21}$$

$$0 \leq \lambda \leq 1 \tag{12.22}$$

The crisp constraints (12.19)–(12.21) determine the space of alternatives. The constraint (12.19) is based on the water quality requirements set by the PCA through the desirable and permissible limits of the water quality parameters i. The aspiration level and maximum acceptable level of pollutant treatment efficiencies set by the dischargers are expressed in constraints (12.20) and (12.21). The constraints (12.17) and (12.18) define the parameter λ as the minimum satisfaction level in the system. The objective is to find X^* corresponding to the maximum value λ^* of the parameter, λ. Note that the concentration levels C_{il} are related to the decision vector X through a water quality transport model. The optimum value λ^* corresponds to the maximised minimum (max–min) satisfaction level in the system. The upper and lower bounds of λ reflect two extreme scenarios in the system. The upper bound, $\lambda = 1$, indicates that all the goals have been completely satisfied and therefore represents a no-conflict scenario. The lower bound, $\lambda = 0$, indicates that at least one

Table 12.2 Some applications of fuzzy optimisation

Application	Issues addressed	Representative literature
Water resource allocation	Rationalisation of water use	Kindler (1992)
Water quality management	Imprecision in standards of PCA and goals of dischargers	Lee and Wen (1996, 1997); Sasikumar and Mujumdar (1998); Mujumdar and Sasikumar (2002); Ghosh and Mujumdar (2006b)
Reservoir operation	Multi-person multi-objective fuzzy decision making for flood control	Yu et al. (2004)
	Imprecision in crop yield response	Jairaj and Vedula (2003); Suresh and Mujumdar (2004)
	Uncertainty modelling in short-term reservoir operation	Akter and Simonovic (2004)

goal has a zero satisfaction level and therefore represents a conflict scenario. Any intermediate value of λ represents the degree of conflict that exists in the system. The fuzzy LP formulation aims at achieving a fair compromise solution by reducing the degree of conflict in the system. A low value of λ^* indicates that a conflict scenario cannot be avoided in the system. The existence of a conflict scenario in water quality management problems is due to the compound effect of the conflicting objectives of the PCA and the dischargers, and the relatively low assimilative capacity of the river network.

The solution X^* is referred to as the best compromise solution to the multi-objective optimisation problem. Many variants of this formulation have been presented in Sasikumar and Mujumdar (1998), Mujumdar and Subba Rao (2004) and Subba Rao et al. (2004).

Table 12.2 gives a broad overview of some applications of fuzzy optimisation in water resource systems.

12.4 Scope for Further Research

Applications of fuzzy logic and fuzzy set theory for modelling uncertainty in water resource systems are relatively recent. For realistic applications, simultaneous accounting of two major sources of uncertainty, viz, uncertainty due to randomness and that due to imprecision or fuzziness, in a single integrated model is useful. Developments in fuzzy systems theory have opened up the question of precision – or, indeed the lack of it – in our ability to assign probabilities to critical events. Representation of knowledge in conventional decision analysis is in the form of precisely specified probability distributions and is the same no matter how weak the information source for this knowledge is. The concept of imprecise/fuzzy probability (e.g. Tonn, 2005) addresses this problem of excessive precision, and a number of methods incorporating this concept have emerged in other fields. Applications of the fuzzy probability concept are, however, yet to be developed in the area of water resource systems.

Kindler and Tyszewski (1995) identify a fundamental problem of the application of fuzzy set theory as that related to identification of membership functions. Studies addressing this specific problem in various areas of water resources are essential. Uncertainty in the membership functions themselves may increase model output uncertainty. One way of addressing uncertainty in the membership functions is to treat the membership parameters as interval grey numbers (i.e. intervals with known lower and upper bounds but unknown distribution information), and formulate the model as a grey fuzzy programming problem (e.g. Karmakar and Mujumdar, 2004). Karmakar and Mujumdar (2006) have presented a multi-objective grey fuzzy optimisation model to minimise the system uncertainty with maximisation of the goal fulfillment level, defined as an interval grey number.

A relatively straightforward extension of the methods described in this chapter is the neuro-fuzzy approach, which is a combination of fuzzy computing and artificial

neural networks (e.g. Nayak et al., 2005; Dixon, 2004). Parameter optimisation in the fuzzy logic model is performed by a combination of backpropagation and least squares methods by Nayak et al. (2005) for a flood forecasting problem. Other recent applications of fuzzy logic in water resources include use of fuzzy clustering and fuzzy regression. Fuzzy clustering is used for regionalisation (Rao and Srinivas, 2006) and prediction of rainfall from a General Circulation Model (Ghosh and Mujumdar, 2006a). Fuzzy regression is applied in rainfall-runoff modelling (Ozelkan and Duckstein, 2001) and for regressing resistivity of a compacted soil liner with its permeability (Bardossy et al., 1990). These research areas are still nascent and a large potential exists for useful contributions. New research areas may also need to be developed for applications of possibility theory (Dubois and Prade, 1997) and fuzzy control theory (Mrozek and Plonka, 1997).

12.5 Concluding Remarks

Traditional approaches to addressing uncertainty in mathematical models have limited applications for large water resource systems with complex interactions among several critical segments. In most water resource system models, setting up of goals, limits on constraints, standards for non-violation and even objective functions introduce uncertainty due to subjectivity and imprecision. Recent interest in addressing uncertainty in water resource systems is due not only to randomness but also to imprecision, subjectivity and human judgement, and lack of data/information has lead to the use of fuzzy systems theory. This chapter provides a broad overview of some applications of the fuzzy systems techniques in typical water resource problems.

Although most fuzzy system models are not very complex computationally, fuzzy rule-based problems suffer from the curse of dimensionality, with the number of fuzzy rules increasing rapidly with an increase in the number of fuzzy variables and number of fuzzy sets used for each variable. A major research concern in the use of fuzzy systems theory is the development of appropriate membership functions for a given problem setting. This issue is analogous to, although not as amenable as, identifying an acceptable probability distribution for a random variable.

References

Akter T, Simonovic, SP (2004) Modeling uncertainties in short term reservoir operation using fuzzy sets and genetic algorithm. Hydrological Sciences Journal 49(6): 1081–1097

Bardossy A, Bogardi I, Duckstein L (1990) Fuzzy regression in hydrology. Water Resources Research 26(7): 1497–1508

Bardossy A, Disse M (1993) Fuzzy rule-based models for infiltration. Water Resources Research 29(2): 373–382

Bardossy A, Duckstein L, Bogardi I (1995) Fuzzy rule-based classification of atmospheric circulation patterns. International Journal of Climatology 15: 1087–1097.

Bellman RE, Zadeh LA (1970) Decision-making in a fuzzy environment. Management Science 17(4): B141–B164

Binoy Alias M, Mujumdar PP (2003) A fuzzy rule based model for estimating agricultural diffuse pollution. In: 7th International Specialized Conference on Diffuse Pollution and Basin Management. University College, Dublin, Ireland

Coppola Jr. EA, Duckstein L, Davis, D (2002) Fuzzy rule-based methodology for estimating monthly groundwater recharge in a temperate watershed. Journal of Hydrologic Engineering, ASCE 7(4): 326–335

Dixon B (2004) Prediction of ground water vulnerability using an integrated GIS-based neuro-fuzzy techniques. Journal of Spatial Hydrology, 4(2)

Dubois D, Prade HA (1997) Synthetic view of belief revision with uncertain inputs in the framework of possibility theory. International Journal of Approximate Reasoning 17: 295–324

Dou C, Woldt X, Bogardi I. (1999) Fuzzy rule-based approach to describe solute transport in the unsaturated zone. Journal of Hydrology 220: 74–85

Fontane DG, Gates TG, Moncada E (1997) Planning reservoir operations with imprecise objectives. Journal of Water Resources Planning and Management, ASCE 123(3): 154–162

Ghosh S, Mujumdar PP (2006a) Future Rainfall Scenario over Orissa with GCM Projections by Statistical Downscaling. Current Science 90(3): 396–404

Ghosh S, Mujumdar PP (2006b) Risk minimization in water quality control problems of a river system. Advances in Water Resources 29(3): 458–470

Guan J, Aral MM (2005) Remediation system design with multiple uncertain parameters using fuzzy sets and genetic algorithm. Journal of Hydrologic Engineering 10(5): 386–394

Hipel KW (1983) Fuzzy set techniques in decision making. Resource Management and Optimization 2(3): 187–203

Jairaj PG, Vedula S (2003) Modeling reservoir irrigation in uncertain hydrologic environment. Journal of Irrigation and Drainage Engineering, ASCE 129(3): 164–172

Karmakar S, Mujumdar, PP (2004) Uncertainty in fuzzy membership functions for a river water quality management problem. In: Liong S-Y, Phoon K-K, Babovic V (eds) Proceedings of the 6th International Conference on Hydroinformatics. World Scientific Publishing Company, Singapore, 2: 1507–1514.

Karmakar S, Mujumdar, PP (2006) An inexact optimization approach for river water quality management. Journal of Environmental management (In Print)

Kindler J (1992) Rationalizing water requirements with aid of fuzzy allocation model. Journal of Water Resources Planning Management, ASCE 18(3): 308–328

Kindler J, Tyszewski S (1995) On the value of fuzzy concepts in hydrology and water resources management. In: Kundzewicz ZW (eds) New uncertainty concepts in hydrology and water resources, International Hydrology Series, 126–132.

Kosko B (1996) Neural networks and fuzzy systems. Prentice Hall of India (Original edition: Prentice Hall Inc., Englewood Cliffs, 1992)

Lee C-H, Wen C-G (1996) River assimilative capacity analysis via fuzzy linear programming. Fuzzy Sets and Systems 79(2): 191–201

Lee C-H, Wen, C-G (1997) Fuzzy goal programming approach for water quality management in a river basin. Fuzzy Sets and Systems 89(2): 181–192

Mrozek A, Plonka L (1997) Knowledge representation in fuzzy and rough controllers. Fundamenta Informaticae 30: 299–311.

Mujumdar PP, Sasikumar K (2002) A fuzzy risk approach for seasonal water quality management of a river system. Water Resources Research 38(1): 5–1 to 5–9, DOI:10.1029/2000WR000126

Mujumdar PP, Subba Rao VVR (2004) Fuzzy waste load allocation model: A simulation-optimization approach. Journal of Computing in Civil Engineering, ASCE 18(2): 120–131

Nayak PC, Sudheer KP, Rangan DM, Ramasastri KS (2005) Short-term flood forecasting with a neurofuzzy model. Water Resources Research 41(4): W0400410.1029/2004WR003562

Ozelkan EC, Galambosi A, Duckstein L, Bardossy A (1998) A multi-objective fuzzy classification of large scale atmospheric circulation patterns for precipitation modeling. Applied Mathematics and Computation 91: 127–142.

Ozelkan EC, Duckstein L (2001) Fuzzy Conceptual Rainfall-Runoff Models. Journal of Hydrology 253: 41–68

Panigrahi DP, Mujumdar PP (2000) Reservoir operation modeling with Fuzzy Logic. Water Resources Management 14 (2): 89–109

Rao AR, Srinivas VV (2006) Regionalization of Watersheds by Fuzzy Cluster Analysis. Journal of Hydrology 318: 57–79

Ross TJ (1995) Fuzzy Logic with Engineering Applications. McGraw-Hill, New York

Russel SO, Campbell PE (1996) Reservoir operating rules with fuzzy logic programming. Journal of Water Resources Planning and Management, ASCE 122(3): 165–170.

Sasikumar K, Mujumdar PP (1998) Fuzzy optimization model for water quality management of a river system. Journal of Water Resources Planning and Management, ASCE 124(2): 79–88

Shrestha BP, Duckstein L, Stakhiv EZ (1996) Fuzzy rule based modeling of reservoir operation. Journal of Water Resources Planning and Management, ASCE 122(4): 262–269

Simunek J, Vogel T, Van Genuchten MTh (1994) The SWMS_2D code for simulating water flow and solute transport in two-dimensional variably saturated media, version 1.2. U.S. Salinity Laboratory, USDA–ARS, Riverside, California. Research Report No. 132.

Singh SK, Mujumdar, PP (2002) Fuzzy membership functions for reservoir operation modeling. In: Proceedings of the International Conference on Advances in Civil Engineering, IIT Kharagpur. 1: 216–223

Subba Rao VVR, Mujumdar PP, Ghosh S (2004) Risk evaluation in water quality management of a river system. Journal of Water Resources Planning and Management, ASCE 130(5): 411–423

Suresh KR, Mujumdar, PP (2004) A fuzzy risk approach for performance evaluation of an irrigation reservoir system. Agricultural Water Management 69: 159–177

Teegavarapu RSV, Simonovic SP (1999) Modeling uncertainty in reservoir loss functions using fuzzy sets. Water Resources Research 35(9): 2815–2823

Tonn B (2005) Imprecise probabilities and scenarios. Futures 37: 767–775

Yu PS, Chen ST (2005) Updating real-time flood forecasting using a fuzzy rule-based model. Hydrological Sciences Journal 50(2): 265–278

Yu YB, Wang BD, Wang GL, Li W (2004) Multi-person multiobjective fuzzy decision making model for reservoir flood control operation. Water Resources Management 18: 111–124.

Zadeh LA (1965) Fuzzy sets. Information and Control 8: 338–353

Zimmermann HJ (1978) Fuzzy programming and linear programming with several objective functions. Fuzzy Sets and Systems 1: 45–55

Chapter 13
Fuzzy Rule-Based Flood Forecasting

A. Bardossy

Abstract Fuzzy rules have been applied to different hydrological problems – in this chapter their application to flood forecasting is presented. They can express non-linear relationships between variables in a form that is easy to understand. Models for forecasting peak discharge and daily discharge are shown. Data available online were selected as the basis for the fuzzy rule arguments. Fuzzy rules were derived from observed flood events using a combinatorial optimisation technique – in this case simulated annealing. The methodology is applied to the Ipoly/Ipel River in northern Hungary and southern Slovakia and to the Upper Neckar catchment in south Germany. Daily forecasts using different information are presented. A split-sampling and cross-validation comparison with the Wiener filter and nearest neighbour methods shows that the fuzzy forecasts perform significantly better. An adaptive rule estimation method for possible operational forecasting is also presented.

Keywords Fuzzy rules · flood · forecasting

13.1 Introduction

The purpose of this chapter is to present the development of a flood forecasting method using fuzzy rules. Flood forecasting is one of the most important operational hydrological tasks. A large number of flood forecasting methods are available. Forecasts can be based on the following:

- Rainfall-runoff modelling combined with flood routing
- Direct use of the available information (functional or black box relationships).

Rainfall-runoff models are based on the knowledge of the most important hydrological processes in a catchment. These processes are represented in a simplified form in a mathematical model. The models describe the non-linear relationship between input and output, using reasonable approximations.

A. Bardossy
Institut für Wasserbau, Universität Stuttgart Pfaffenwaldring 61, D-70569 Stuttgart, Germany, e-mail: bardossy@iws.uni-stuttgart.de

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics*. Water Science and Technology Library 68,

Direct relationships between input and output are assessed by making assumptions about the possible relationships. Most of the traditional methods are based on linear or linearisable assumptions. Recently, methods based on techniques such as chaos theory, neural networks or fuzzy rules offer possibilities to incorporate non-linear relationships in these models.

For the application of these models, the crucial point is that the online information available is limited. Furthermore, the forecasting system should also work in the case of missing data. In addition, forecasting has to be carried out quickly, leaving no time for detailed analysis.

In rainfall-runoff modelling, limited available information leads to very uncertain forecasts. Moreover, these models are often very vulnerable to missing data.

With direct estimation, fast and robust solutions can be obtained. However, these methods often do not consider the physical background of the problem. Furthermore, in cases which differ from those used as the model basis, they might deliver non-realistic results.

The suggested approach essentially belongs to the second class but tries to incorporate as much information from the first step as possible.

The following techniques can be applied to the direct estimation of the flood peaks:

1. Linear regression (multiple)
2. Non-linear regression (multiple)
3. Nearest neighbour methods
4. Neural networks
5. Fuzzy rule-based systems

With very extensive and good quality observations, all these techniques would perform very well. Unfortunately, this is seldom the case. Linear regression is a very robust method of estimation. It requires a moderate amount of good quality data but cannot cope with missing data. Non-linear regression also works well with a moderate amount of good quality data. It is less robust than linear regression and cannot cope with missing data.

Nearest neighbour type methods are local estimators using linear or non-linear regression type models with (weighted) sub-samples of past observations for the assessment of the forecast (Abarbanel, 1995). Like the previous methods, they cannot cope with missing data. Neural networks are performing very well for large data sets (Dawson and Wilby, 1998). Unfortunately, the number of observed flood events is limited. Furthermore, the predictions cannot be explained.

Fuzzy rules can take the non-linear relationships of the variables into account. As rules, they are case dependent – thus similar to the nearest neighbour methods. However, the form of the relationship is more flexible and their advantage is that they can easily cope with missing data and data of poor quality. The derived rules can be understood easily, and expert knowledge can be incorporated.

The application of fuzzy rules for flood forecasting is described in the next sections. The methodology is applied to two forecasting cases: peak water level forecasts on the Ipoly/Ipel River and discharge forecasts on the Upper Neckar River.

13.2 Fuzzy Rules

Fuzzy sets were first introduced in Zadeh (1965) and have been applied in various fields, such as decision making and control. Basic definitions of fuzzy sets and fuzzy arithmetic can be found in Zimmermann (1984) or Dubois and Prade (1980). A brief review of the definitions of fuzzy sets, fuzzy numbers and fuzzy operations is given below.

A fuzzy set is a set of objects without clear boundaries; in contrast with ordinary sets where for each object it can be decided whether it belongs to the set or not, a partial membership in a fuzzy set is possible. Formally, a fuzzy set is defined as follows:

Definition 1. Let X be a set (universe). A is called a fuzzy subset of X if A is a set of ordered pairs:

$$A = \{(x, \mu_A(x)); x \in X \mu_A(x) \in [0,1]\} \tag{13.1}$$

where $\mu_A(x)$ is the grade of membership of x in A. The function $\mu_A(x)$ is called the membership function of A. The closer $\mu_A(x)$ is to 1 the more x belongs to A – the closer it is to 0 the less it belongs to A. If [0,1] is replaced by the two-element set $\{0,1\}$, then A can be regarded as an ordinary subset of X. In this text, for simplicity, we use the notion fuzzy set instead of fuzzy subset.

Definition 2. A fuzzy subset A of the set of real numbers is called a fuzzy number if there is at least one z such that $\mu_A(z) = 1$ (normality assumption) and such that for every real number a, b, c with $a < c < b$

$$\mu_A(c) \geq \min(\mu_A(a), \mu_A(b)) \tag{13.2}$$

This second property is known as the quasi-convexity assumption, meaning that the membership function of a fuzzy number usually consists of an increasing and a decreasing part. Any real number can be regarded as a fuzzy number with a single point support and is called a "crisp number" in fuzzy mathematics. The simplest fuzzy numbers are the triangular fuzzy numbers.

Definition 3. The fuzzy number $A = (a_1, a_2, a_3)_T$ with $a_1 \leq a_2 \leq a_3$ is a triangular fuzzy number if its membership function can be written in the form:

$$\mu_A(x) = \begin{cases} 0 \text{ if } x \leq a_1 \\ \dfrac{x - a_1}{a_2 - a_1} \text{ if } a_1 < x \leq a_2 \\ \dfrac{a_3 - x}{a_3 - a_2} \text{ if } a_2 < x \leq a_3 \\ 0 \text{ if } x > a_3 \end{cases} \tag{13.3}$$

A fuzzy rule consists of a set of premises $A_{i,j}$ in the form of fuzzy sets with membership functions $\mu_{A_{i,j}}$ and a consequence B_i also in the form of a fuzzy set:

$$\text{If } A_{i,1} \text{ AND } A_{i,2} \text{ AND} \ldots \text{AND } A_{i,J} \text{ then } B_i \tag{13.4}$$

A fuzzy rule system consists of I such rules. The applicability of a fuzzy rule for a certain case depends on the "truth grade" or "truth value" of the certain rule, and it depends also on the arguments $(a_1,\ldots,a_J)$ to which the rule is to be applied. The truth value is not a qualitative statement on the accuracy of a rule, but the degree to which the rule can be applied to the particular case. The truth value corresponding to the fulfillment of the conditions of a rule is called the degree of fulfillment (DOF) of that rule. There are several different ways of calculating the DOF. A common method used throughout this chapter is the product inference:

$$v\,(A_{i,1}\ \text{AND}\ A_{i,2}\ \text{AND}\ldots\text{AND}\ A_{i,J}) = \prod_{k=1}^{K} \mu_{A_{i,J}}(a_K) \tag{13.5}$$

Fuzzy rules are usually formulated so that more rules can be applied to the same situation expressed as a vector of premises. These rules not only have different consequences but, depending on the conditions, also have different DOFs for the given input $(a_1,\ldots,a_J)$. Therefore, the overall response which can be derived from the rule system has to be a combination of those individual rule responses, while taking into consideration the individual DOFs. There are several ways of combining the fuzzy responses of the different rules. The method used in this study is the normalised weighted sum combination of responses (B_i,v_i) for $i = 1,\ldots,I$ (where I is the number of rules) being the fuzzy set B with the membership function:

$$\mu_B(x) = \frac{\sum_{i=1}^{I} \nu_i \beta_i \mu_{B_i}(x)}{\max_u \sum_{i=1}^{I} \nu_i \beta_i \mu_{B_i}(u)} \tag{13.6}$$

where

$$\frac{1}{\beta_i} = \int_{-\alpha}^{+\alpha} \mu_{B_i}(x)\mathrm{d}x \tag{13.7}$$

This combination method delivers a fuzzy set as a response for each vector of arguments. However, in order to calculate exact values as required in models, this fuzzy response has to be replaced by a well-defined or "crisp" number. The procedure of replacing the fuzzy response with a single value is called defuzzification. There are several defuzzification methods; in this chapter, the fuzzy mean defuzzification was chosen. The fuzzy mean (or centre of gravity) of a fuzzy set A defined on the real line is the number $M(A)$ for which:

$$\int_{-\alpha}^{M(A)} (M(A)-t)\mu_A(t)\mathrm{d}t = \int_{M(A)}^{+\alpha} (t-M(A))\mu_A(t)\mathrm{d}t \tag{13.8}$$

The advantage of this combination and defuzzification method is that the calculation of the defuzzified response is extremely fast and simple. It can be

shown (Bárdossy and Duckstein, 1995) that the fuzzy mean of the combined response $M(B)$ is

$$M(B) = \frac{\sum_{i=1}^{I} v_i M(B_i)}{\sum_{i=1}^{I} v_i} \tag{13.9}$$

A detailed discussion on fuzzy rules can be found in Bárdossy and Duckstein (1995).

13.3 Fuzzy Rules for Flood Forecasting

In order to use fuzzy rules for flood forecasting, the rule arguments and the rule responses must first be selected. The rule arguments consist of variables:

1. describing the present state of the catchment (rainfall, discharge)
2. describing the recent evolution in the catchment (discharge changes)
3. forecasts of flood-relevant external variables (rainfall, temperature).

The variables should be selected according to their direct (online) availability and relevance to the forecast location. Depending on the type of problem, the rule responses can be

1. peak discharge (or water level)
2. discharge at a given temporal resolution
3. flood volumes.

The rules can be assessed directly from speculation. Unfortunately, the system knowledge (which is based on experience) is usually difficult to quantify. Therefore, methods for the assessment of rules from observational data are necessary. The following section describes a learning algorithm, which allows the derivation of rules from data directly.

13.4 Learning Fuzzy Rules Using Simulated Annealing

Given a data set T, the goal is to describe the relationship between the variables x and y using fuzzy rules.

$$T = (x_1(t), \ldots, x_J(t), y(t) \quad t = 1, \ldots, T) \tag{13.10}$$

The rule system consisting of I rules should deliver results such that the rule response R should be close to the observed value:

$$R(x_1(t), \ldots, x_J(t)) \approx y(t) \tag{13.11}$$

The fuzzy rules are formulated using predefined fuzzy sets for each variable j $\{A_{j,1},\ldots,A_{j,Kj}\}$. For flood forecasting, numerical variables are used; thus the possible fuzzy sets are chosen to be triangular fuzzy numbers. The fuzzy rules are described in the form:

$$\text{IF } x_1 \text{ is } A_{1,k_{i,1}} \quad \text{AND}\ldots x_J \text{ is } A_{J,k_{i,J}} \text{ THEN } y_1 \text{ is } B_{l_i} \tag{13.12}$$

Here i is the index of the rule. The rule system can thus be described in the form of a matrix consisting of natural numbers $k_{i,j}$

$$\boldsymbol{R} = \begin{pmatrix} k_{1,1}\ldots k_{1,J} l_1 \\ k_{I,1}\ldots k_{I,J} l_I \end{pmatrix} \tag{13.13}$$

where

$$1 \le k_{i,j} \le K_j \quad 1 \le l_i \le L$$

The goal is to find the best matrix. It is assumed that the rules are applied with a product inference and a weighted linear combination of the results. This means that for each vector $(x_1,\ldots,x_J)$, the response is calculated as

$$\hat{y} = \frac{\sum\limits_i v_i(x_1,\ldots,x_J) M(B_i)}{\sum\limits_i v_i(x_1,\ldots,x_J)} = \frac{\sum\limits_i \prod_i \mu_{A_{ij}} M(B_i)}{\sum\limits_i \prod_{i(x_1,\ldots,x_J)}} \tag{13.14}$$

These calculations are done for each element of the training set. Then the results are compared to the observed $y(t)$ values. The performance of the rule system is calculated using the observed and calculated values:

$$P = \sum_t F(\hat{y}_1(t), y(t)) \tag{13.15}$$

Typically, F can be chosen as an l_p measure:

$$F(\hat{y}_1(t), y_1(t)) = |\hat{y}(t) - y(t)|^p \tag{13.16}$$

Other performances such as a likelihood type measure or a performance related to proportional errors can also be formulated. Once one has a measure of performance, an automatic assessment of the rules can be established. This means that the goal is to find the $\boldsymbol{R}$ for which the performance is the best:

$$P(\boldsymbol{R}) \to \min \tag{13.17}$$

The number of possible different rules is

$$\prod_j K_j \times L$$

This means that the number of possible rule matrices is

$$\binom{\prod_j K_j \times L}{I}$$

which is usually a very big number. For example, in the case of a small rule system with $J = 3$ arguments with $K_j = 6$ possibilities for each of them and considering one rule response with five possibilities, the number of rule sets consisting of $I = 5$ rules would be

$$\binom{6^3 \times 5}{5} \approx 1.2 \times 10^{13}$$

Thus, one has no way of trying out each possible rule combination. Therefore, optimisation methods have to be used to find "good" rule systems.

The method selected to find the rule system $\boldsymbol{R}$ with optimal performance $P(\boldsymbol{R})$ is based on simulated annealing using the Metropolis algorithm. The algorithm is as follows:

1. The possible fuzzy sets for the arguments $A_{j,k}$ and the responses B_l are defined.
2. A rule system $\boldsymbol{R}$ is generated at random.
3. The performance of the rule system $P(\boldsymbol{R})$ is calculated.
4. An initial annealing temperature t_a is selected.
5. An element of the rule system is picked at random. Suppose the index of this element is (i,h).
6. If $h \leq J$, an index $1 \leq h^* \leq K_h$ is chosen at random and a new rule system $\boldsymbol{R}^*$ with k_{i,h^*} replacing $k_{i,h}$ is considered.
7. If $h > J$, an index $1 \leq h^* \leq L$ is chosen at random and a new rule system $\boldsymbol{R}^*$ with l_{i,h^*-J} replacing $l_{i,h-J}$ is considered.
8. The performance of the new rule system $P(\boldsymbol{R}^*)$ is evaluated.
9. If $P(\boldsymbol{R}^*) < P(\boldsymbol{R})$, then $\boldsymbol{R}^*$ replaces $\boldsymbol{R}$.
10. If $P(\boldsymbol{R}^*) \geq P(\boldsymbol{R})$, then the quantity

$$\pi = \exp\left(\frac{P(\boldsymbol{R}) - P(\boldsymbol{R})^*}{t_\mathrm{a}}\right)$$

    is calculated. With probability π, the rule system $\boldsymbol{R}^*$ replaces $\boldsymbol{R}$.
11. Steps 5–10 are repeated NN times.
12. The annealing temperature t_a is reduced.
13. Steps 5–12 are repeated until the proportion of positive changes becomes less than a threshold $\varepsilon > 0$.

The above algorithm yields a rule system with "optimal" performance. However, the rules obtained might only reflect specific features of the training data set and not the process to be modelled. This can be recognised by the number of cases for which a given rule is applied. As an alternative, the degree of fulfillment of the rules can also be considered. In order to ensure the transferability of the rules, the performance of the rule system is modified, by taking the sum of the DOFs into account.

$$P'(\boldsymbol{R}) = P(\boldsymbol{R}) \prod_i \left[1 + \max \left(0, \left(\frac{v' - \sum_t v_i(x_1(t), \ldots, x_J(t))}{v_i'} \right) \right) \right] \tag{13.18}$$

where v' is the desired lower limit for the applicability of the rules, in this case expressed by the sum of DOFs. If P' is used in the optimisation procedure, then rules which are seldom used are penalised. The degree of penalty depends on the grade to which the desired limit v' exceeds the actual sum of DOFs for a selected rule.

One of the most important things in assessing rules is the recognition that not all arguments play an important role for each case. The role of an argument varies according to the other arguments. This means that one has to consider the membership function $\mu(x) = 1$ among the selected possibilities for each argument. This makes it possible to formulate rules using only some of the arguments. This is an advantage of fuzzy rule systems, as a functional representation always considers all arguments.

Partial knowledge can be incorporated into the rule system by fixing some elements of the matrix $\boldsymbol{R}$. These fixed elements are not altered randomly in the algorithm. Thus rules can be fixed, or rules of given structure can be identified.

Missing data or fuzzy data in the training set can also be considered. For each missing value, a membership function identical to 1 is chosen. For fuzzy data, the corresponding membership function is considered. The DOF is evaluated using the fuzzy input as

$$\mu_{A_{i,k}}(\hat{x}_k) = \max_x \min \left(\mu_{A_{i,k}}(x), \mu_{\hat{x}_k}(x) \right) \tag{13.19}$$

This means for each corresponding argument, the membership 1 is assumed for missing data.

13.5 Application

The application of the methodology is presented for two case studies. In the first case, flood peaks were forecasted; in the second, forecasts were done on all days where the daily discharge exceeded a certain pre-defined level.

13.5.1 Ipoly/Ipel

The first case study area is the Ipoly/Ipel catchment, a sub-catchment of the Danube River situated in northern Hungary and southern Slovakia. The catchment has an area of 5010 km^2. There are several gauges on the river. These are

1. Holisa (685 km^2) distance from the outlet: 143 km
2. Nógrádszakál (1850 km^2)
3. Balasagyarmat (2747 km^2)
4. Visk (4687 km^2)
5. Ipolytölgyes (5010 km^2).

Table 13.1 Cross-validation performance of fuzzy rule systems with different number of rules for the forecast of peak water levels at Ipolytölgyes

	Mean error (cm)	Mean squared error (cm)	Correlation
8 rules	−0.17	41.82	0.81
9 rules	−1.24	40.89	0.82
10 rules	−3.13	42.80	0.79

Other stations and precipitation measurement locations are also situated in the catchment. Unfortunately, only data (water levels) from the five gauges listed above are available online.

For this catchment, a forecast of the peak water levels was obtained using fuzzy rules. Water levels of 65 flood events from the time period 1990–1998 were used to train the model. Input variables of the model were

- Peak water level at Holisa
- Water level at Nógrádszakál at the time of peak water level at Holisa
- Water level at Ipolytölgyes at the time of peak water level at Holisa.

The output was the peak water level at Ipolytölgyes. The peak in Holisa is 12–24 h before that in Ipolytölgyes. The mean peak water level was 212 cm, the minimum 124 cm and the maximum 467 cm.

Different numbers of fuzzy rules were assessed. A cross-validation was conducted in order to evaluate the performance. This means that for each event, rules were assessed from all other flood events and then applied to the selected event. This way the real performance of the model could be assessed. Table 13.1 shows the performance of different rule-based systems using a different number of rules. The table shows that the performance of the rule system does not improve by adding new rules. In fact more rules mean that the system might capture features of non-typical events. This leads to a worse performance on independent data sets.

13.5.2 Neckar

Data used as input for the fuzzy rules were

- Daily discharge of the previous two days at Plochingen
- Daily discharge of the previous two days at Horb
- Areal precipitation of the previous day
- Precipitation forecast ($>$ 5mm or not)

Flood events from 20 years (1961–1980) were used for the assessment of the rules. The partial series of the upper 8% of the data were only used since the goal was to predict flood discharge. The rules were then applied to the 10 year period 1981–1990. Table 13.2 shows the statistics of the selected flood events.

Table 13.2 Statistics of the selected flood events on the Neckar catchment

	Mean (m^3/s)	Standard deviation	Maximum (m^3/s)
1961–1980	113	70	1031
1981–1990	120	76	761

Table 13.3 Performance of the different methods in predicting daily discharge in the Neckar catchment

	Linear regression	Nearest neighbour	Fuzzy rules
Mean error (m^3/s)	−1.84	−2.00	−5.78
Mean squared error (m^3/s)	61.57	59.56	51.24
Correlation	0.60	0.63	0.76

In order to compare the performance of the method with other techniques, the same data were used for a forecast using the Wiener filter and a nearest neighbour method (Abarbanel, 1995). Table 13.3 shows the results. The fuzzy forecasts are based on eight rules. All values are calculated for the cross-validation case.

The performance of the fuzzy rules is far better than that of the other two methods. Note that the low correlation is due to the fact that only the partial series containing the flood events was used in the forecasting. The Wiener filter is a linear estimator. As the rainfall-runoff process is highly non-linear, its relatively poor performance is not surprising. The nearest neighbour method uses only a subset of similar flood events for the forecast, and thus the estimator is only locally linear. Fuzzy rules can be regarded as a blend of the previous methods – a non-linear forecast using rules instead of similar data.

13.6 Summary and Conclusions

In this chapter, the applicability of fuzzy rules for flood forecasting was investigated. The rules were assessed from observed flood events using a simulated annealing algorithm. The performance of the models was defined as the mean squared error of the prediction. The suggested methodology was applied to two catchments. The objective in the first case was the estimation of the peak water levels, in the second the mean daily discharge. A more detailed forecast using hourly data is also possible.

The model was compared to the Wiener filter and a nearest neighbour method, and performed considerably better than both of these. The model could be applied using a non-symmetric loss function expressing the worst consequences of a possible underestimation of flood peaks. The method is based on observed data without any preprocessing. Thus, it can also be used in an adaptive way.

References

Abarbanel HDI (1995) Analysis of Observed Chaotic Data. Springer-Verlag: New-York Berlin, Heidelberg.

Bárdossy A, Duckstein L (1995) Fuzzy Rule-Based Modeling with Applications to Geophysical, Biological and Engineering Systems. CRC Press: Boca Raton.

Dawson CW, Wilby R (1998) An artificial neural network approach to rainfall runoff modelling. Hydrological Sciences Journal 43: 47–66.

Dubois D, Prade H (1980) Fuzzy Sets and Systems. Theory and Applications. Academic Press: New York.

Zadeh, L (1965) Fuzzy Sets. Information and Control 8: 338–353.

Zimmermann HJ (1984) Fuzzy Set Theory – and its Applications. Kluwer Nijhoff Publishing: Boston Dordrecht Lancester.

Chapter 14
Development of Rainfall–Runoff Models Using Mamdani-Type Fuzzy Inference Systems

A.P. Jacquin and A.Y. Shamseldin

Abstract This study explores the application of Mamdani-type fuzzy inference systems (FIS) to the development of rainfall–runoff models operating on a daily basis. The model proposed uses a Rainfall Index, obtained from the weighted sum of the most recently observed rainfall values, as input information. The model output is the daily discharge amount at the catchment outlet. The membership function parameters are calibrated using a two-stage constrained optimization procedure, involving the use of a global and a local search method. The study area is the Shiquan-3 catchment in China, which has an area of 3092 km^2 and is located in a typical monsoon-influenced climate region. The performance of the fuzzy model is assessed through the mean squared error and the coefficient of efficiency R^2 performance indexes. The results of the fuzzy model are compared with three other rainfall–runoff models which use the same input information as the fuzzy model. Overall, the results of this study indicate that Mamdani-type FIS are a suitable alternative for modelling the rainfall–runoff relationship.

Keywords Mamdani fuzzy inference systems · rainfall–runoff model

14.1 Introduction

Fuzzy inference systems (FIS) are data-driven non-linear input–output models that describe the operation of a real system using a set of fuzzy rules. Each fuzzy rule m is a proposition of the form

$$\text{IF } (x_1 \text{ is } A_{1,m}) \text{ AND } (x_2 \text{ is } A_{2,m}) \text{ AND} \cdots \text{AND } (x_K \text{ is } A_{K,m}) \text{ THEN } y \text{ is} \ldots \quad (14.1)$$

A.P. Jacquin
Departamento de Obras Civiles, Universidad Técnica Federico Santa María, Casilla 110-V, Valparaíso, Chile, e-mail: alejacquin@yahoo.com

A.Y. Shamseldin
Department of Civil and Environmental Engineering, The University of Auckland, Private Bag 92019, Auckland, New Zealand, e-mail: a.shamseldin@auckland.ac.nz

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

expressing the relationship between K input variables $x_1, x_2, \ldots, x_K$ and the output y, for a particular region of the input space. The terms $A_{k,m}$ in the antecedents of the rules (IF parts) represent fuzzy sets (Zadeh, 1965), used to partition the input space into overlapping regions. The structure of the rule consequents (THEN parts) depends on the type of FIS under consideration. In the case of Mamdani-type FIS (Mamdani, 1974), the rule consequents have the form

$$y \text{ is } B_m \tag{14.2}$$

where B_m are fuzzy sets in the output space. Given a particular input $(x_1, x_2, \ldots, x_K)$, the inference mechanisms of fuzzy logic (Zadeh, 1975) are used to obtain the output of each rule and the overall response of the FIS. The number of rules needed to adequately represent the real system depends on the complexity of its input–output relationship. The resolution of the model can generally be improved by using a finer partition of the input space, at the cost of increasing the number of fuzzy rules and hence the complexity of the model.

The major strength of FIS over other data-driven modelling techniques is that FIS represent knowledge about the system being modelled in a way that can be easily interpreted by humans. Each rule m in a Mamdani-type FIS expresses a tendency for input vectors $(x_1, x_2, \ldots, x_K)$ in the region described by the fuzzy sets $A_{1,m}, A_{2,m}, \ldots, A_{K,m}$ to be associated with outputs y in the region defined by the fuzzy set B_m. Furthermore, knowledge provided by experts can be incorporated into the model in a natural and transparent way, by translation of knowledge into fuzzy rules. For instance, one of the rules describing the operation of a sluice gate that regulates the water level upstream in a channel could be "IF (*water level* is HIGH) THEN (*sluice gate position* is HIGH)". Finally, FIS are very flexible modelling tools as their architecture and the inference mechanisms can be adapted to the given modelling problem.

One of the main weaknesses of FIS is the *curse of dimensionality* (Kosko, 1997). The curse of dimensionality refers to the situation where the number of fuzzy rules that is necessary to model the input–output relationship increases exponentially with the number of inputs. The use of too many rules would result in a non-parsimonious model which can be very difficult to calibrate. The number of necessary rules can be substantially reduced if clustering algorithms are used to detect the general trends of data and the rules are allocated to the relevant regions of the input–output space (see e.g. Babuska and Verbruggen, 1997; Delgado et al., 1998).

Although FIS have been widely applied in areas as diverse as control and decision making, their use in the field of hydrological sciences is rather limited. Previous applications of FIS in hydrology include modelling the interdependence between global circulation and precipitation (Galambosi et al., 1998; Pongracz et al., 2001), flow routing (See and Openshaw, 2000; Chang et al., 2001; Bazartseren et al., 2003), conceptual rainfall–runoff modelling (Hundecha et al., 2001), and river flow forecasting with time series models (Nayak et al., 2004, Gautam and Holz, 2001; Vernieuwe et al., 2004). This study explores the application of Mamdani-type FIS to the development of rainfall–runoff models operating on a daily basis, using a systems-based approach. The study area is the Shiquan-3 catchment in west China.

This paper is structured as follows. Section 14.2 explains the general structure of Mamdani-type FIS. Section 14.3 describes the structure of the proposed fuzzy rainfall–runoff model. Section 14.4 gives a brief description of the study area and discusses the results of applying the model to the field data. Finally, the conclusions of this study are given in Sect. 14.5.

14.2 Mamdani-Type Fuzzy Inference Systems

Fuzzy sets are a generalization of the classical concept of set theory. While membership in a classical set is defined in a binary manner (either non-membership or full membership), membership in a fuzzy set is a question of degree. Each fuzzy set $A_{k,m}$ in (14.1) is described by its membership function $\mu_{A_{k,m}}$, which evaluates the degree of membership (ranging from zero to one) of any value x_k in the fuzzy set. The membership value $\mu_{A_{k,m}}(x_k)$ is equal to unity if the membership of x_k in the fuzzy set $A_{k,m}$ is complete, and it is equal to zero if x_k does not belong to the fuzzy set $A_{k,m}$. Thus, the feasible range of each input variable x_k is divided into a series of fuzzily defined and overlapping intervals $A_{k,m}$. Similarly, the fuzzy sets B_m are described by membership functions μ_{B_m}.

Several types of membership function can be used to describe the fuzzy sets in the rule system. Gaussian-type membership functions given by

$$\mu(z) = \exp\left[-\frac{(z-c)^2}{2\sigma^2}\right] \tag{14.3}$$

are a common choice (Chang et al., 2001; Gautam and Holz, 2001). In this case, each membership function has two parameters, namely the centre c and the spread σ. The use of Gaussian membership functions facilitates the theoretical analysis of the FIS performance, because of the simplicity of their analytical expression. In addition, the fact that Gaussian membership functions are continuously differentiable allows one to obtain differentiable model response surfaces and hence, under certain conditions, a derivative-based algorithm can be used to calibrate the model. Finally, as the support of Gaussian membership functions is infinite, they are more suitable for modelling systems with unbounded input spaces than other membership function types. Asymmetrical Gaussian membership functions are more flexible than the symmetrical Gaussian membership functions (14.3), as they have a different spread in their left and right limbs. Their analytical expression is given by

$$\mu(z) = \begin{cases} \exp\left[-\dfrac{(z-c)^2}{2\sigma_{\text{left}}^2}\right], z \le c \\ \exp\left[-\dfrac{(z-c)^2}{2\sigma_{\text{right}}^2}\right], z > c \end{cases} \tag{14.4}$$

The centres and spreads of the membership functions are the parameters of the model that are estimated by calibration.

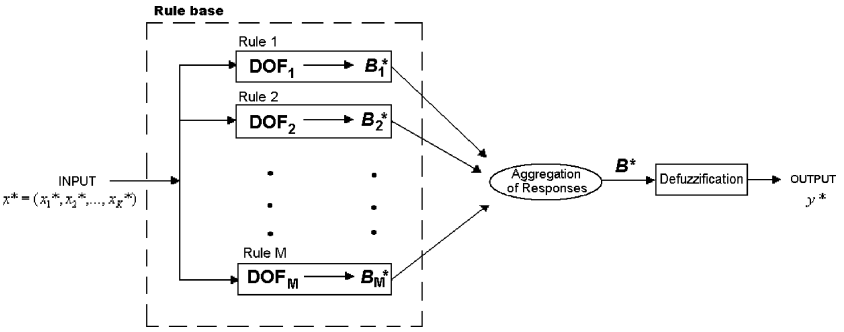


Fig. 14.1 Functioning of a multiple-input single-output Mamdani-type FIS

Figure 14.1 shows a schematic diagram of the functioning of a typical multiple-input single-output Mamdani-type FIS. The first stage in the inference process of a TSK fuzzy model is the calculation of the degree of fulfilment of each rule, evaluating the degree to which the input vector belongs to the region of the input space defined by the antecedent fuzzy sets. Each degree of fulfilment DOF_m is next combined with the corresponding consequent fuzzy set B_m, using an implication operator (logical connective THEN), to obtain the rule-implied fuzzy set $B_m{}^*$ (i.e. the response of the rule). The overall response of the FIS is a fuzzy set B^*, obtained from the aggregation of the individual rule responses $B_m{}^*$. Finally, the fuzzy response B^* of the FIS is converted into a crisp (real) number by a process called defuzzification.

The degree of fulfilment is normally evaluated using a t-norm (see Piegat, 2001), such as the minimum t-norm or the algebraic product, in the role of the AND connective. If the algebraic product t-norm is used in the role of the AND connective, the degree of fulfilment of each rule can be expressed as

$$\mathrm{DOF}_m(x) = \mu_{A_{1,m}}(x_1) \cdot \mu_{A_{2,m}}(x_2) \cdot \cdots \cdot \mu_{A_{K,m}}(x_K) \tag{14.5}$$

If Gaussian (symmetrical or asymmetrical) membership functions are used, to model the antecedent fuzzy sets, the degree of fulfilment of a rule is never zero. Therefore, each time an input vector is presented to the FIS, all the rules are activated (even if only to a very small degree) and thus all of them contribute to the overall model output.

Each fuzzy rule m describes an existing association between inputs in the region defined by the multi-dimensional antecedent fuzzy set (with membership function DOF_m) and output values in the region of the output space defined by the output fuzzy set B_m (with membership function μ_{B_m}). This association can be mathematically described as a new fuzzy set R_m in the input (x)–output (y) space, with membership function

$$\mu_{R_m}(x, y) = I(\mathrm{DOF}_m(x), \mu_{B_m}(y)) \tag{14.6}$$

where I represents an implication operator modelling the logical connective THEN. A number of operators have been proposed for this purpose (see Cordón et al., 1997,

2000), but most engineering applications of fuzzy inference systems use a t-norm, in particular the minimum t-norm or the algebraic product, as implication operators. If the algebraic product is used as the implication operator, the membership function of the fuzzy set R_m is given by

$$\mu_{R_m}(x,y) = \mathrm{DOF}_m(x) \cdot \mu_{B_m}(y) \tag{14.7}$$

For a given input $x^* = (x_1{}^*, x_2{}^*, \ldots, x_K{}^*)$, this membership function becomes a function of only the output variable y, thus obtaining an implied fuzzy set $B_m{}^*$ in the output space. The membership function of the rule-implied fuzzy set is

$$\mu_{B_m{}^*}(y) = \mathrm{DOF}_m(x^*) \cdot \mu_{B_m}(y) \tag{14.8}$$

The overall FIS response B^* is obtained from the aggregation of the individual rule responses $B_m{}^*$. There are several possible choices of response aggregation method (see e.g. Piegat, 2001), but the arithmetic sum of the individual membership function values (selected in this study) is a convenient choice. In this case, the membership function of the fuzzy response B^* is given by

$$\mu_{B^*}(y) = \sum_{m=1}^{M} \mu_{B_m{}^*}(y) \tag{14.9}$$

where M is the number of rules in the FIS. The advantage of the arithmetic sum over other operators is that the arithmetic sum explicitly considers the contributions of agreeing rule outputs, through the summation of membership values corresponding to outputs y in the region of agreement. The main drawback of using the arithmetic sum for combining the rule responses is that the membership values $\mu_{B^*}(y)$ are not guaranteed to be smaller than one. However, this is not a serious problem from the point of view of the practical operation of the FIS.

Finally, a real number y^*, in some sense representative of the fuzzy set B^*, is selected as the output of the FIS, through a process called defuzzification. Several defuzzification methods have been proposed in the literature (see e.g. Cordón et al., 1997, 2000), one of which is the centroid of the membership function μ_{B^*}. As demonstrated by Bárdossy and Duckstein (1995), small changes in the membership function μ_{B^*} produce small variations in the location of its centroid. Therefore, if the membership functions in the antecedents are continuous, the fuzzy logic operators (AND, implication, and aggregation) are continuous and the rule base is complete, and the FIS response surface is continuous. This property of the centroid defuzzifier makes it suitable for modelling real systems with continuous responses. Nevertheless, the centroid defuzzifier has some disadvantages. The main disadvantage of the method is its generally high computational cost, as it involves numerical integration of irregular functions. Another problem is that if only one rule is activated, the defuzzified output becomes insensitive to changes in the input variables (Piegat, 2001).

Combining the algebraic product as implication operator, the arithmetic sum for rule response aggregation, and centroid defuzzifier, the FIS final output results can be calculated from

$$y^* = \frac{\sum_{m=1}^{M} \mathrm{DOF}_m(x)\mathrm{Area}_m b_m}{\sum_{m=1}^{M} \mathrm{DOF}(x)\mathrm{Area}_m} \tag{14.10}$$

where Area_m is the area under the membership function μ_{B_m} and b_m is its centroid. These quantities are constants of the model, i.e. they do not change as the input vector varies. The simplicity of the latter expression for the final FIS output is a result of the specific choice of fuzzy logic operators. Other combinations of fuzzy logic operators lead to more complicated expressions, usually without an explicit analytical solution. From (14.10), it is evident that the maximum possible FIS output is equal to the maximum value b_m in the rule system.

14.3 Proposed Fuzzy Model

The model proposed is a single-input single-output FIS, intended to simulate the rainfall–runoff relationship on a daily basis. The fuzzy logic operators used during the inference process are algebraic product as implication operator, arithmetic sum for the aggregation of rule responses, and centroid defuzzification.

In order to reduce the number of rules necessary to model the rainfall–runoff relationship, the number of inputs must be kept to a minimum. For this purpose, the output of the simple linear model (Nash and Foley, 1982) is used to construct a Rainfall Index RI, which is the only input variable of the model. This Rainfall Index RI is intended to reflect the most recent precipitation forcing and, to some extent, the current moisture condition of the catchment. The analytical expression for the Rainfall Index is

$$\mathrm{RI}_i = G^a \cdot \sum_{j=1}^{L} P_{i-j+1} \cdot {h_j}^a \tag{14.11}$$

where RI_i represents the Rainfall Index value at time step i, P_j is the rainfall measurement at time step j, L is the memory length of the catchment, G^a is the gain factor of the simple linear model (SLM), and ${h_j}^a$ is the jth ordinate of the discrete pulse response function of the SLM such that

$$\sum_{j=1}^{L} {h_j}^a = 1 \tag{14.12}$$

The discrete pulse response ordinates ${h_j}^a$ of the SLM are obtained in a parametric form, using the gamma distribution model of Nash (1957), as in the work of Shamseldin (1997). The model output is the daily discharge amount Q at the catchment outlet.

Prior to the application of the fuzzy models, the data are normalized with respect to their maximum value during the calibration period, defining the normalized Rainfall Index RI^n and the normalized discharge Q^n. This transformation eliminates scale

differences between catchments, as the input and the output data are contained in the interval [0,1]. Following normalization, the general structure of the fuzzy rules is

$$\text{IF}(\text{RI}^{\text{n}} \text{ is } A_{\text{m}}) \text{ THEN } Q^{\text{n}} \text{ is } B_m \tag{14.13}$$

As the outputs of the rule consequents are given in terms of the normalized variable Q^{n}, a denormalization step is necessary to convert them into non-normalized discharge values:

$$Q = Q^{\text{n}} \cdot Q_{\max} \tag{14.14}$$

where $Q_{\max}$ is the maximum discharge during the calibration period. The maximum possible model output is equal to the maximum value of the product $b_m Q_{\max}$ encountered in the rule system. This maximum possible output can be either smaller or greater than the maximum discharge during the calibration period $Q_{\max}$, depending on whether the maximum value of b_m in the rule system is greater or smaller than unity. Similarly, the minimum discharge that the model is able to predict is equal to the minimum value of the product $b_m Q_{\max}$.

14.4 Application

The study area corresponds to the Shiquan-3 catchment in west China, located upstream of Wuhouzhen station in the Han River (the largest tributary of the Yangtze). The Shiquan-3 catchment has an area of 3092 km^2, mountainous topography, and mixed forest natural vegetation. The area has a semiarid climate, influenced by summer monsoons. Figure 14.2 shows the seasonal variation of rainfall, evaporation, and daily discharge amounts. Most of the rainfall occurs during summer, while the highest evaporation rates occur during the period June–August. The seasonal variation of discharge exhibits a similar pattern to that of rainfall. The data used in the study consist of daily mean areal rainfall and daily mean discharge at the catchment outlet (expressed in equivalent depths of water) during the period 1973–1980. The first 6 years of the record are used as the calibration period, while the remaining data are used for verification.

Asymmetrical Gaussian membership functions are used for the antecedent fuzzy sets, while Gaussian-type membership functions are used to model the consequent fuzzy sets. The objective function to be minimized during calibration of the model is the mean squared error. The membership function parameters are calibrated using a two-stage constrained optimization procedure, involving the sequential use of a global and a local search method. Attempts were made to obtain a first approximation for the location of the membership functions using a variety of clustering algorithms, such as the fuzzy c-means algorithm (Bezdek, 1981) and the Gustafson–Kessel algorithm (Gustafson and Kessel, 1979). However, it was observed that better results were obtained with the chosen optimization methodology. The number of rules adopted in this study is three, because preliminary tests carried out by the

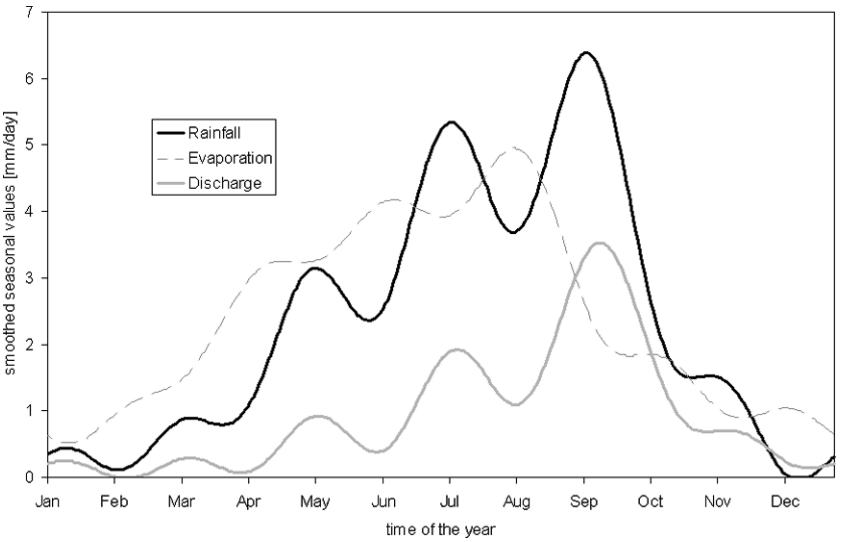


Fig. 14.2 Seasonal variation of rainfall, evaporation, and discharge in the Shiquan-3 catchment, smoothed by Fourier harmonic analysis

authors indicated that a further increase in the number of rules did not significantly improve the performance of the FIS.

Figure 14.3 shows the optimized antecedent and consequent fuzzy sets. The fact that the leftmost consequent fuzzy set has positive membership values for discharge entries smaller than zero does not mean that the model may eventually predict negative discharge values. The centre of this fuzzy set, determining the minimum discharge that the model is able to produce, is in fact very small having a value close to zero. Figure 14.4 shows how the model response is affected by a lower bound of about 0.4[mm]. Similarly, the centre of the rightmost fuzzy set gives the upper bound for the discharge estimated by the model, which Fig. 14.4 shows to be nearly 39[mm].

Figures 14.5 and 14.6 show the observed and the fuzzy model estimated discharge hydrographs for 1 year of the calibration period (1975) and 1 year of the verification period (1980). There is a tendency in the model response to underestimate the medium- and high-range discharges. As shown in Fig. 14.4, the model response function is normally below the observed discharge in the range above 15[mm]. However, no persistent trend in the sign of the model errors can be observed in the low flow ranges, with the exception of the inability of the model to predict discharge equal to zero.

The results of the fuzzy model are compared with those from three other rainfall–runoff models, namely the SLM, the nearest neighbour linear perturbation model (NNLPM) of Shamseldin and O'Connor (1996), and the neural network model N1 previously proposed by Shamseldin (1997), all of which use the same input information as the fuzzy model proposed here. The performance of the models was evaluated using the R^2 efficiency criterion of Nash and Sutcliffe (1970) given by

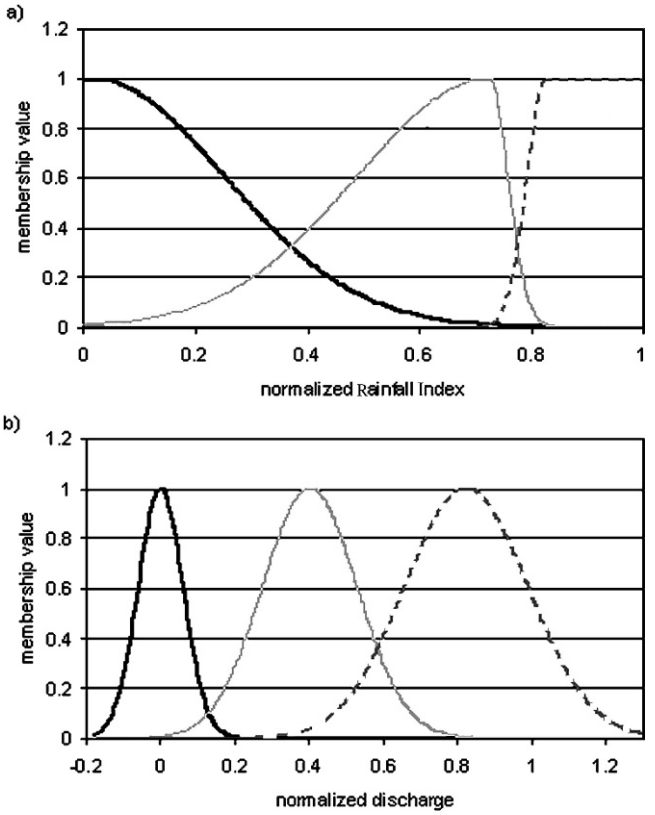


Fig. 14.3 Optimized fuzzy sets for (**a**) normalized Rainfall Index and (**b**) normalized discharge

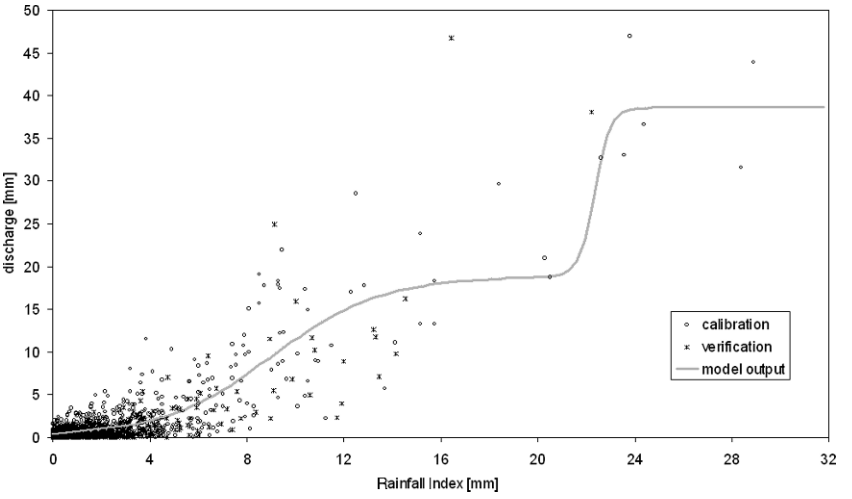


Fig. 14.4 Fuzzy model response function

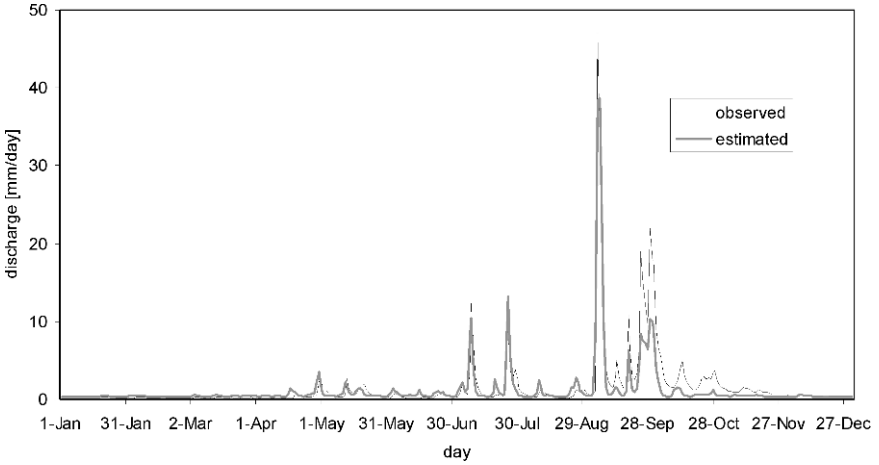


Fig. 14.5 Observed and fuzzy model estimated discharges during 1975 (calibration period)

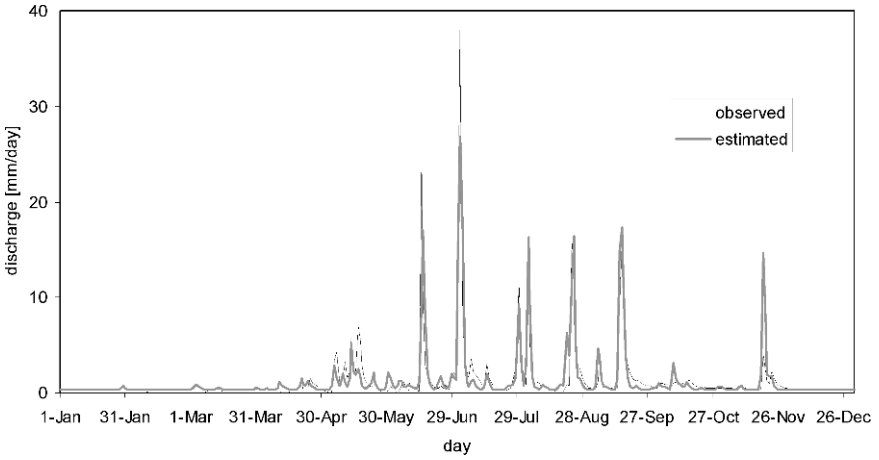


Fig. 14.6 Observed and fuzzy model estimated discharges during 1980 (verification period)

$$R^2 = \frac{\mathrm{MSE}_0 - \mathrm{MSE}}{\mathrm{MSE}_0} \times 100\% \tag{14.15}$$

where MSE_0 is the mean of the squares of the differences between the observed discharges and the long-term mean during the calibration period and MSE is the mean squared error. Table 14.1 summarizes the results obtained by the fuzzy model and the benchmark models, during the calibration and the verification period. The performance of the fuzzy model was significantly better than that of the naive SLM, where the output is used as an input variable by the fuzzy model. The NNLPM obtained the highest efficiency values, both in calibration and verification. In particular,

Table 14.1 Summary of results of the fuzzy model and the benchmark models

Model	Calibration			Verification		
	MSE_0	MSE	R^2 (%)	MSE_0	MSE	R^2 (%)
SLM	8.697	2.523	70.99	8.245	4.033	51.09
NNLPM		1.289	85.18		2.390	71.01
N1		1.619	81.38		2.717	67.04
Fuzzy model		1.548	82.20		3.076	62.69

the performance of the NNLPM during the verification period is notably better than that of the fuzzy model. The neural network model N1 and the fuzzy model have similar efficiency values.

14.5 Summary and Conclusions

This study explores the application of Mamdani-type FIS to the development of rainfall–runoff models operating on a daily basis, using a systems-based approach. The model proposed uses a Rainfall Index, obtained from the weighted sum of the most recently observed rainfall values, as input information. The model output is the daily discharge amount at the catchment outlet. Preliminary results obtained in the Shiquan-3 catchment (west China) are presented.

The performance of the fuzzy model is assessed through the coefficient of efficiency R^2, in both the calibration and the verification period. The results of the fuzzy model are compared with those from three other rainfall–runoff models, namely the simple linear model, the nearest neighbour linear perturbation model, and a neural network model, all of which use the same input information as the fuzzy model. Overall, the results of this study indicate that Mamdani-type FIS are a suitable alternative for modelling the rainfall–runoff relationship. Further work should explore alternative input vector structures that could help in improving the performance of the model.

References

Babuska R, Verbruggen HB (1997) Constructing fuzzy models by product space clustering. In: Fuzzy model identification, edited by H. Hellendoorn and D. Driankov, Springer-Verlag, Berlin, pp. 53–90

Bárdossy A, Duckstein L (1995) Fuzzy rule-based modelling with applications to geophysical, biological and engineering systems. CRC Press: Boca Raton, Florida

Bazartseren B, Hildebrandt G, Holz K-P (2003) Short-term water level prediction using neural networks and neuro-fuzzy approach. Neurocomputing 55: 439–450

Bezdek JC (1981) Pattern recognition with fuzzy objective function. Plenum Press: New York

Chang F-J, Hu H-F, Chen Y-C (2001) Counterpropagation fuzzy-neural network for streamflow reconstruction. Hydrological Processes 15: 219–232

Cordón O, Herrera G, Peregrín A (1997) Applicability of the fuzzy operators in the design of fuzzy controllers. Fuzzy Sets and Systems 86: 15–41

Cordón O, Herrera G, Peregrín A (2000) Searching for basic properties obtaining robust implication operators in fuzzy control. Fuzzy Sets and Systems 111: 237–251

Delgado M, Gómez-Skarmeta AF, Martín F (1998). A methodology to model fuzzy systems using fuzzy clustering in a rapid-prototyping approach. Fuzzy Sets and Systems 97: 287–301

Galambosi A., Duckstein L, Ozelkan EC, Bogardi I (1998). A fuzzy rule-based model to link circulation patterns, ENSO and extreme precipitation. In: Risk-based decision making in water resources, VIII, ASCE Press, New York, 83–103

Gautam DK, Holz KP (2001) Rainfall–runoff modelling using adaptive neuro-fuzzy systems. Journal of Hydroinformatics 3: 3–10

Gustafson DE, Kessel WC (1979) Fuzzy clustering with a fuzzy covariance matrix. In: Proceedings IEEE CDC, San Diego, CA, pp. 761–766, 10–12 January 1979

Hundecha Y, Bárdossy A, Theisen H-W (2001) Development of a fuzzy logic-based rainfall–runoff model. Hydrological Sciences-Journal-des Sciences Hydrologiques 46(3): 363–376

Kosko B (1997) Fuzzy Engineering. Prentice-Hall Inc.: Upper Saddle River, NY

Mamdani EH (1974) Application of fuzzy algorithms for control of simple dynamic plant. Proceedings of the IEEE 121(12): 1585–1588

Nash JE (1957) The form of the instantaneous unit hydrograph. IASH Publ. 45(3): 114–118

Nash JE, Foley JJ (1982) Linear models of rainfall–runoff systems. In: Rainfall–Runoff Relationships, edited by V. P. Singh, Proceedings of the International Symposium on Rainfall–Runoff Modelling, May 1981, Mississippi State University, USA, Water Resources Publications, pp. 51–66

Nash JE, Sutcliffe JV (1970) River flow forecasting through conceptual models, Part I – A discussion of principles. Journal of Hydrology 10: 282–290

Nayak PC, Sudheer KP, Rangan DM, Ramasastri KS (2004). A neuro-fuzzy computing technique for modelling hydrological time series. Journal of Hydrology 291: 52–66

Piegat A (2001) Fuzzy modelling and control. Physica-Verlag Heidelberg: New York

Pongracz R, Bartholy J, Bogardi I (2001) Fuzzy rule-based prediction of monthly precipitation. Physics and Chemistry of the Earth Part B -Hydrology Oceans and Atmosphere 26(9): 663–667

Shamseldin A (1997) Application of a neural network technique to rainfall–runoff modelling. Journal of Hydrology 199: 272–294

Shamseldin A, O'Connor KM (1996) A nearest neighbour linear perturbation model for river flow forecasting. Journal of Hydrology 179: 353–375

See L, Openshaw S (2000) A hybrid multi-model approach to river level forecasting. Hydrological Sciences Journal 45(4): 523–536

Vernieuwe H, Georgieva O, De Baets B, Pauwels VRN, Verhoest NEC, De Troch FP (2004) Comparison of data-driven Takagi-Sugeno models of rainfall-discharge dynamics. Journal of Hydrology 302: 173–186

Zadeh LA (1965) Fuzzy sets. Information and Control 8(3): 338–353

Zadeh LA (1975) The concept of a linguistic variable and its applications to approximate reasoning I, II and III. Information Sciences 8: 199–249, 301–357 and 9:43–80

Chapter 15
Using an Adaptive Neuro-fuzzy Inference System in the Development of a Real-Time Expert System for Flood Forecasting

I.D. Cluckie, A. Moghaddamnia and D. Han

Abstract This chapter describes the development of a prototype flood forecasting system provided in a real-time expert system shell called COGSYS KBS. Current efforts on the development of flood forecasting approaches have highlighted the need for fuzzy-based learning strategies to be used in extracting rules that are then encapsulated in an expert system. These strategies aim to identify heuristic relationships that exist between forecast points along the river. Each upstream forecast point automatically produces extra knowledge for target downstream forecast points. Importantly, these strategies are based on the adaptive network-based fuzzy inference system (ANFIS) technique, which is used to extract and incorporate the knowledge of each forecast point and generate a set of fuzzy "if–then" rules to be exploited in building a knowledge base. In this study, different strategies based on ANFIS were utilised. The ANFIS structure was used to analyse relationships between past and present knowledge of the upstream forecast points and the downstream forecast points, which were the target forecast points at which to forecast 6-hour-ahead water levels. During the latter stages of development of the prototype expert system, the extracted rules were encapsulated in COGSYS KBS. COGSYS KBS is a real-time expert system with facilities designed for real-time reasoning in an industrial context and also deals with uncertainty. The expert system development process showed promising results even though updating the knowledge base with reliable new knowledge is required to improve the expert system performance in real time.

Keywords Expert system · adaptive neuro-fuzzy inference system (ANFIS) · COGSYS KBS · flood forecasting

I.D. Cluckie
Water and Environmental Management Research Centre (WEMRC), Department of Civil Engineering, University of Bristol, Bristol, BS8 1UP, UK, e-mail: I.D.Cluckie@bristol.ac.uk

A. Moghaddamnia
Formerly of Water and Environmental Management Research Centre (WEMRC), Department of Civil Engineering, University of Bristol, Bristol, BS8 1UP, UK (currently at University of Zabol, Iran)

D. Han
Water and Environmental Management Research Centre (WEMRC), Department of Civil Engineering, University of Bristol, Bristol, BS8 1UP, UK

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

15.1 Introduction

In recent years, the utilisation of expert systems in real-time flood forecasting has been an issue of considerable importance. The real-time flood forecasting systems development process is essentially an experience-based process. An expert system is defined as a system that uses human knowledge captured in a computer to solve problems that ordinarily require human expertise (Turban and Aronson, 2001). Expert systems can represent that expertise or knowledge as a set of "if–then" rules. Historically, despite numerous successes in the use of expert systems in different fields, their widespread use in water resources has been limited. Expert systems are principally logic processors rather than efficient numerical processors. For problems that require a large number of input parameters, repetitive calculations or a confined solution space, procedural programming approaches have significant advantages over the rule-based programming approaches of most expert systems (Brachman et al., 1983). Expert systems are not well understood by most professionals in the field. For many areas of study in water resources, traditional programming approaches have served the profession admirably. Expert systems have been employed where traditional approaches have failed, such as HYDRO, an expert system used for determining parameters for use in watershed modelling (Reboh et al., 1982), and "Aviso Watch" (WatchDog), an expert system developed and deployed to be integrated with site-specific forecasting models to permit anticipation of future threats. In general, expert systems are an attractive analysis tool (Palmer and Holmes, 1988) that can encapsulate available knowledge for the benefit of future users.

The knowledge that can be encapsulated in hydroinformatic systems can only be knowledge which represents rational reasoning skills (Amdisen, 1994). Heuristic knowledge of incoming floods plays a key role in reducing flood damage, as does better planning of non-structural measures such as forecasting system development which helps in minimising the losses due to floods. Expert system applications that perform reasoning in dealing with uncertainty are enhanced by the use of techniques such as fuzzy logic, Bayesian logic, multi-valued logic and certainty factors (Openshaw and Openshaw, 1997). One of the key issues that hydrologists often face is the problem of scarce historical flood records imposing large uncertainty in forecasting flood probabilities. The adaptive neuro-fuzzy inference system (ANFIS) technique when used in the context of expert system development can improve the decision-making process in a complex and dynamic environment such as floods.

15.2 Real-Time Flood Forecasting Techniques

Real-time data availability, progress in software and hardware technologies, development of hydrological and hydraulic models, radar and remote sensing techniques, state-of-the-art information and communication technologies, advanced technical knowledge in different areas of hydrology and water resources and advances in artificial intelligence techniques bring many challenges for developing state-of-the-art

real-time flood forecasting systems. The inevitable need to couple numerical modelling activity from the mesoscale atmospheric to decision support system ends of the "model train" brings about an increasing additional interest in the exploitation of e-science technologies.

Many types of real-time flood forecasting techniques and models have been developed, which reflect the inherently stochastic nature of such environmentally driven processes and this has led to the recent interest in artificial neural network (ANN) and fuzzy logic (FL) techniques. ANN and FL techniques that consider the non-linearity in the rainfall–runoff process and the utilisation of soft computing techniques such as support vector machines (SVM), expert systems and genetic algorithms (GA) can be grouped together under the general term of "artificial intelligence".

So far the expert systems approach in real-time flood forecasting has not been seriously applied. In the literature, most applications for prediction are based on Takagi–Sugeno models and neuro-fuzzy techniques. An expert system such as "Aviso Watch" (WatchDog), which works much like the antivirus software on a computer, is one of the more successful cases of an expert system application to flood forecasting. By using rules that are specified by a rule editor and are fired by an inference engine, "Aviso Watch" identifies flood threats and notifies a dispatcher as well as disseminating information to other interested parties.

This chapter focuses on the process of forecasting water level and its time sequence at selected forecast points along a river during floods and is considered as a "real-time flood forecasting" process. Since forecasting discharge in real time is difficult and water level is the main concern in flood events, water level was chosen as the forecast variable. ANFIS is a good example of computing applied to real-time flood forecasting. It recognises patterns and adapts them to cope with the real-time environment thanks to its neural learning ability. It then incorporates human knowledge and expertise in the real-time decision-making process. ANFIS, as one of the available fuzzy techniques, provides expert systems with a powerful approximate reasoning capability. It provides an adaptive behaviour with a strong knowledge representation characteristic inherent in fuzzy inference systems. The main idea behind using the learning capabilities of a neural network in the ANFIS technique is to generalise a Takagi–Sugeno–Kang model in order to tune the membership function parameters on the premise of the fuzzy rules as well as the estimation of the coefficients of each local model on the consequent part of each rule.

Before assembling the ANFIS structure, data pre-processing was carried out to refine data collected during the case study period from January 1997 to May 1999. Apart from using standard interpolation methods for the reconstruction of missing values, the main purpose of this step was to reduce the data size, filter the noise and remove incorrect data. In addition, in order to split the data into three data sets for training, testing and validating, suitable and non-repeated trends and patterns were identified. In order to simplify the ANFIS model, principal components analysis (PCA) was utilised to select suitable variables, reduce dimensionality and reproduce the ANFIS model variability. Rainfall and discharge require more pre-processing before they can be passed through the ANFIS model. Low-pass Butterworth digital

filters with different orders and cut-off frequencies were also used to overcome the computational problems caused by noise-corrupted data.

15.3 The ANFIS Model

A number of neuro-fuzzy systems have been developed but the most well known is ANFIS (Jang, 1993); other approaches have been suggested by Nomura et al. (1992), Halgamuge and Glesner (1994), Wang and Mendel (1992), Shi et al. (1996) and Shi and Mizumoto (2000). ANFIS is one of the best trade-offs between neural and fuzzy systems, providing smoothness and adaptability. It is a powerful tool to extract rules from data as well as extraction of the impact of each input from this model. ANFIS is based on a fusion of ideas from fuzzy systems and neural networks. It possesses the advantages of both neural networks (e.g. learning abilities, optimisation abilities and connectionist structures) and fuzzy systems (e.g. human "if–then" rule thinking and ease of incorporation of expert knowledge). The self-learning ability of the ANFIS eliminates the use of complex mathematical models for flood forecasting despite the long time required to train the ANFIS model. ANFIS can be used to approximate and prototype a dynamic flood forecasting model with multiple inputs to provide the knowledge base for a real-time flood forecasting expert system. Other authors have utilised the ANFIS technique in different applications.

ANFIS provides a hybrid combination of neural and fuzzy systems. First-order Sugeno models containing 4, 27 and 32 rules were considered for the prototypes developed in this case study. Trapezoidal, Gaussian and generalised Bell membership functions with 4, 2 and 3 parameters, respectively, and the product inference rule were used at the fuzzification level. The firing strength was estimated as the fuzzifier output for each rule. The firing strength of each of the rules was normalised. Using the algebraic product aggregation method, defuzzification was carried out by the product of the firing strengths and the corresponding linear functions. The parameters defining the ANFIS membership functions, as well as the first-order polynomial coefficients, were obtained through a supervised training process using input/output data and the Matlab program. A description of the ANFIS model given the above can be summarised by the following equation:

$$F = \frac{\sum_{j}^{R} \left(w_{j0} + w_{j1}u_1 + \cdots + w_{jn}u_n\right) \prod_{j}^{R} \mu_{A_j}(u_1, u_2, \ldots, u_n)}{\sum_{j}^{R} \prod_{j}^{R} \mu_{A_j}(u_1, u_2, \ldots, u_n)} \tag{15.1}$$

where F is the output, $u_1, u_2, \ldots, u_n$ are the inputs, $\mu_{A_j}(u_1, u_2, \ldots, u_n)$ are the membership functions and $w_{j0}, w_{j1}, \ldots, w_{jn}$ are the adjustable coefficients. The gradient-descent algorithm was used to determine the parameters in the conditions and the least mean square algorithm was used to determine those in the consequent part. Equation (15.1) is used to describe the combination of numerous simple local

models and seeks to globally represent a system that may be highly non-linear. The system is described by membership functions that define the nature of the contribution of each local model to the global model and then decompose the input space into subspaces and consequently approximate the complex system in each subspace by a simple linear local model.

In order to partition the input space, the grid partitioning method was used and in some cases it could be taken as an initial state of partition for some adaptive partitioning methods. The ANFIS technique provided a more transparent representation of the non-linear systems when there was an ill-defined input/output system or the lack of a well-defined mathematical model. Owing to the dynamic nature of hydrological processes such as floods, fuzzy rules and membership functions must be adaptive to the changing environment and hydrological conditions of the catchment of interest in order for the forecast flood to be better represented. The tuned types and numbers of membership functions and fuzzy rules utilised could significantly improve the performance of the ANFIS model. A main advantage of the ANFIS model was that it could be made more applicable by employing well-adjusted membership functions and coefficients from the training process for the forecasting application in real time as it can be treated as a set of piecewise linear regression models. Figures 15.1 and 15.2 provide a schematic of the ANFIS network and the learning algorithm.

Figure 15.2 illustrates the use of two sets of ANFIS parameters for the premise (i.e. the non-linear parameters) and uses linear parameters for the consequent. The former represents the fuzzy partitions used in the rules and the latter represents the coefficients of the linear functions used in the rules. ANFIS uses a two-pass learning cycle including forward and backward passes. In the forward pass, parameters of the premise are fixed and parameters of the consequent are computed using a least squared error algorithm. In the backward pass, consequent parameters are fixed and parameters of the premise are computed using a gradient-descent algorithm.

In this study, the Matlab v.7 software package was used, which offers an adaptive neuro-fuzzy inference system (ANFIS) function by using a hybrid algorithm.

The Welland and Glen catchment was used for the case study. It is located in eastern England (1150 km^2), drained by the Rivers Welland and Glen. The geology and landscape of the area varies considerably from west to east. The flow

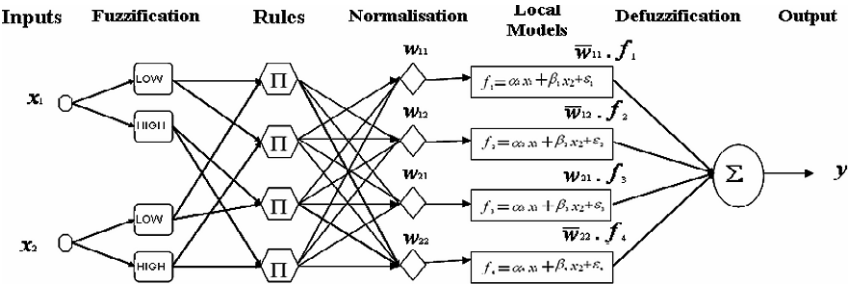


Fig. 15.1 An ANFIS network

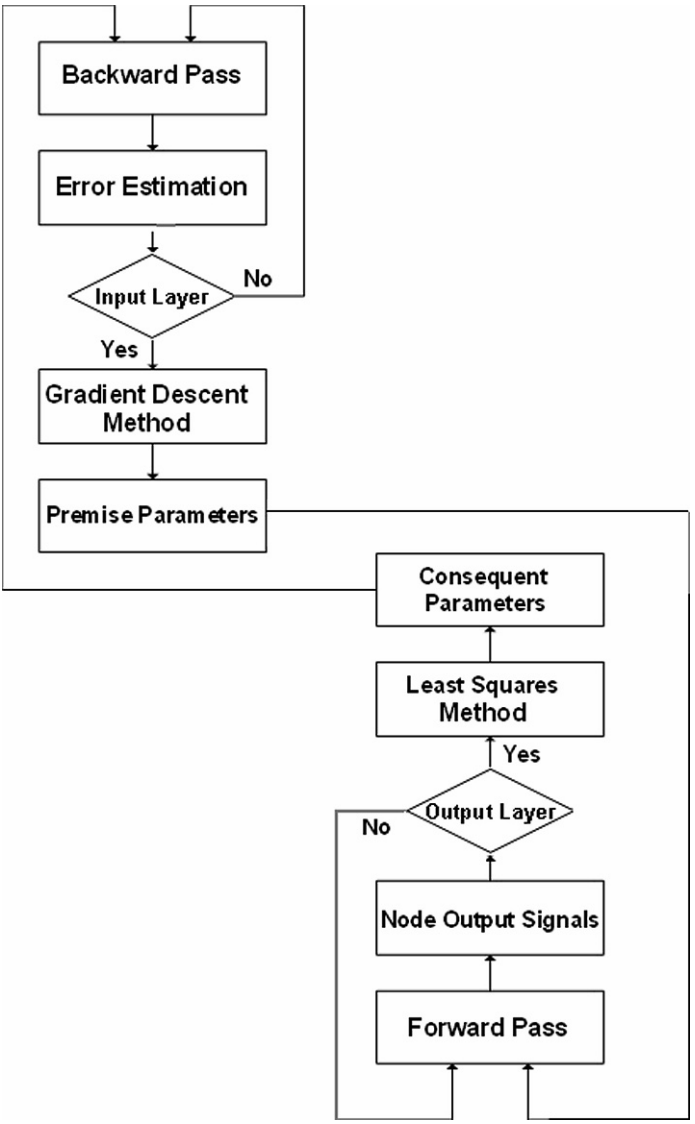


Fig. 15.2 The ANFIS learning algorithm

characteristics of the wetland area – known as the Fens – are complicated by a developed drainage system to protect it from flooding and to facilitate its drainage. Two major flood relief channels are used to route flood water away from population centres and to reduce the high-flow impact, impacts of pumping (irrigation) and seepage on river flow in the lower reaches. Flood control gates are also constructed on the Welland and Glen catchment indicating the complexity of the river and catchment system. The Welland and Glen catchment suffers from minor flooding in most years,

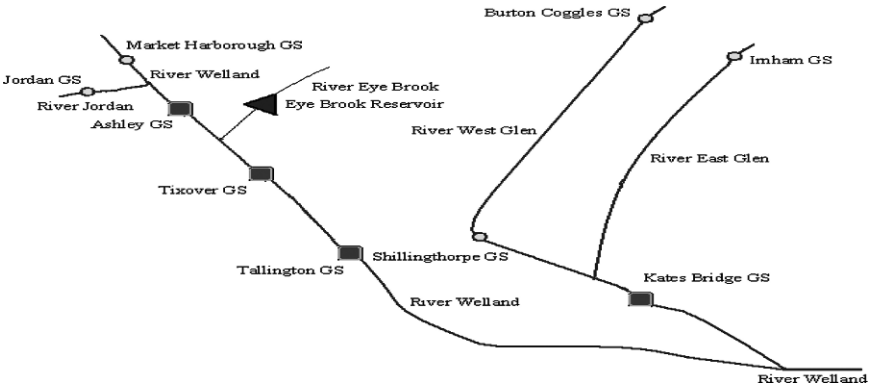


Fig. 15.3 Schematic diagram of the River Welland and the River Glen

particularly in the upper reaches. The catchment experienced significant flooding during March 1947 and Easter 1998 with villages and communication links affected in the valleys of the West and East Glens with further flooding in the town of Stamford and at other locations along the River Welland. A schematic of the river network of the Welland and Glen catchment and the forecast points along the main river and its tributaries is shown in Fig. 15.3.

15.4 Development Environment

The COGSYS KBS expert system shell was used in this study for reasons of performance, flexibility and industrial availability. COGSYS KBS has been developed in the face of heavy competition from Gensim's G2. COGSYS employs objects such as "super objects" (which create the knowledge engineer's own knowledge base objects), elements and enclosures. Popular inference mechanisms for AI applications compete with functional approaches and, more interestingly, with object-oriented programming.

In this study, four prototype forecasting locations were developed based on a selection of points taken from the existing forecast points at which the data were of sufficient quality for the ANFIS models to be trained and tested. The first prototype was based on past and present knowledge of two upstream forecast points and a downstream forecast point to forecast water level (5 hours ahead) at the downstream site. In this prototype, suitable explanatory variables were selected on the basis of principal components analysis (PCA) and cross-correlation analysis. These variables included decaying basin wetness and water level with different delay times. A second prototype was employed to use past and present knowledge of three upstream forecast points and the downstream forecast point in order to forecast 6-hour-ahead water levels at this point. The third prototype employed was based on the fuzzification of water level thresholds for a single forecast point upstream of Stamford town

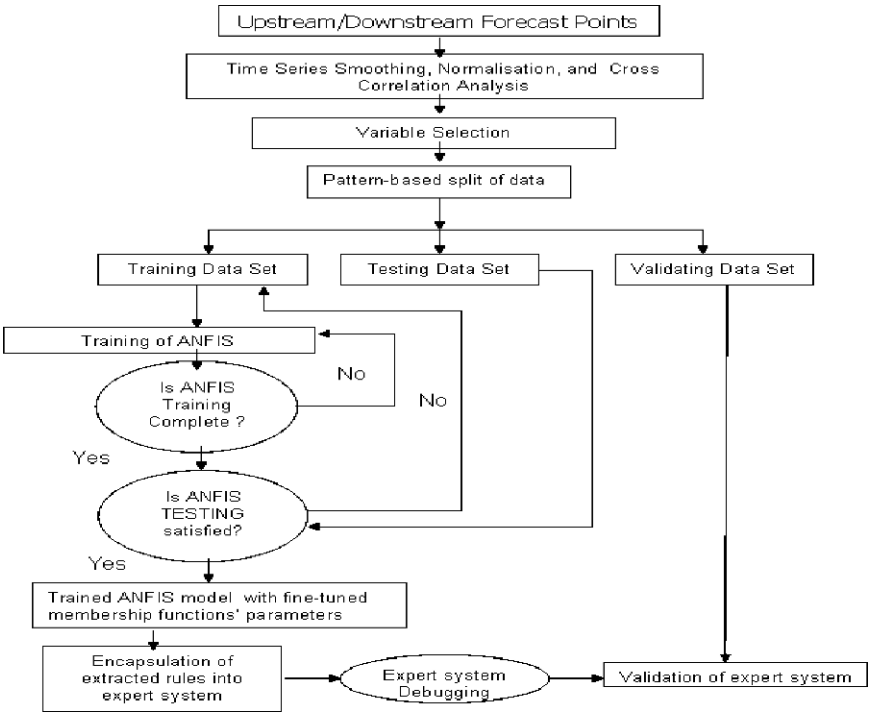


Fig. 15.4 Expert system development process and schematic of data flow diagram

in an attempt to reduce false alarms. The fourth and final prototype was developed to use past and present knowledge of a single forecast point based on three inputs of flow and water level with different delay times and a single output of 6-hour-ahead forecast water level at that point. A brief flow chart of the expert system development process and the data flow for this study is given in Fig. 15.4. The figure gives a step-by-step process of how the expert system was developed. It also describes the flow of data from the data pre-processing phase to the set-up phase for prototyping in the COGSYS KBS expert system shell.

Table 15.1 summarises the input variables employed in the ANFIS learning process and consequent prototyping in the COGSYS KBS expert system shell. The numbers in front of each variable describe the variable selection and the pre-processing steps on the input variables of the ANFIS models. The following is a brief description of the procedure applied to pre-process the time series of rainfall, discharge and water level:

a. A first-order low-pass Butterworth infinite impulse response (IIR) filter with cut-off frequency at 0.25 rad/sec was used to remove noise from the water level time series at Market Harborough and Jordan forecast points.
b. To calculate the decaying basin wetness, an *r*-value equal to 0.4 was determined by a trial-and-error method. Then a forward-moving average decaying basin

Table 15.1 ANFIS inputs and parameters used in the prototyping process

Prototype no.	Variable	Time	Forecast point	Location/ channel	Change (m)	Membership type	Membership type number	Number of rules
1	Water level (1)	$t-13$	Market Harborough GS	Welland	5960	Bell shape	2	32
	Water level (1)	$t-16$	Jordan GS	Jordan	5299			
	Decaying basin wetness (2)	$t-16$	Hallaton (3)	Sub-catchment no. 2	–			
	Water level (4)	$t-5$	Ashley GS	Middle Welland	7300			
		t						
2	Discharge (5)	t	Tixover GS	Welland T-S	65	Gaussian	3	27
	Water level (6)	$t-6$	Tixover GS	Welland T-S	0			
		t						
3	Water level	t	Tallington GS	Maxey Cut	40	Trapezoidal	4	4
4	Discharge (5)	$t-6$	Kates Bridge GS	Glen u/s SS	4480	Bell shape	2	64
	Water level (7)	$t-11$	Burton Coggles GS	Upper West Glen	100			
		$t-10$	Shillingthorpe GS	West Glen Shillingthorpe	−130			
		$t-6$	Irnham GS	Upper East Glen	0			
		$t-6$	Kates Bridge GS	Glen u/s SS	4360			
		t						

wetness with a time span equal to 168 hours was estimated by the smoothing functions in the HEC-DSS software.

c. A first-order low-pass Butterworth infinite impulse response (IIR) filter with cut-off frequency at 0.09 rad/sec was used to remove noise from the water level time series at Ashley forecast point.
d. A central-moving average discharge with a time span equal to 59 time steps for Tixover GS forecast point and a forward-moving average with a time span equal to 20 time steps for Kates Bridge GS forecast point were estimated by the smoothing functions in the HEC-DSS software.
e. A second-order low-pass Butterworth infinite impulse response (IIR) filter with cut-off frequency at 0.1 rad/sec was used to remove noise from the water level time series of Tixover forecast point.
f. A second-order low-pass Butterworth infinite impulse response (IIR) filter with cut-off frequency at 0.08 rad/sec was used to remove noise from the water level time series at Kates Bridge forecast point.

Table 15.1 shows the strong influence of water level as an input variable in each model. Observed data were split into three data sets for training, testing and validation. Training data have been used for learning purposes and testing data have been used for validating the ANFIS model. To let the model learn the behaviour of data during low flow and high flow (floods), the training data set was chosen so that it contained the flood event that occurred in April 1998 in the Welland and Glen catchment.

15.5 The Graphical User Interface

A graphical user interface (GUI) is a computer program that employs icons and graphics to display and manipulate data. In this study, the developed expert system was supported by a graphical user interface in order to present information in an

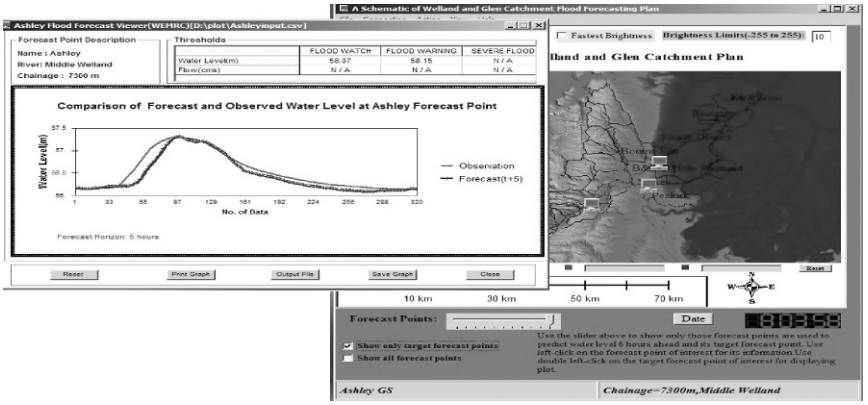


Fig. 15.5 Graphical user interface for the validation of prototype no. 1

easy to use format. The GUI was developed using Visual Basic. There are services such as “display picture” and “display plot”, with CSV format for the inputs and outputs. What appears on the screen is a main window indicating forecast points under consideration and is composed of menus, buttons and entry fields. Figure 15.5 represents the designed graphical user interface for one of the prototypes.

15.6 Results and Conclusions

This chapter has presented the objectives behind this study, which would be crowned by the development of a neuro-fuzzy system for each of the prototypes under consideration. This study highlighted the novel development of an expert system to real-time flood forecasting.

Table 15.2 shows the result of using ANFIS to forecast water level for the four prototypes. The interpretation of the results indicates that the prototype expert systems are capable of offering acceptable performance. For the analysis and comparison of the results, two performance criteria were used in this study: the normalised RMSE (root mean squared error) and CoE (Nash–Sutcliffe efficiency coefficient).

The ANFIS system converged rapidly to a solution with 4 epochs and an RMSE of 0.020 for prototype no. 2. Figure 15.6a and b shows observed and 6-hour forecast

Table 15.2 Comparison of observed and forecast water level for the four prototypes

Prototype no.	Epochs	Training		Testing		Validation	
		RMSE	CoE	RMSE	CoE	RMSE	CoE
1	92	0.010	0.996	0.087	0.964	0.051	0.978
2	4	0.005	0.9995	0.006	0.9995	0.020	0.9990
3	28	0.084	0.991	0.107	0.978	0.061	0.997
4	90	0.006	0.999	0.018	0.989	0.022	0.991

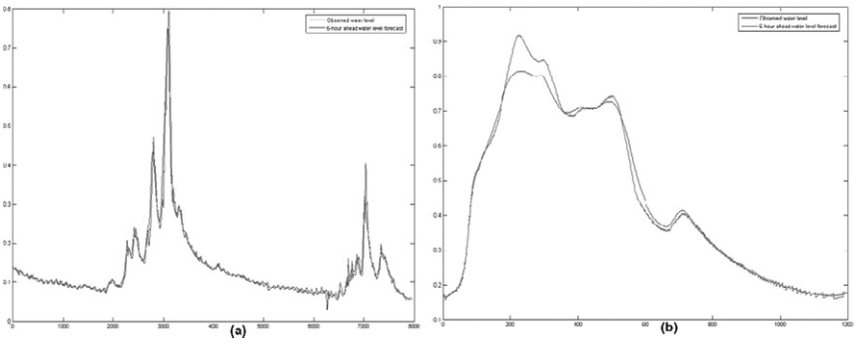


Fig. 15.6 The forecast performance of the testing data set for the prototype no. 4 (**a**) and the validation data set for the prototype no. 2 (**b**)

Table 15.3 Contingency table for evaluation of the accuracy of prototype no. 3

	Alarm type	Alarm status	Training		Testing		Validation	
			Yes	No	Yes	No	Yes	No
Observation	Flood watch	Yes	50	0	107	0	3	1
		No	2	255	3	93	0	95
	Flood warning	Yes	25	6	29	4	0	0
		No	0	255	0	93	0	95
	Severe flood	Yes	71	0	33	0	1	0
		No	4	255	1	93	0	95

water levels at Kates Bridge GS and Tixover GS forecast points. In prototype no. 3, as can be seen in Table 15.3, an ANFIS model with four rules has reasonably good accuracy. The adaptive learning behaviour of ANFIS indicated that this was a promising approach to knowledge acquisition in development of expert systems in real-time complex environments. A fusion system, which was based on the four ANFIS models developed in this study, can be utilised for real-time flood forecasting. The proposed methodology was able to learn and also provide transparency through its rules.

Acknowledgements The authors would like to thank the Iranian Ministry of Science, Research and Technology (IMSRT) for scholarship support for Mr. Alireza Moghaddamnia and TCCL Ltd for access to COGSYS KBS. Thanks are also due to the EU FP5 FLOODRELIEF Project (EVKI-CT-2002-00117) and the EPSRC Flood Risk Management Research Consortium (FRMRC – GR/S76304/01) for financial support.

References

Amdisen LK (1994) An architecture for hydroinformatic systems based on rational reasoning. Journal of Hydraulic Research 32: 183–194

Brachman RJ, Ameral S, Engleman C, Englemore RS, Feigenbaum EA, Wilkens DE (1983) What are expert systems. In: Building Expert Systems, F. Hayes-Roth, W.A. Waterman, and D.B. Lenat, eds., Addison-Wesley: Reading, MA, pp. 31–57

Halgamuge S, Glesner M (1994) Neural networks in designing fuzzy systems for real world applications. International Journal of Fuzzy Sets and Systems 65: 1–12

Jang JSR (1993) ANFIS: adaptive-network-based fuzzy inference system. IEEE Transactions on Systems, Man, and Cybernetics 23(3): 665–685

Nomura H, Hayashi I, Wakami N (1992) A self-tuning method of fuzzy control by descent method. In: Proc. IEEE Int. Conf. on Fuzzy Systems, San Diego, pp. 203–210

Openshaw S, Openshaw C (1997) Artificial Intelligence in Geography. Wiley: New York

Palmer RN, Holmes KJ (1988) Operational guidance during droughts: An expert system approach. Journal of Water Resources Planning and Management Division ASCE 114(6): 647–666

Reboh R, Reiter J, Gaschnig J (1982) Development of Knowledge-based interface to A Hydrological Simulation Program. SRI Int.: Menlo Park, California

Shi Y, Mizumoto M, Yubazaki N, Otani M (1996) A method of fuzzy rules generation based on neuro-fuzzy learning algorithm. Journal for Japan Society of Fuzzy Theory and Systems 8(4): 695–705

Shi Y, Mizumoto M (2000) Some considerations on conventional neuro-fuzzy learning algorithms by gradient descent method. Fuzzy Sets and Systems 112: 51–63

Turban E, Aronson JE (2001) Decision Support Systems and Intelligent Systems. Prentice Hall: Upper Saddle River, NJ

Wang, LX, Mendel JM (1992) Back-propagation fuzzy system as nonlinear dynamic system identifiers. In: Proc. IEEE Int. Conf. on Fuzzy Systems, San Diego, pp. 1409–1416

Chapter 16
Building Decision Support Systems based on Fuzzy Inference

C.K. Makropoulos, D. Butler and C. Maksimovic

Abstract Uncertainty and ambiguity are inherent properties of our understanding of the world and our ability to communicate this understanding to others, in both a quantitative and qualitative way. This fact makes uncertainty and ambiguity inherent in decision making processes and thus decision support tools need to provide capabilities for their effective handling. This paper presents an overview of a number of decision support systems, integrating quantitative and qualitative criteria, primarily by means of fuzzy inference as a tool for handling linguistic ambiguity and uncertainty. The decision support systems discussed cover a wide range of spatial scales, from local to regional, a number of different contexts, from urban to rural and address a variety of objectives, from urban sustainability to regional environmental protection. They have all been developed in the Urban Water Research Group of the Civil and Environmental Engineering Department at Imperial College London, over a period of 10 years. Despite their differences, the models discussed possess common underlying methodological concepts and have been developed to some extent with similar "building blocks". Issues of complementarities and added value which result from both the conceptual and methodological approaches adopted are explored and an indication of possible future directions is presented. It is concluded that a flexible, component-based, rapid-prototyping method for developing decision support systems capable of explicit handling of ambiguity and uncertainty through fuzzy inference are fundamental to the development of tools, which can be adopted in practice and can truly support inclusive decision making.

Keywords Ambiguity · decision support · fuzzy inference

C.K. Makropoulos
Department of Water Resources & Environmental Engineering, School of Civil Engineering, National Technical University of Athens, Greece, e-mail: cmakro@mail.ntua.gr

D. Butler
Centre for Water Systems, School of Engineering, University of Exeter, UK

C. Maksimovic
Dept. of Civil and Environmental Engineering, Imperial College London, UK

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

16.1 Introduction

The natural complexity, ever-present in real life, is transformed into uncertainty when building an abstraction of the real world, in other words, a model (Makropoulos and Butler, 2004a). For simple systems, where cause-effect relationships are well understood and are easily quantifiable, closed form mathematical expressions can be used to provide insight in the way the system functions. For complex systems, we can make a distinction between two cases: (a) the case where significant data are available (e.g. through well regulated experiments), in which case, model-free, or universal approximator methods (e.g. artificial neural networks) provide a methodology for dealing with uncertainty through learning, based on knowledge encapsulated in the data (Ross, 1995) and (b) the case where only scarce data exist and where only ambiguous or imprecise information is available. The ability of data to capture the knowledge required to model a system would tend to decrease with increasing system complexity. In the latter case, fuzzy logic provides a framework for formally handling this uncertainty, which includes the notions of ambiguity and imprecision rather than the more commonly addressed notion of randomness (Makropoulos and Butler, 2004a). The imprecision therefore in fuzzy systems is high particularly if there is a need for quantification of qualitative criteria and synthesis of non-homogeneous information (technical, economical, social etc), which are of major importance to the socio-technical contexts in which hydroinformatics is often applied (Abbott, 2000). In these cases, lack of knowledge per se, is coupled with scarce available data and ambiguous cause-effect relationships. In the discussion that follows, five different decision support systems are presented, all of which utilise fuzzy logic to handle, to some extent, imprecision and ambiguity and make use of non-homogenous information, including preference and perceptions of risk.

16.2 Decision Support Systems (DSS) in Context

Over the past 10 years, the need to integrate knowledge encapsulated in models, databases and expert opinions into the decision making process and more importantly communicate this knowledge to relevant stakeholders has been repeatedly stated within the hydroinformatics community (Price, 2000). Tejada-Guilbert and Maksimovic (2001) emphasise the seven Is approach (first outlined by Butler and Maksimovic (1999)) to environmental modelling: Integration (the development of coordinated and integrated models); Interaction (or the search for the additive, cumulative synergistic effects of complex systems); Interfacing (especially with the public and the environment); Instrumentation (in terms of real time control, sensors and non-invasive techniques); Intelligence (and the expansion of data, information, and knowledge through Geographic Information Systems (GIS) and hydroinformatics); Interpretation (the complementarity of data and judgement, and a wise combination of structured reasoning and disciplined imagination); and, finally Implementation (the capacity for true and concerted action and the transformation of

policies into reasonable practice that combines corrective and proactive strategies and tactics). The increasing user friendliness of GIS and the advent of general purpose, rapid prototyping programming languages such as MATLAB recently provided a computational and presentational framework facilitating the implementation of, at least some of, these principles.

We will briefly present below five tools, developed with these ideas in mind:

- The Urban Water Demand Management DSS
- The Catchment Pollution Assessment and Intervention Prioritisation DSS
- The New Urban Developments Screening Tool
- The New Urban Developments Optioneering Tool
- The New Urban Developments Site Suitability Assessment Tool

Although these tools cover a wide range of different domains and application scales (see Fig. 16.1) there are important common elements in their conceptual and developmental underpinnings:

- They are GIS-based or at least make significant use of spatial information in terms of analysis and presentation and are thus spatially-relevant and customisable to site specific characteristics and constraints.
- They handle linguistic variables (Zadeh, 1975) by means of fuzzy inference systems (FIS) and can thus make use of qualitative and quantitative characteristics and constraints.
- They are developed in a modular way, loosely coupling GIS, analytical engines, spreadsheets and databases. This allows elements of the system to be significantly updated without a need to re-code the entire system.

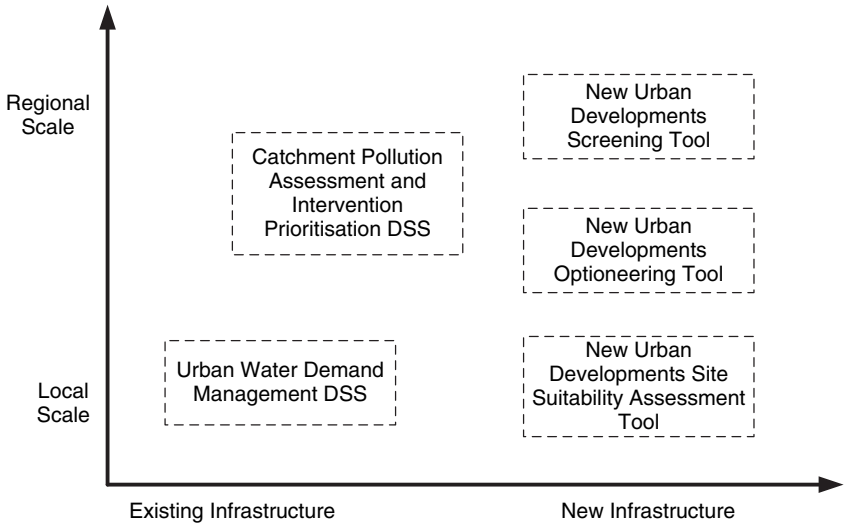


Fig. 16.1 Application domains of the Decision Support Systems presented

- They use MATLAB as the mathematical environment which allows for rapid prototyping and development of "exploratory" tools and components, thus allowing the testing of new ideas without the significant overheads in development time required for coding a wide variety of knowledge-based, fuzzy inference, control and optimisation algorithms in conventional computational languages, such as FORTRAN.
- As a consequence of the above, the tools developed are experimental prototypes, testing new ideas, mostly complementary to, rather in substitution of, commercial, off-the-shelf software, a fact that is compatible, in our view, with the role of academic research in hydroinformatics.

16.3 Presentation of the Tools

16.3.1 The Urban Water Demand Management DSS

This tool is a decision support system (Fig. 16.2) applied in the field of Water Demand Management (WDM) (Makropoulos and Butler, 2004b) to:

- Compare different water demand reducing scenarios (on the basis of water savings for a fixed investment or on a cost basis for a target water saving)
- Plan the site-specific implementation of these scenarios (identify where to apply each technique to get maximum results on a house-by-house basis)
- Assess the investment needed to achieve a desired reduction or
- Identify the best composite scenario dependent on a desired investment scheme.

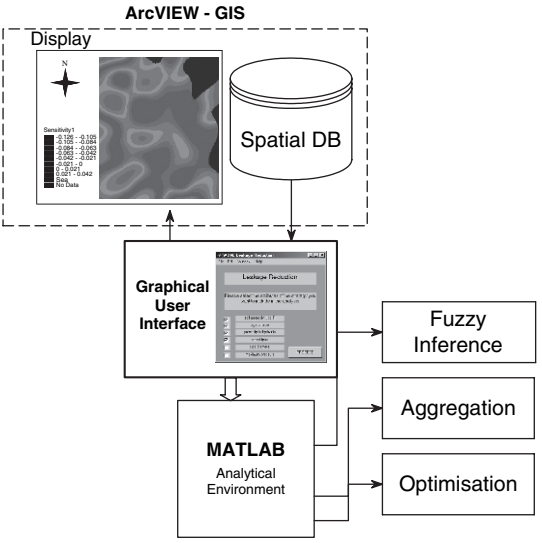


Fig. 16.2 Schematic of the Urban Water Demand Management DSS

The decision process supported by the tool can be described as follows: The user chooses which of available WDM strategies he/she wishes to take into account. Each strategy is broken down into a number of attributes, which directly influence its applicability in a given location. This decomposition to less complex elementary

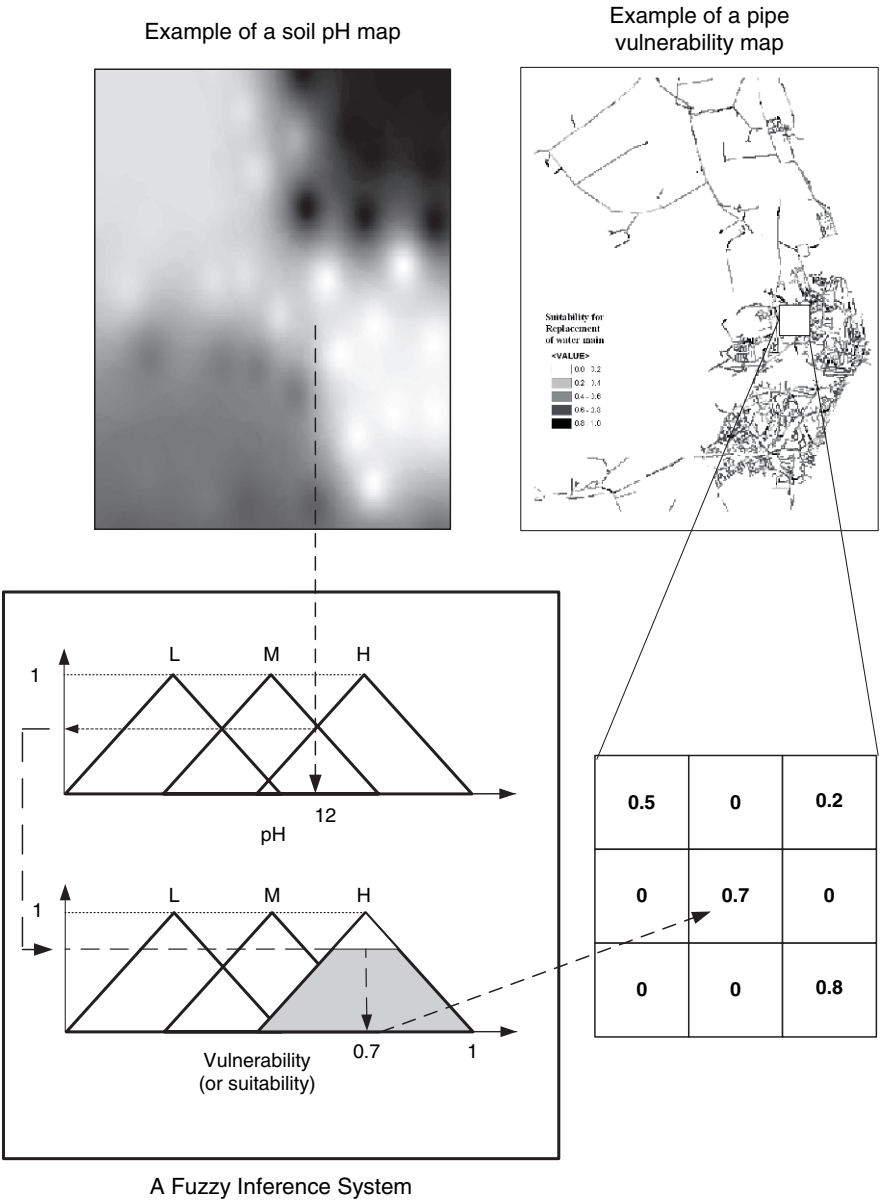


Fig. 16.3 Fuzzy inference for vulnerability map creation

components whose effect on the suitability of the strategy's application is easier and safer to define can be thought of as a good example of defining fuzzy membership functions for complex ill defined sets (Despic and Simonovic, 1997). The user chooses the attributes he/she wishes to take into account. This scalar approach tackles to a certain degree the problem of having to make decisions under various levels of data availability. Each selected attribute is imported into the system as a raster GIS layer and is then transformed in a suitability map for the relevant strategy to the extent that the attribute influences the strategy's applicability (Fig. 16.3). As the output is now homogenous (suitability maps in a [0,1] scale), the results of heterogeneous input are now comparable. The process is performed through fuzzy inference systems (both type-1 and type-2), which can analyse both singleton and non-singleton input (Mendel, 2001; Makropoulos and Butler, 2004a).

The standardised suitability maps for each attribute of a given strategy are then aggregated to provide one all-inclusive suitability map for the strategy's application. The user is presented with different options of aggregation including simple additive weighting (with direct weight assignment) and ordered weighted averaging (Yager, 1988). The later enables the incorporation of the decision maker's attitude towards risk in the outcome in the aggregation process. The optimism or pessimism of the decision maker is provided to the system as a user input and appropriate weights are calculated to convey this level of risk aversion into the aggregation outcome. The output strategy suitability maps for all strategies are then imported into the optimisation module. An optimisation algorithm is then employed to select a composite strategy. Optimisation includes a number of user input parameters (e.g. overall desired investment cost, per capita water consumption, cost for network rehabilitation per meter of network, cost for grey water recycling scheme introduction per household and cost of introducing metering per household). The multi-objective, evolutionary optimisation algorithm developed and used (Makropoulos and Butler, 2005) maximises water savings per day using overall investment cost as a constraint. The result is a combined strategy in the form of a detailed proposed WDM application map (master-plan) on a house-by-house level indicating the optimal location for installing (a) grey water recycling devices, (b) metering or (c) identifying network areas prone to leakage for replacement prioritisation.

16.3.2 The Catchment Pollution Assessment and Intervention Prioritisation DSS

This Decision Support Tool is a multicriteria, data driven tool with an Excel User Interface, combining MIKE Basin/GIS, Matlab and Excel/Visual Basic (see Fig. 16.4). The tool uses Multi-criteria analysis (MCA) to evaluate decision scenarios based on both environmental and economic criteria. It advises on recommended environmental actions to meet desired scenarios at a catchment scale. The tool accepts as inputs: data on pollution sources (point or diffused, including agricultural

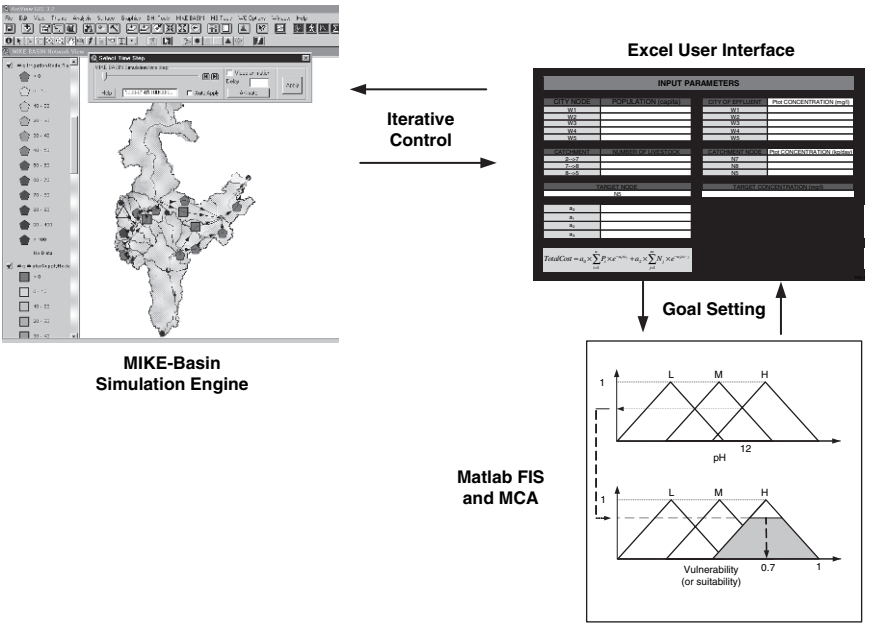


Fig. 16.4 Schematic of the Catchment Pollution Assessment and Intervention Prioritisation DSS

loads and livestock numbers), water abstraction and consumption and desired target effluent concentrations at given nodes. The user specifies the cost functions for a number of intervention options, the relative importance of a number of criteria, standardised through a fuzzy inference process, and an optimisation algorithm interactively runs the MIKE-Basin engine to achieve the required target concentration while minimising the stated objectives (Matsouris et al., 2005; Legesi et al., 2006).

The results from MIKE-Basin are combined with results from an economic model using a MCA approach based on fuzzy inference developed in MATLAB. Finally, a scenario analysis is used to explore both prescriptive and predictive scenarios for decision making at a catchment level.

16.3.3 The New Urban Developments Screening Tool

This decision support tool assists the selection of appropriate areas for new urban developments on the basis of sustainability criteria. The framework links diverse elements of the decision process, including articulated objectives, criteria and indicators as well as decision makers' preferences. Data are handled in the form of GIS raster files, structured in a series of rows and columns, forming cells. Each cell has its own properties, linked to various physical and geographical attributes and can be considered an "alternative" in the context of site suitability assessment, in that it potentially represents a solution to the siting problem (Malczewski, 1999).

Multi-criteria analysis is used to achieve a selection of the best alternatives based on consistent application of expert judgment and user preferences. A set of user-defined criteria are used to (a) narrow down the number of alternatives through a screening process, and (b) evaluate the remaining locations in terms of their suitability/sustainability. Criteria are divided into "crisp" and "fuzzy". Crisp criteria, also referred to as hard constraints, differentiate areas where development is feasible from those where it is not. The rest of the criteria are treated as fuzzy. In practical terms, each fuzzy criterion is linked to a relevant indicator (spatial attribute) associated with a fuzzy inference system (FIS) using "IF…THEN" rules (see for example, Makropoulos et al. (2003)). The input data are then analysed to provide suitability outputs for each location. This process is repeated for every criterion resulting in a set of different suitability values for any given location. A weight is then attached to each criterion depending on its (user specified) importance. Aggregation of the criteria yields a value for each location, representing its overall suitability. The users can create and apply their own scenarios. Scenario variability can be expressed in terms of selected criteria and weight assignment. Sensitivity analysis can thus be performed to quantify the impact of different choices to the final outcome. By ignoring certain criteria and placing more importance on others, users can focus on different aspects of sustainability and assess their impact on the siting of alternative urban development schemes.

The tool is comprised of ArcView GIS, an analysis module developed in MATLAB and a user interface created as an Excel spreadsheet. The user can specify the required area for the new site, on the basis of which the tool identifies which location within the entire study area is the most favourable and provides a breakdown of the scores of the selected site with respect to the criteria (Fig. 16.5). The user can therefore see the strong and weak points of the proposed site and modify his/her weightings if needed.

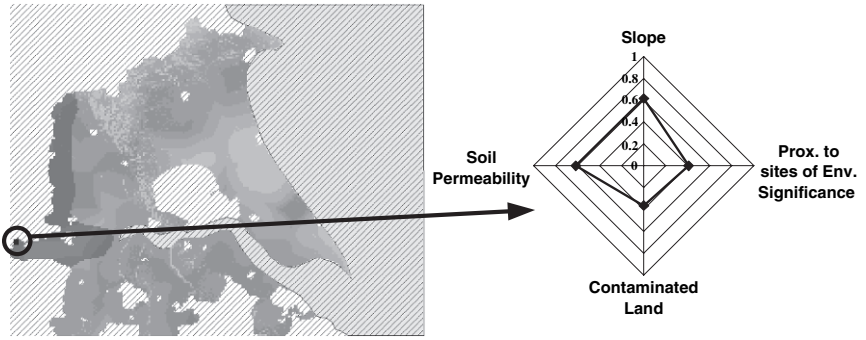


Fig. 16.5 Suitability Evaluation for Urban Development based on sustainability criteria

16.3.4 New Urban Developments Optioneering Tool

The New Urban Developments Optioneering Tool facilitates the selection of combinations of water managing strategies and technologies to support the delivery of integrated, sustainable water management for new developments. The tool is based on a water balance model which investigates the interactions between the major urban water cycle streams. A knowledge base library including sustainable urban water management options and technologies is included, containing data and information on the options' characteristics and performance. The methodology used for the comparison between various composite water management strategies and the assessment of the overall sustainability of the solutions provided is based on a sustainability criteria framework (Ashley et al., 2004).

The optioneering tool is then able to compare the performance of the various water technologies that are stored in the Technology Library, applied at all possible scales – from a local Sustainable Drainage System (SUDS) solution at a single household level, to a greywater recycling scheme for the entire development. A flow chart representing the optioneering procedure can be seen in Fig. 16.6. The process is driven by an evolutionary optimisation algorithm, which in turn evaluates the sustainability criteria against a "benchmark system" performance while treating them as fuzzy criteria (in the form of linguistic variables). This allows the presentation of the composite solution's sustainability evaluation as a spider diagram (Fig. 16.6), facilitating negotiations between the decision makers and allows for the exploration of what-if scenarios.

16.3.5 New Urban Developments Site Suitability Assessment Tool

The site suitability assessment tool deals with a specific type of urban water management problem: the object location problem which is defined by Makropoulos and Butler (2005) as "the determination of optimum locations for facilities in a given geographical area" with respect to technical, environmental social and economic objectives. This tool operates at a site, rather than a regional level (as compared to the screening tool). Initially, each facility, or technical option, is broken down into a (superset) of criteria, which directly influence its applicability at a given location. From this superset, the user chooses the criteria to be taken into account, subject to data availability. Criteria that are to be handled as constraints are also identified at this point and their cut-off points defined. Due to the heterogeneous scales with which different attributes are measured, each selected attribute is imported into the system as a map layer and is then standardized into a suitability map for the relevant strategy. This is performed through a fuzzy inference system (FIS). For example, the soil permeability criterion map of an infiltration option will be transformed into a suitability map with respect to soil permeability.

The standardized suitability maps for each criterion of a given water management option are aggregated to produce a composite suitability map for the option's

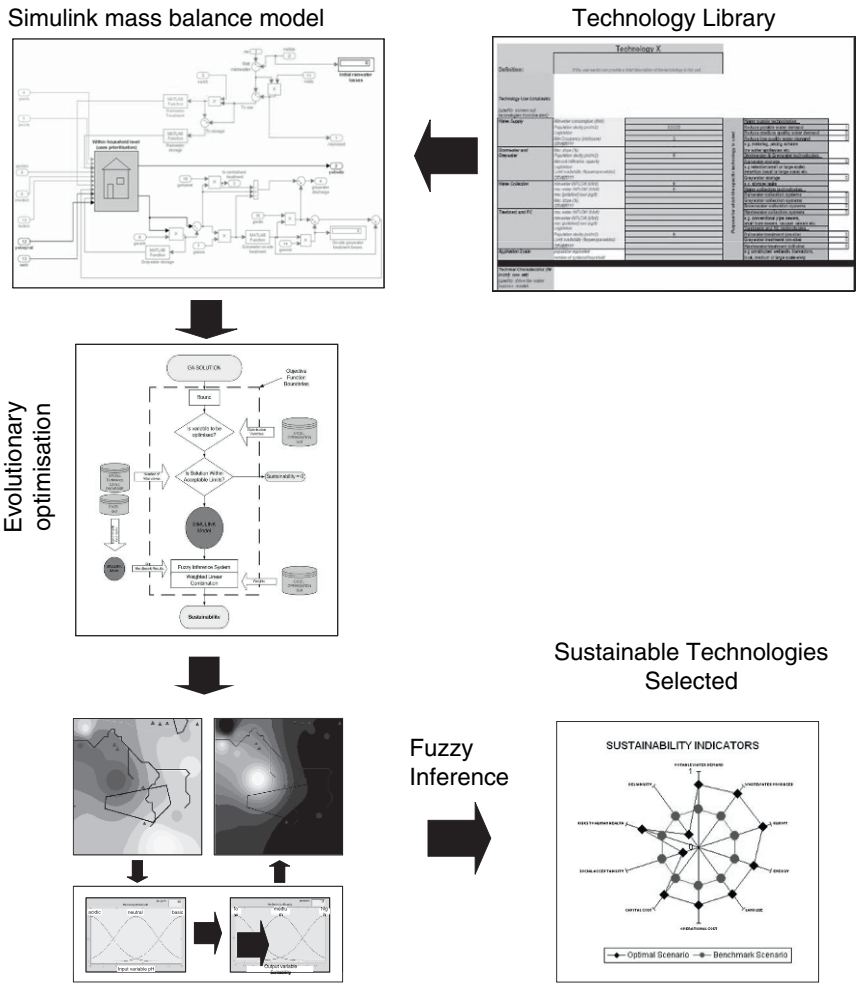


Fig. 16.6 Flow chart representing the rationale of the optioneering tool

application. The decision-maker is presented with different options of aggregation techniques including ordered weighted averaging (OWA) and spatial ordered weighted averaging (SOWA). An example of sensitivity of final results to decision-maker risk perceptions (ranging from optimistic to pessimistic) can be seen in Fig. 16.7. The SOWA technique (Makropoulos and Butler (2006)), has the major advantage of being able to include within the aggregation process a spatially variable risk index, which could be, for example, flood risk or health risk to sensitive population. The risk index can be used, through a set of linguistic FIS rules, to "bias" the applicability of specific options (e.g. "reduce" the applicability of constructed ponds for drainage purposes "near" schools). The composite suitability map for the option under evaluation then undergoes a "masking" process, imposing the selected

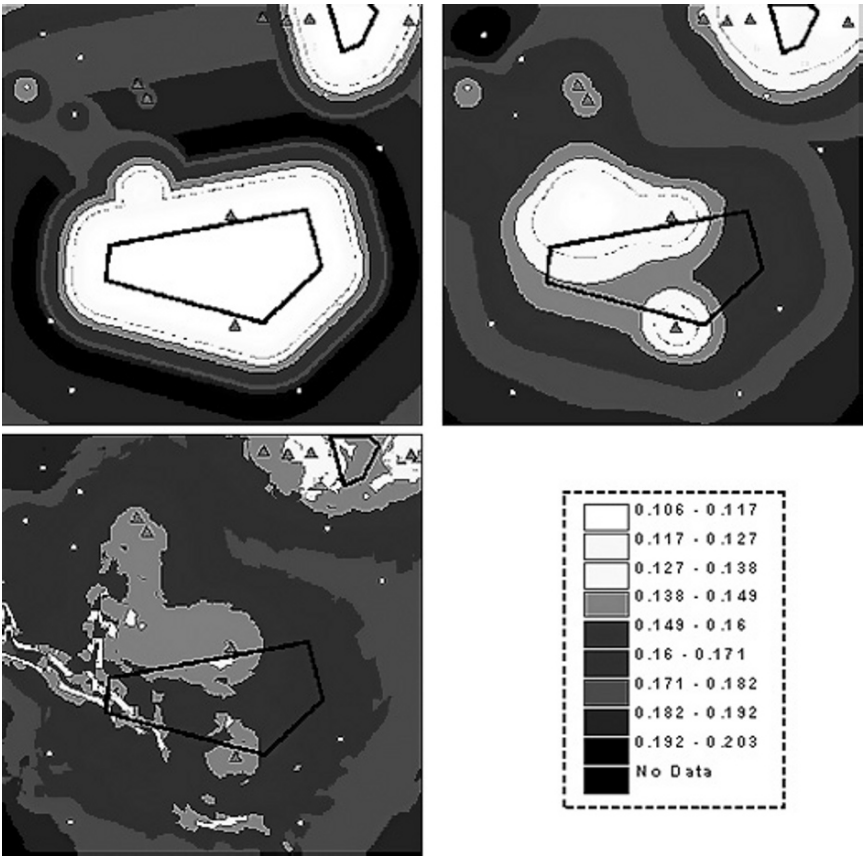


Fig. 16.7 Composite suitability maps representing sensitivity of the OWA method to different indicators of attitude towards risk

constraints. Based on the composite suitability maps, the tool provides the option of making a final recommendation as to the best possible (optimal) solution to the problem.

16.4 Discussion

The tools presented above utilise fuzzy inference to capture linguistic ambiguity and "operationalise" it within the context of environmental decision making. They all recognise the spatial character of (most of) the problems in this domain and have strong GIS components. They are all based on a loose coupling architecture resulting from file exchanges between separate, stand-alone, software, thus allowing for rapid development and testing of new ideas without a significant programming

overhead. Fuzzy inference engines, weighting and spatial analysis procedures have been developed in a modular way to maximise the possibilities for re-use when appropriate thus reducing the need to "re-invent" the wheel. The tools are strategic in nature, and can be used for thinking, negotiation and option screening at early stages of environmental planning. As such, these and other similar developments, inhabit the "great divide" (Turner, 1997) between social systems (and their research) and technical systems (and their modelling).

In our view, there is a growing need to overcome this divide in order to allow informatic tools (and hydroinformatics in particular) to move more towards their originally intended function of "augmenting the human intellect" (Engelbart, 1962), by acting as "thinking environments". Such thinking environments should be able to allow for the rapid integration of different models (ideally in a pick n' mix fashion) as well as the integration of qualitative and quantitative information and knowledge to provide support to a wide variety of problems, crossing traditional scientific domains, linking for example, social drivers to technical system developments and vice versa instead of the highly segmented approach still in use today. A key to the success of such environments would be their ability to allow for rapid development, to overcome the major problem of most DSS systems in use today, i.e. their cost-ineffectiveness vis-à-vis the considerable time investment they require for a usually case-specific end-product. This may require new collaboration and tool development platforms (similar to the discussion lead by Abbott, Harvey and colleagues in wiki.waterknowledge.org (for a more detailed discussion on this issue, see also Abbott's paper in this volume)). It may also require the development and integration within these tools of knowledge bases to facilitate the understanding of issues related to the crossing of traditional domains and scientific barriers, and would certainly suggest a need for stronger association of technological/natural science exploratory and simulation tools with social science exploration and simulation (including for example, agent-based modelling and dynamic systems modelling). Within this context, knowledge representation and inference (including its linguistic expressions and expert judgement encapsulation) become central to the success of such an ambitious project.

16.5 Conclusions

The tools described above, attempt to provide for some Integration (in the form of models, analytical engines, GIS and spreadsheets), explicitly address issues of Interaction (between for example environmental interventions and their impact as well as between interventions themselves), use GIS for Interfacing with the public and other stakeholders and for communicating results, include Intelligent components through primarily fuzzy inference, learning algorithms and evolutionary optimisation and support Interpretation through fuzzy inference assisted linguistic variables (Zadeh, 1975) and the incorporation of multicriteria techniques and ordered weighted averaging (Yager, 1988). It can only be hoped that these elements, together

with current and future developments in hydroinformatics, such as the ones discussed above, support a better Implementation process, by making the results more relevant to the socio-technical context in which they are required to function. The application of much of the work and the principles discussed here, both in research and practice, for example in Bosnia and Herzegovina (Pistrika et al., 2006), seem to support this claim and allow for an optimistic view of the future of knowledge-based environmental decision support.

References

Abbott MB (2000) The gender issue in hydroinformatics, or Orpheus in the Underworld. Journal of Hydroinformatics 2(2): 87–104

Ashley R, Blackwood D, Butler D, Jowitt P (2004) Sustainable Water Services: A procedural guide (SWARD), IWA Publishing

Butler D, Maksimovic C (1999) Urban water management – Challenges for the third millennium, Progress in Environmental Science, 1, 3, 213–235

Despic O, Simonovic S (1997) Methods in evaluating qualitative criteria in water resources multicriteria decision-making. Water Resources Research Rep. No. 37, Department of Civil and Geological Engineering, University of Manitoba, Manitoba, Canada

Engelbart DC (1962) Augmenting Human Intellect: A Conceptual Framework. Summary Report AFOSR-3223 under Contract AF 49(638)-1024, SRI Project 3578 for Air Force Office of Scientific Research, Stanford Research Institute, Menlo Park, Ca

Legesi K, Makropoulos C, Maksimovic C, Matsouris A (2006) A Multi-Criteria Decision Support Tool for Prioritising Environmental Actions in Data Scarce Areas. Proc. 7th Int. Conf. on Hydroinformatics

Makropoulos C, Butler D (2004a) Spatial decisions under uncertainty: fuzzy inference in urban water management. Journal of Hydroinformatics 6 (1): 3–18

Makropoulos C, Butler D (2004b) Planning site-specific Water Demand Management Strategies. Journal of the Chartered Institution of Water and Environmental Management (CIWEM), 18(1): 29–35

Makropoulos C, Butler D (2005) A multi-objective evolutionary programming approach to the "object location" spatial analysis and optimisation problem within the urban water management domain. Civil and Environmental Systems 22(2): 85–107

Makropoulos C, Butler D (2006). Spatial ordered weighted averaging: incorporating spatially variable attitude towards risk in spatial multicriteria decision-making. Environmental Modelling & Software 21(1): 69–84

Makropoulos C, Butler D, Maksimovic C (2003) Fuzzy Logic Spatial Decision Support System for Urban Water Management. J. of Wat. Res. Planning and Management 129(1): 69–78

Malczewski J (1999) GIS and Multicriteria Decision Analysis. John Wiley & Sons, Inc., NY

Matsouris A, Makropoulos C, Maksimovic C (2005) Spatial decision support for prioritising water resources management interventions at a catchment scale. Proc. 7th Hellenic Hydrogeology Conference, pp. 265–272, October 2005, Athens

Mendel J (2001) Uncertain, rule-based fuzzy logic systems. Prentice-Hall: Upper Saddle River NJ

Pistrika A, Makropoulos C, Maksimovic C (2006) A Decision Support Framework for Wetland Rehabilitation and Management – A Case Study of Bardača Wetland. Proc. XXIII Conference of Danubian Countries on the Hydrological Forecasting and Hydrological Bases of Water Management, Belgrade, no. 39 (on CDROM)

Price RK (2000) Hydroinformatics and urban drainage: an agenda for the beginning of the 21st century. Journal of Hydroinformatics 2(2): 133–147

Ross T (1995) Fuzzy logic with engineering applications. McGraw-Hill: NY

Tejada-Guilbert A, Maksimovic C (eds.) (2001) Outlook for the 21st century, In: Frontiers in Urban Waters: Deadlock or Hope, UNESCO, IWA (publ.) 410p. Cornwall, UK

Turner W (1997) Introduction: The sociotechnical system and computer supported cooperative work. In: Social Science, Technical Systems, and Cooperative Work, GC Bowker, S Leigh Star, W Turner and L Gasser (eds). Lawrence Erlbaum Associates: New Jersey

Yager R (1988) On ordered weighted averaging aggregation operators in multicriteria decision-making. IEEE Transactions on Systems, Man and Cybernetics 18(1): 183–190

Zadeh LA (1975) The concept of a linguistic variable and its applications to approximate reasoning. Information Science 8: 199–249

Part IV
Global and Evolutionary Optimization

Chapter 17
Global and Evolutionary Optimization for Water Management Problems

D. Savic

Abstract Optimization is a type of modelling which provides solutions to problems that concern themselves with the choice of a "best" configuration or a set of parameters that achieve some objective. The search for the "best" (maximum or minimum) is called global optimization where the goal is to find the best solution in nonlinear models that may have a multitude of local optima. Evolutionary optimization is a term denoting algorithms that mimic natural evolution and "evolve" better solutions through evolutionary mechanisms such as selection, reproduction, crossover and mutation. They are particularly suited for water system applications where the optimization problem often involves a mixture of discrete and continuous decision variables or potential solutions that must be evaluated with complex simulation models, whose linearization or derivative calculations would be difficult or impossible. Evolutionary optimization is a mature technology with a large number of applications in a multitude of areas, including water engineering and management. This chapter explains the basics of evolutionary optimization and illustrates its use in many areas of water system management.

Keywords Evolutionary optimization · water supply management · groundwater remediation · urban drainage system design · model calibration

17.1 Introduction

Providing effective decision support to improve planning, decision making and performance of water systems is of critical importance to water managers. There are many examples where lack of decision support for water-related problems has led to huge environmental, social and economic costs to societies. The latest information technology in the form of decision-support software can prove to be crucial for minimizing such adverse effects. Environmental decision making now relies upon

D. Savic
Centre for Water Systems, School of Engineering, Computer Science and Mathematics University of Exeter, Exeter, EX4 4QF, United Kingdom, e-mail: d.savic@exeter.ac.uk

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

a portfolio of techniques and models to provide information for the managers and politicians who will make decisions. With its focus on the application of information and communication technologies to water system problems, hydroinformatics provides an important pool of techniques and models used to aid decision making. The main reason to rely on any model in a decision-making process is to provide a quantitative assessment of the effects of decisions on the system being considered. A model also provides a fairly objective assessment as opposed to subjective opinions of system behaviour. Thus, models should be used in support of decision making. Optimization is just another type of modelling which provides solutions to problems that concern themselves with the choice of a "best" configuration or a set of parameters that achieve some objective. The search for the "best" (maximum or minimum) is also called *global optimization* where the goal is to find the best solution in nonlinear models that may have a multitude of local optima. This type of optimization is also called single-objective optimization because it finds a single best solution corresponding to the minimum or maximum value of a single-objective function that often lumps different objectives into one. Although this simple definition could lead to criticisms that emphasizing "find the best" leads users to oversimplify real-world problems, it should be stressed that optimization is viewed here as a tool for *supporting decisions* rather than for *making decisions*, i.e. should not substitute decision makers nor the decision-making process!

Many real-world decision-making problems need to achieve several objectives: minimize risks, maximize reliability, minimize deviations from desired levels, minimize cost, etc. Single-objective optimization is useful as a tool which should provide decision makers with insights into the nature of the problem, but usually cannot provide a set of alternative solutions that trade different objectives against each other. On the contrary, in a multi-objective optimization with conflicting objectives, there is no single optimal solution. The interaction among different objectives gives rise to a set of compromised solutions, largely known as the trade-off, nondominated, noninferior or Pareto-optimal solutions.

Over the course of the last two decades computer algorithms mimicking certain principles of nature have proved their usefulness in various domains of application. Researchers and practitioners have found especially worth copying those principles by which nature has discovered "stable plateaus" in a "rugged landscape" of solution possibilities. Such phenomena can be found in annealing processes, central nervous systems and biological evolution, which in turn have lead to the following optimization methods: simulated annealing, artificial neural networks and evolutionary optimization (EO). This article will focus on EO and its use in supporting decision makers faced with water management problems.

17.2 Evolutionary Optimization

Over many generations, natural populations evolve according to the principles first clearly stated by Charles Darwin (Darwin, 1859). The main principles are those of preferential survival and reproduction of the fittest members of the population. In

addition, the maintenance of a population with diverse members, the inheritance of genetic information from parents and the occasional mutation of genes characterize natural systems. Evolutionary optimization encompasses general stochastic artificial-evolution search methods based on natural selection and the aforementioned mechanisms of population genetics. This form of search evolves throughout generations, improving the features of potential solutions by means of biologically inspired operations (Michalewicz, 1999). Although it represents a crude simplification of natural evolution, EO provides efficient and extremely robust search strategies.

The advantage of using EO over traditional optimization methods, such as gradient descent methods or linear and nonlinear programming, is that they can solve a wide variety of problems without needing to linearize the problem or requiring derivative calculations of the objective function. If a simulation model capable of providing a quantitative assessment of the effects of decisions on the system is available, then an EO tool could be used to optimize the system being considered. This property is particularly important for water-related applications, where the optimization problem often involves a mixture of discrete and continuous decision variables or potential solutions that must be evaluated with complex simulation models, whose linearization or derivative calculations would be difficult or impossible. In addition, EO sampling is global, rather than local, thus reducing the tendency to become entrapped in local minima and avoiding dependency on a starting point. However, these stochastic methods cannot guarantee global optimality with certainty, although their robustness often makes them the best available method for global optimization problems. Therefore, EO is well suited for analytical decision support for complex water-related problems and thus can be easily incorporated into a decision-support framework. On the other hand, if, for a certain problem, there exists a mathematical programming methodology that can solve the problem satisfactorily, the use of EO techniques is not recommended due to their usually slow progress (as compared to classical mathematical programming methods). This chapter presents a simple, binary genetic algorithm (GA) as one of the many pathways EO researchers have followed in the last three decades. Other EO techniques are just briefly outlined. A review is then made of EO applied to various water-related decision-making problems.

17.3 Genetic Algorithms

Genetic algorithms are probably the best known type of EO methods. John Holland (1975) proposed the theory behind GAs in 1975, which was further developed by Goldberg (1989) and others in the 1980s. These algorithms rely on the collective learning process within a population of individuals, each of which represents a search point in the space of potential solutions. A variety of applications has been presented since the first works and GAs have clearly demonstrated their capability to yield good approximate solutions even in cases of complex multimodal, discontinuous, nondifferentiable models.

```
1)  BEGIN                                   /* Genetic Algorithm */

2)          t := 0;                         /* start with an initial time */

            Init_population P (t);          /* initialise a random popula-
                                               tion of initial individuals */
            evaluate P (t);                 /* evaluate fitness of all indi-
                                               viduals of initial population
                                               */

3)          WHILE NOT finished DO           /* test for termination criterion
                                               (time, fitness, etc.) */
                BEGIN
                    t := t + 1;             /* increase the time counter */
    (a)             P' := select_parents P (t);  /* select parents for offspring
                                                    production */
    (b)             recombine P' (t);       /* recombine "genes" of se-
                                               lected parents */
    (c)             mutate P' (t);          /* perturb the population sto-
                                               chastically */
    (d)             evaluate P' (t);        /* evaluate its new fitness */
    (e)             P := survive P,P' (t);  /* select the survivors by us-
                                               ing actual fitness */
                END
4)  END.
```

Fig. 17.1 Pseudo-code for a genetic algorithm

Genetic algorithms are used for a number of different application areas and the complexity of the algorithm depends on the problem in hand. However, for the purpose of illustration the following description of GAs is given under the assumption that it is used for function optimization. The standard GA steps in the form of a pseudo-code are shown in Fig. 17.1.

The analogy with nature is established by the creation within a computer of an initial population of individuals (step 2 in Fig. 17.1) represented by chromosomes, which is, in essence, a set of character strings that are analogous to the chromosomes found in DNA. If a simple function optimization example is used to illustrate the GA procedures, e.g. optimize $f(x)$, a chromosome may represent a set of parameter values x_i (being optimized) generated at random within pre-specified bounds.

17.3.1 Coding

Standard GAs use a binary alphabet (characters may be 0s or 1s) to form chromosomes. Parameters being optimized are coded using binary strings. Figure 17.2

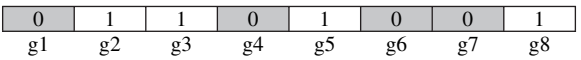


Fig. 17.2 A binary chromosome

shows a typical 8-character string, i.e. 8-bit chromosome (each bit is analogous to a gene).

If only one real-valued parameter is coded in Fig. 17.2, then a decoded parameter value may be calculated as

$$x = x_{\min} + \frac{x_{\max} - x_{\min}}{2^7 - 1} \left(\sum_{i=0}^{7} g_i \times 2^i \right)$$

where $x_{\min}$ and $x_{\max}$ are the lower and upper bounds on parameter x. If the parameters are discrete, then a total of 28 possible discrete values can be represented using the 8-bit string. At this point it should be noted that not all EO algorithms restrict representation to the binary alphabet which makes them more flexible and applicable to a variety of decision-making problems.

17.3.2 Fitness and Selection

Once the initial population has been generated the individuals in the population then go through a process of evolution. In nature, different individuals compete for resources (food, water and shelter) in the environment. Some are better than others. Those that are better are more likely to survive, attract mates, have and successfully rear offspring and thus propagate their genetic material. The measure of how good the individual is at competing in its environment is called the fitness of the individual. Consequently, the selection of who gets to mate is a function of the fitness of the individual. The selection operator has to be formulated to ensure that selection pressure is applied appropriately, i.e. that better individuals (with higher fitness) have a greater likelihood of being selected for mating, but also to ensure that worse individuals are not completely excluded (i.e. still have a small probability of being selected) since they may contain some important genetic material. Tournament selection is the most often used method of selecting individuals for mating in GAs. It runs a "tournament" among a few individuals chosen at random from the population and selects the winner (the one with the best fitness) for the reproduction phase. The probability of an individual winning the tournament is proportional to its fitness.

17.3.3 Reproduction

In nature, sexual reproduction allows the creation of genetically radically different offspring that still belong to the same species as their parents. A simplified look at

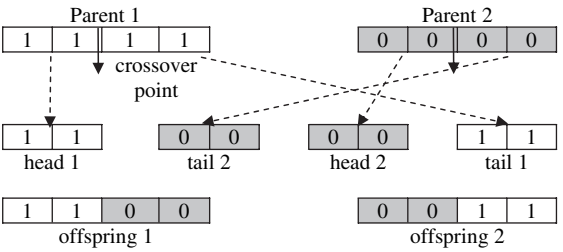


Fig. 17.3 Single-point crossover

what happens at the molecular level reveals that two chromosomes exchange pieces of genetic information. This is the recombination operation, which is generally referred to as *crossover* because of the way that genetic material crosses over from one chromosome to another (Fig. 17.3). During the reproductive phase of the GA (step 3 in Fig. 17.1), individuals are selected from the population and recombined, producing offspring which will comprise the next generation. A single-point crossover takes two individuals (parents in Fig. 17.3) and cuts their chromosome strings at some randomly chosen point if a pre-specified probability is achieved. The newly created head segments stay in their respective places while the tail segments are then crossed over to produce two new chromosomes. If the crossover probability is not achieved, the two parents are transferred to the mating pool unchanged.

17.3.4 Mutation

Mutation also plays a role in the reproduction phase, though it is not the dominant role, as is popularly believed, in the process of evolution (Fig. 17.4). In GAs mutation randomly alters each gene with a small probability, thus providing a small amount of random search. If the probability of mutation is set too high, the search degenerates into a random search. This should not be allowed and a properly tuned GA is not a random search for a solution to a problem. As a simulation of a genetic process a GA uses stochastic mechanisms, but the result is distinctly nonrandom (better than random).

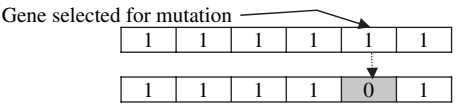


Fig. 17.4 A single mutation

17.3.5 Multi-objective Evolutionary Optimization

Many real-world optimization problems involve multiple objectives that need to be considered simultaneously. If these objectives are conflicting, as is usually the case,

a single solution optimal for all the objectives cannot be found. As a consequence, the assessment of the fitness of a set of solutions is not straightforward (compared to single-objective optimization) and calls for a method that provides a formal definition of the qualitative notion of compromise. The vast majority of multi-objective EO algorithms presented to date solve this predicament through the concept of Pareto dominance, which is embedded in their ranking procedures. Each solution of the Pareto-optimal set is not dominated by any other solution, i.e. in going from one solution to another, it is not possible to improve on one objective (e.g. reduce the risk) without making at least one of the other objectives worse (e.g. increase cost). Pareto dominance is then used to classify the solutions generated and to drive the search for a better set of nondominated solutions. The need to identify as many solutions as possible within the Pareto-optimal range often represents a problem for standard optimization techniques. By maintaining and continually improving a population of solutions, an EO algorithm can search for many nondominated solutions at the same time (in a single run), which makes it a very attractive tool for solving multi-objective optimization problems.

17.4 Other Evolutionary Optimization Algorithms

Genetic algorithms are not the only algorithms based on similarities with biological evolution. Evolutionary programming (Fogel and Atmar, 1992), evolution strategy (Schwefel, 1981), classifier system (Holland, 1975) and genetic programming (Koza, 1992) are all techniques which draw their power from principles adopted from natural processes.

Evolutionary programming and evolution strategy are stochastic optimization techniques similar to GAs, but which do not use a binary representation for parameters being optimized and crossover as a search strategy. These optimization techniques use a representation which follows from the problem, i.e. if parameters are real-valued numbers then they are represented by a floating-point string (chromosome). The mutation operation simply changes the solution parameters according to a statistical distribution which weights minor variations in offspring as highly probable and substantial variations as increasingly unlikely as the (global) optimum is approached. Like GAs, the evolutionary programming and evolution strategy techniques are useful for optimizing problem solutions when other techniques like gradient descent or direct, mathematical programming methods are not possible.

Classifier systems are cognitive systems which use adaptive techniques to evolve (learn) rules of the type “IF–THEN”. A single rule is termed as a “classifier” and a representation is chosen that makes it easy to manipulate rules by, for example, encoding them into binary strings. A set of classifiers constitutes a population and new rules evolve by employing a GA to generate a new set of rules from the current population.

Genetic programming is a technique for evolving programs to solve problems. Therefore, the individuals of the population are programs and by manipulating them in a GA-like manner new programs are generated. The process obviously needs new ways of representation and even new programming languages (Koza, 1992).

Evolutionary optimization algorithms are currently popular tools and are being applied to a wide range of water-related problems. Their appeal is their conceptual simplicity and an apparent applicability to problems where little knowledge is available. The following section presents various application areas to which these programs have been applied in the past.

17.5 Applications of Global and Evolutionary Optimization to Water-Related Problems

There are many examples of genetic algorithms and evolutionary optimization algorithms successfully applied to water-related optimization problems, with the number of real-world applications steadily growing. The following is a nonexhaustive review of applications, some of which have been used in practice while others remain as research topics.

17.5.1 Water Supply Management

In the area of water supply management, most applications of evolutionary optimization are for reservoir operations and control. Esat and Hall (1994) were the first to demonstrate advantages of using a GA over standard dynamic programming (DP) techniques in terms of computational requirements. Further studies by Oliveira and Loucks (1997), Wardlaw and Sharif (1999), Merabtene et al. (2002), Cui and Kuczera (2003) and Reis et al. (2005) followed, showing that the main advantages of the EO (and hybrid) approaches were that (i) discretization of state and decision spaces were not required nor were the initial trial state trajectories needed; (ii) complex reservoir systems need not be decomposed as in successive approximation-based approaches; (iii) EO approaches have the ability to work with highly complex, even discontinuous objective functions; and (iv) noninvertible systems can be easily handled without introducing dummy state variables as in DP-based approaches.

17.5.2 Groundwater Remediation

Optimal design of a groundwater pump-and-treat system is a difficult task, especially given the computationally intensive nature of field-scale remediation design. Genetic algorithms have been used extensively for remediation design because

of their flexibility and global search capabilities. McKinney and Lin (1994) and Ritzel et al. (1994) were the first to implement evolutionary optimization techniques on groundwater remediation designs. While McKinney and Lin used a single-objective GA to maximize the pumping rate from an aquifer, minimize the water supply development cost and minimize the total aquifer remediation cost, Ritzel et al. used a multi-objective GA to allow better consideration of different objectives. Subsequently, many other authors developed further the GA approaches to the pump-and-treat remediation method, including Aly and Peralta (1999), Yoon and Shoemaker (1999), Smalley et al. (2000), Erickson et al. (2002) and Espinoza et al. (2005).

17.5.3 Model Calibration

One of the earliest applications of EO model calibration problems is the work by Wang (1991) who calibrated a conceptual rainfall–runoff model using a simple binary GA. Following that application, a number of papers appeared in the literature applied to calibration of (i) rainfall–runoff models (Franchini, 1996; Ndiritu and Daniell, 2001), (ii) urban drainage models (Liong et al., 2001; di Pierro et al., 2005), (iii) water distribution models (De Schaetzen et al., 2000; Kapelan et al., 2003) and (iv) groundwater models (Lingireddy, 1998; Solomatine et al., 1999), etc.

17.5.4 Urban Drainage System Design

Although EO algorithms seem well suited to urban drainage system infrastructure design, there have not been many applications in this area. Rauch and Harremoes (1999) discussed the potential of GA applications in urban drainage modelling and highlighted model calibration and model predictive control as the main applications. Yeh and Labadie (1997) applied a multi-objective GA to generate non-dominated solutions for system cost and detention effects for a catchment in Taiwan while Diogo et al. (2000) applied a GA to optimize the design of a tri-dimensional urban drainage system. The application of EO to find cost-effective solutions to sewer flooding has been reported by Long and Hogg (2005).

17.5.5 Water Distribution System Design

Water distribution system design optimization is one of the most heavily researched areas in the hydraulics profession. Recently, EO algorithms have become the preferred water system design optimization technique for many researchers (Simpson et al., 1994; Dandy et al., 1996; Savic and Walters, 1997) because they demonstrate

good ability to deal with complex, nonlinear and discrete optimization problems. Most of these early works have used pipe diameters as the only decision variables, although storage tanks, pumps, etc. are also important decision variables. Furthermore, attaining acceptable system reliability has been a challenge in the optimal design of water distribution networks. Tolson et al. (2004) used the first-order reliability method coupled with a genetic algorithm to find reliable optimal design solutions to the water distribution system design problem. Kapelan et al. (2003) and Babayan et al. (2005) came up with two different methods of incorporating robustness/reliability into the design of the water distribution system based on efficient use of EO algorithms. Multi-objective EO algorithms based on structured messy GAs were first used by Halhal et al. (1997) to plan improvements of ageing water distribution systems, while Vamvakeridou-Lyroudia et al. (2005) have combined multi-objective EO algorithms with fuzzy reasoning for benefit/quality evaluation of both design and operation of a water distribution system including tanks and pumps as decision variables.

17.6 Conclusions

Typing in the words "genetic algorithms" into one of the popular Internet search engines results in over 12 million hits! This shows that if genetic algorithms and evolutionary optimization in general were a novelty only two decades ago, they are certainly a mature technology now with an astonishingly large number of applications in a multitude of areas. The usefulness of these algorithms in solving difficult, less structured, real-world problems has made them a favourite choice among the traditional methods, namely gradient search, random search and other methods. However, where specialized techniques exist for solving a particular problem, they are likely to outperform EO approaches in both speed and accuracy of the final result. Therefore, the appeal of EO techniques is in their conceptual simplicity, global search capabilities and an apparent applicability to problems where little knowledge is available (because they have the ability to explore and learn from the problem domain). Various areas of water management have already benefited from the use of EO as evidenced by the vast literature available in this domain.

There is, however, a need to emphasize that EO algorithms are "black box" stochastic iterated search methods and rely on repeated evaluation of a large number of candidate solutions to the problem in question, which means they typically need to incur the computational expense of thousands of simulations. This leads to several issues which should be addressed by researchers in the future. For example, hybridizing stochastic search with machine learning (Jourdan et al. 2005) has shown the great promise to reduce the number of computationally expensive fitness evaluations, with 40–50% time savings on large-scale problems coupled with reliably better solutions, despite relatively naïve configurations and algorithm design choices in this initial work. Several other promising methodologies like the use of *parallel* or *grid computing* or replacing computationally expensive fitness evaluators

with approximate ones – *surrogate modelling* (Bandler and Madsen, 2001) and *meta modelling* (Kleijnen and Sargent, 2000) – have also been suggested as a way forward. Another example of a further research topic is the so-called *hyperheuristics*, a technique for discovering algorithms which directly construct good solutions to a class of problems – so far investigated on bin-packing, timetabling and scheduling problems – in which an EO algorithm attempts to find a fast constructive algorithm for the problem class in question. Further ideas and challenges for EO could be found in Corne et al. (1999).

This chapter has explained the basics of EO and illustrated its use in many areas of water system management. It is hoped that by introducing the reader to these interesting examples one could grasp the idea of EO and their potential use with greater ease.

References

Aly AH, Peralta RC (1999) Comparison of a genetic algorithm and mathematical programming to the design of groundwater cleanup systems. Water Resour. Res. 35(8): 2415–2425.

Babayan AV, Kapelan Z, Savic DA, Walters GA (2005) Least Cost Design of Robust Water Distribution Networks Under Demand Uncertainty, J. Water Resour. Plng. Mgt. ASCE 131(5): 375–382.

Bandler X, Madsen K (2001) Editorial – surrogate modelling and space mapping for engineering optimisation. Opt. and Engrg. 2: 367–368.

Corne D, Dorigo M, Glover F (1999) New Ideas in Optimisation (Advanced Topics in Computer Science), McGraw-Hill Education.

Cui L-J, Kuczera G (2003) Optimizing urban water supply headworks using probabilistic search methods. J. Water Resour. Plng. Mgt., ASCE 129(5): 380–387.

Dandy GC, Simpson AR, Murphy LJ (1996) An improved genetic algorithm for pipe network optimization. Water Resour. Res. 32(2): 449–458.

Darwin C (1859) On the Origin of Species by the Means of Natural Selection, John Murray, London.

De Schaetzen WBF, Walters GA, Savic DA (2000) Optimal sampling design for model calibration using shortest path, genetic and entropy algorithms, Urban Water 2(2): 141–152.

Diogo AF, Walters GA, de Sousa ER, Graveto VM (2000) Three-dimensional optimization of urban drainage systems. Computer Aided Civil and Infrastructure Engrg. 15: 409–426.

di Pierro F, Djordjevic S, Kapelan Z, Khu S-T, Savic DA, Walters GA (2005) Automatic calibration of urban drainage model using a novel multi-objective genetic algorithm, Water Science and Technology, IWA 55(5): 43–52.

Erickson M, Mayer A, Horn J (2002) Multi-objective optimal design of groundwater remediation systems: application of the niched Pareto genetic algorithm (NPGA). Adv. Water Resour. 25(1): 51–65.

Esat V, Hall MJ (1994) Water resources system optimization using genetic algorithms. Proc. 1st Int. Conf. on Hydroinformatics, Balkema, Rotterdam, The Netherlands, pp. 225–231.

Espinoza FP, Minsker BS, Goldberg DE (2005) Adaptive hybrid genetic algorithm for groundwater remediation design, J. Water Resour. Plng. Mgt., ASCE, 131(1): 14–24.

Fogel DB, Atmar JW eds. (1992) Proceedings of the First Annual Conference on Evolutionary Programming, Evolutionary Programming Society, San Diego, USA.

Franchini M (1996) Use of a genetic algorithm combined with a local search method for the automatic calibration of conceptual rainfall-runoff models, Hydrological Sci. J. 41(1): 21–39.

Goldberg DE (1989) Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley.

Halhal D, Walters GA, Ouazar D, Savic DA (1997) Multi-objective improvement of water distribution systems using a structured messy genetic algorithm approach, J. Water Resour. Plng. Mgt., ASCE 123(3): 137–146.

Holland JH (1975) Adaptation in Natural and Artificial Systems, MIT Press.

Jourdan L, Corne D, Savic D, Walters G (2005) Preliminary Investigation of the 'Learnable Evolution Model' for Faster/Better Multiobjective Water Systems Design, In Carlos A. Coello Coello, Arturo Hernández Aguirre and Eckart Zitzler (eds.), Evolutionary Multi-Criterion Optimization. Third International Conference, EMO 2005, pp. 841–855, Springer. Lecture Notes in Computer Science Vol. 3410, Guanajuato, México, March.

Kapelan Z, Savic DA, Walters GA (2003) A hybrid inverse transient model for leakage detection and roughness calibration in pipe networks, J. Hydraul. Res., IAHR 41(5): 481–492.

Kleijnen JPC, Sargent RG (2000) A methodology for fitting and validating metamodels in simulation. European J. Op. Res. 120: 14–29.

Koza JR (1992) Genetic Programming: On the Programming of Computers by Means of Natural Selection, MIT Press.

Lingireddy S (1998) Aquifer parameter estimation using genetic algorithms and neural networks. Civil Engrg. Env. Syst. 15: 125–144.

Liong SY, Khu ST, Chan WT (2001) Derivation of Pareto front with genetic algorithm and neural network. J. Hydrol. Engrg., ASCE 6(1): 52–61.

Long R, Hogg S (2005) Fastnett – optimized flooding solutions, in Proc. of the Eight Int. Conf. on Computing and Control for the Water Industry, In Savic, D., G.A. Walters, R. King and S-T. Khu (eds.), University of Exeter, UK, Vol. 2, pp. 289–293.

McKinney DC, Lin MD (1994) Genetic algorithm solution of groundwater management models. Water Resour. Res. 30(6): 1897–1906.

Merabtene T, Kawamura A, Jinno K, Olsson J (2002) Risk assessment for optimal drought management of an integrated water resources system using a genetic algorithm. Hydrol. Process. 16: 2189–2208.

Michalewicz Z (1999) Genetic Algorithms + Data Structures = Evolutionary Programs, Springer-Verlag.

Ndiritu JG, Daniell TM (2001) An improved genetic algorithm for rainfall-runoff model calibration and function optimization. Math. Comput. Modell. 33: 695–705.

Oliveira R, Loucks DP (1997) Operating rules for multi-reservoir systems. Water Resour. Res. 33(4): 839–852.

Rauch W, Harremoes P (1999) On the potential of genetic algorithms in urban drainage modelling. Urban Water 1: 79–89.

Reis LFR, Walters GA, Savic DA, Chaudhry FH (2005) Multi-reservoir operation planning using Hybrid Genetic Algorithm and Linear Programming (GA-LP): An Alternative Stochastic Approach, Water Resour. Mgmt. 19 (6): 831–848.

Ritzel BJ, Eheart JW, Ranjithan S (1994) Using genetic algorithms to solve a multiple objective groundwater pollution containment problem, Water Resour. Res. 30(5): 1589–1603.

Savic DA, Walters GA (1997) Genetic algorithms for the least-cost design of water distribution networks. ASCE J. Water Resour. Plng. and Mgmt. 123(2), 67–77.

Schwefel H-P (1981) Numerical Optimization of Computer Models, John Wiley.

Simpson AR, Dandy GC, Murphy LJ (1994) Genetic algorithms compared to other techniques for pipe optimization. ASCE J. Water Resour. Plng. and Mgmt. 120(4), 423–443.

Smalley JB, Minsker BS, Goldberg DE (2000) Risk-based in-situ bioremediation design using a noisy genetic algorithm. Water Resour. Res. 36(10): 3043–3052.

Solomatine DP, Dibike YB, Kukuric N (1999) Automatic calibration of groundwater models using global optimization techniques. Hydrological Sci. J. 44(6): 879–894.

Tolson BA, Maier HR, Simpson AR, Lence BJ (2004) Genetic algorithms for reliability-based optimisation of water distribution systems. *J. Water Resour. Plng. Mgmt.* 130(1): 63–72.

Vamvakeridou-Lyroudia LS, Walters GA, Savic DA (2005), Fuzzy multiobjective optimization of water distribution networks, J. Water Resour. Plng. Mgt., ASCE 131(6): 467–476.

Wang QJ (1991) The genetic algorithm and its application to calibrating conceptual rainfall-runoff models. Water Resour. Res. 27(9): 2467–2471.

Wardlaw R, Sharif M (1999) Evaluation of genetic algorithms for optimal reservoir system operation. J. Water Resour. Plng. Mgt., ASCE 125(1): 25–33.

Yeh C-H, Labadie JW (1997) Multiobjective watershed-level planning of storm water detention systems. J. Water Resour. Plng. Mgt., ASCE 123(6): 336–343.

Yoon J-H, Shoemaker CA (1999) Comparison of optimization methods for groundwater bioremediation. J. Water Resour. Plng. Mgt., ASCE 125(1): 54–63.

Chapter 18
Conditional Estimation of Distributed Hydraulic Conductivity in Groundwater Inverse Modeling: Indicator-Generalized Parameterization and Natural Neighbors

F.T-C. Tsai and X. Li

Abstract This research develops an indicator-generalized parameterization (Indicator GP) method to improve the conditional estimation of spatially distributed hydraulic conductivity in groundwater modeling. Indicator GP, which honors distinct pointwise hydraulic conductivity measurements, integrates a natural neighbor (NN) interpolation method and Voronoi tessellation (VT) through a set of data indicators. The indicators use binary values to determine the selection of measured hydraulic conductivity data in the parameterization method. With Indicator GP, hydraulic conductivity is conditionally estimated between the Voronoi zones and the NN interpolated (smoothed) field. The conditional estimation is greatly improved beyond any specific parameterization method. Indicator GP is used to identify the distribution of hydraulic conductivity by finding the optimal binary values of data indicators such that the misfit between observed and calculated groundwater head observations is minimized. As a consequence, the identification problem is formulated as an integer nonlinear programming (INLP) problem, which involves groundwater modeling and leads to a complicated combinatorial optimization problem. We use a genetic algorithm (GA) to globally search for the optimal value of data indicators. A GA tremendously reduces the problem complexity and improves the inverse solution. We demonstrate the potential of using Indicator GP in groundwater inverse modeling via a numerical model where a synthetic hydraulic conductivity distribution is unknown and needs to be identified. The results show that using GA and Indicator GP outperforms both VT and NN methods, avoids the local optimum, captures the non-smoothness of heterogeneity, and gives a small misfit value with reasonable estimation error.

Keywords Inverse modeling · hydraulic conductivity · parameter estimation · parameterization · genetic algorithm · geostatistics

F.T-C. Tsai
Department of Civil and Environmental Engineering, Louisiana State University, 3418G Patrick F. Taylor Hall, Baton Rouge, LA 70803-6405, e-mail: ftsai@lsu.edu

X. Li
Department of Civil and Environmental Engineering, Louisiana State University, 3418G Patrick F. Taylor Hall, Baton Rouge, LA 70803-6405, e-mail: xli11@lsu.edu

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

18.1 Introduction

Hydraulic conductivity estimation in groundwater modeling is essential for the purposes of gaining a better understanding of aquifer characteristics and improving groundwater modeling reliability. It is generally understood that identification of a hydraulic conductivity value at each location is nearly impossible when only a limited amount of data (e.g., hydraulic conductivity measurements and groundwater head observations) are available. The approximation of the spatially distributed hydraulic conductivity is necessary and is usually conducted through parameterization methods (McLaughlin and Townley, 1996).

Current parameterization schemes are typically either a zonation method or an interpolation method, which result in either piecewise constant distribution (zones) or a smooth continuous distribution, respectively. However, geological formation processes do not constrain the hydraulic conductivity to be a smooth distribution or zonation structure. Presumption of both types of distributions is unrealistic and will mislead the consequence of estimation. Improving the existing parameterization technique is necessary in order to improve the hydraulic conductivity representation. Recently, a generalized parameterization (GP) method was developed to integrate a zonation structure and an interpolation method to improve the identification of hydraulic conductivity in a deterministic field (Tsai and Yeh, 2004).

This study focuses on the GP method and its theoretical development when a random field of hydraulic conductivity is considered. First, an indicator-generalized parameterization (Indicator GP) is introduced to conditionally estimate the hydraulic conductivity field. Indicator GP introduces the data indicators to all sampled hydraulic conductive measurements to determine the selection of a set of sample data for conditional estimation in the GP method. Second, the Voronoi tessellation (VT) and natural neighbor (NN) interpolation methods are adopted in the Indicator GP. The conditional variance of estimation error using Indicator GP is then derived. Third, a genetic algorithm (GA) is used to search for the optimal binary value of the data indicators by minimizing the misfit of groundwater head observations to obtain the best conditional estimation of hydraulic conductivity. Last, a numerical example demonstrates the advantage of using GA to find the global optimal solution and shows that Indicator GP is better than any single parameterization method.

18.2 Indicator-Generalized Parameterization (Indicator GP)

Consider m sample data of hydraulic conductivity (K) from an aquifer (Ω). The natural logarithmatic K value (i.e., $\pi = \ln(K)$) will be analyzed in this study under the assumption of the log-normal distribution of hydraulic conductivity. We first construct m distinct zones as shown in Fig. 18.1 by honoring the sample data such that each sample data represents a distinct zone, denoted as ($\Omega^{(i)}$), $i = 1, 2, \ldots, m$. Therefore, $\Omega = \Omega^{(1)} + \Omega^{(2)} + \cdots + \Omega^{(m)}$. For example, an estimate at a location $\mathbf{x}_0$ is determined by.

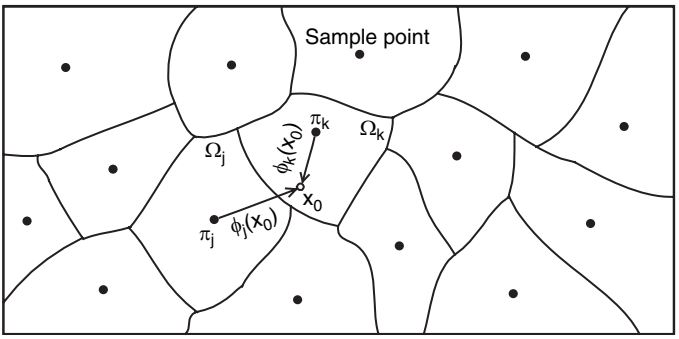


Fig. 18.1 Sample locations, a zone structure, and an interpolation method

$$\pi_{\text{Zonation}}(\mathbf{x}_0) = \left\{ \pi_k | \mathbf{x}_0 \in \Omega^{(k)} \right\} \tag{18.1}$$

The zonal values are assigned to be the corresponding sampled values. To avoid confusion, the zonation method in (18.1) should be distinguished from other zonation studies in groundwater inverse modeling (e.g., Carrera and Neuman, 1986) where the zonal values are identified through an inverse procedure. In (18.1), the number of zones is the same as the number of sample data of hydraulic conductivity. The zonal values are determined by the sampled values. In the case that two adjacent sample sites have the same sampled K values, two zones are still considered according to (18.1).

As shown in Fig. 18.1, an interpolation method can also be selected to interpolate the sampled values in space to obtain a smooth distribution of hydraulic conductivity as follows:

$$\pi_{\text{Interpolation}}(\mathbf{x}_0) = \sum_{j=1}^{m} \phi_j \pi_j \tag{18.2}$$

where ϕ_j are the basis functions according to the chosen interpolation method and $\sum_{j=1}^{m} \phi_j = 1$. Equation (18.2) can also represent a functional-based interpolation method, e.g., algebraic polynomials and trigonometric polynomials, which are popular in statistical learning theory (Cherkassky and Mulier, 1998). However, the functional-based interpolation methods do not honor the sampled values and are beyond the scope of the present work.

An indicator-generalized parameterization (Indicator GP) method is developed to integrate the designated zone structure and interpolation method through m data indicators as follows:

$$\pi_{\text{IGP}}(\mathbf{x}_0) = \sum_{\substack{j=1 \\ j \neq k(\mathbf{x}_0)}}^{m} (\pi_j - \pi_{k(\mathbf{x}_0)}) \phi_j I_j + \pi_{k(\mathbf{x}_0)} \tag{18.3}$$

where $\{I_1, I_2, \ldots, I_m\}$ have binary values $\{0, 1\}$, and $k(\mathbf{x}_0)$ represents the kth sample data index for the unsampled location $\mathbf{x}_0$ in the kth zone.

The data indicators determine the selection of a set of sample points for estimation. If an indicator value is one, the corresponding sampled site is considered in the estimation at $\mathbf{x}_0$. A sample data point is not used in the estimation if its indicator value is zero. An unsampled location $\mathbf{x}_0$ in the k th zone always has the sample data $\pi_{k(\mathbf{x}_0)}$ in the estimation. According to (18.3), the Indicator GP conditionally estimates a mixed distribution of hydraulic conductivity between the zone structure ($\{I_1, I_2, \ldots, I_m\} \in \{0\}$) and the smooth distribution ($\{I_1, I_2, \ldots, I_m\} \in \{1\}$).

In the past, the parameterization method concentrated on finding the best basis functions, e.g., the kriging method where the basis functions (kriging weights) are obtained by minimizing the conditional variance (Delhomme, 1979). However, the kriging method may overly smooth hydraulic conductivity heterogeneity. The kriged field may not result in good agreement with groundwater observations when groundwater hydrology is considered. Other than finding the best basis functions, this study focuses on the natural neighbor (NN) interpolation method and investigates its geostatistical property.

18.3 Geostatistics of Indicator GP with Natural Neighbors

The Indicator GP is applicable to many existing interpolation methods. Regardless of the zone shape, the Indicator GP is also applicable to many zone structures that honor the sample values and partition the hydraulic conductivity distribution into m zones according to the m sample data. This study employs an NN interpolation method and Voronoi tessellation (VT) in the Indicator GP.

18.3.1 Voronoi Tessellation

Voronoi tessellation is a zonation method that neutrally partitions a region into zones according to the nearest sample location. In a two-dimensional field as shown in Fig. 18.2(a), VT determines the zone Ω_j (first-order Voronoi cell) according to

$$\Omega_j = \Omega(\mathbf{x}_j) = \{\mathbf{x} | d(\mathbf{x}, \mathbf{x}_j) < d(\mathbf{x}, \mathbf{x}_l),\ \forall l \neq j,\ l = 1, \ldots, m\} \tag{18.4}$$

where $d(\mathbf{x}, \mathbf{x}_j) = \sqrt{(\mathbf{x} - \mathbf{x}_j)^{\mathrm{T}} (\mathbf{x} - \mathbf{x}_j)}$ is the Euclidean distance between the unsampled location $\mathbf{x}$ and the sample location $\mathbf{x}_j$. Again, each Voronoi cell includes only one sample data point. VT was originally developed by mathematicians (Voronoi, 1908) and subsequently rediscovered in many fields, e.g., the Thiessen method in meteorology and hydrology. VT was also used to find the optimal equivalent zonation structure in the parameter structure identification problem (Tsai et al., 2003).

There are many zonation methods that can be employed to partition the hydraulic conductivity field into zones when additional geophysical information is available. However, if geophysical information is not available, Voronoi tessellation can be considered as an objective zonation method due to its neutral creation of zones.

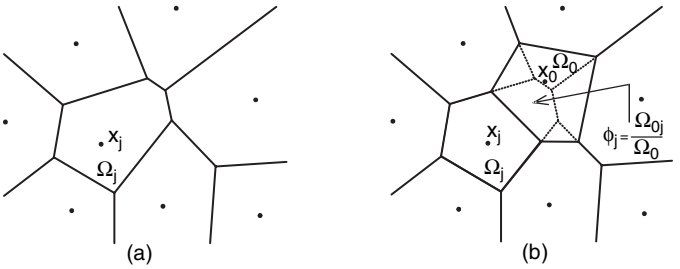


Fig. 18.2 (**a**) Voronoi tessellation based on the sample locations. (**b**) Natural neighbor coordinate

18.3.2 Natural Neighbor Interpolation

Consider that the first-order Voronoi cells (Ω_j) are originally created by the sample data as shown in Fig. 18.2(a). Another first-order Voronoi cell (Ω_0) is created by an unsampled location $\mathbf{x}_0$, which overlaps parts of the first-order Voronoi cells of the sample data as shown in Fig. 18.2(b). A second-order Voronoi cell Ω_{0j} is defined as the overlapping area of Ω_0 to Ω_j according to the following

$$\Omega_{0j} = \{\mathbf{x} | d(\mathbf{x},\mathbf{x}_0) < d(\mathbf{x},\mathbf{x}_j) < d(\mathbf{x},\mathbf{x}_l), \quad \forall l \neq j, 0, l = 1,\cdots,m\} \tag{18.5}$$

The sampled location $\mathbf{x}_j$ is defined as a natural neighbor of $\mathbf{x}_0$ if the second-order Voronoi cell Ω_{0j} (the overlapping area) exists. The NN interpolation method is developed with the basis functions based on the natural neighbor coordinates defined as $\phi_j = \Omega_{0j}/\Omega_0$ (Sibson, 1981; Sambridge et al., 1995; Tsai et al., 2005). For those sampled sites which are not the natural neighbors of $\mathbf{x}_0$, it shows $\phi_j = 0$. The NN interpolation method is

$$\pi_{\mathrm{NN}}(\mathbf{x}_0) = \sum_{j=1}^{\mathrm{nn}} \frac{\Omega_{0j}}{\Omega_0} \pi_j \tag{18.6}$$

where $\{\mathrm{nn}\}$ represents the number of the natural neighbors of $\mathbf{x}_0$, and $\sum_{j=1}^{\mathrm{nn}} \Omega_{0j}/\Omega_0 = 1$.

18.3.3 Conditional Estimation and Conditional Variance Using Indicator GP

Consider $\pi = \ln(K)$ is an intrinsic field with a semivariogram $\gamma(\mathbf{x}_i,\mathbf{x}_j)$ between any two locations $\mathbf{x}_i$ and $\mathbf{x}_j$. The semivariogram is defined as $\gamma(\mathbf{x}_i,\mathbf{x}_j) = \frac{1}{2}\sigma^2[\pi(\mathbf{x}_i) - \pi(\mathbf{x}_j)] = \frac{1}{2}\sigma^2[\pi_i - \pi_j]$. The Indicator GP will be used to estimate the intrinsic field of hydraulic conductivity. From the geostatistical viewpoint, (18.3) is regarded as

the conditional estimation (or conditional mean) conditioned on the sample data (Dagan, 1982). Consider that the estimation error is defined to be the discrepancy between the true value and the estimated value by Indicator GP, i.e., $e = \pi - \pi_{\text{IGP}}$. The expectation of the estimation error is

$$\mathrm{E}[e] = \mathrm{E}[\pi - \pi_{\text{IGP}}] = \mathrm{E}[\pi] - \left\{ \sum_{\substack{j=1 \\ j\neq k}}^{\text{nn}} \phi_j \mathrm{E}\left[\pi_j - \pi_{k(\mathbf{x}_0)}\right] I_j + \mathrm{E}\left[\pi_{k(\mathbf{x}_0)}\right] \right\} \tag{18.7}$$

By definition, the intrinsic field has a constant mean (statistical homogeneity) throughout the region, i.e., $\mathrm{E}[\pi] = \mathrm{E}[\pi_j]$, $j = 1, \ldots, \text{nn}$. Therefore, $\mathrm{E}\left[\pi_j - \pi_{k(\mathbf{x}_0)}\right] = 0$ and $\mathrm{E}[\pi] = \mathrm{E}\left[\pi_{k(\mathbf{x}_0)}\right]$ in (18.7). The unbiasedness $\mathrm{E}[e] = 0$ and statistical homogeneity using Indicator GP are attained regardless of the data indicator values. Especially, the intrinsic field is valid even though the zone structure is used as the conditional estimation.

The estimation error variance (conditional variance) is obtained as follows:

$$\begin{aligned} \sigma^2_{\text{IGP}}[e] = & \sigma^2[\pi - \pi_{\text{IGP}}] = 2 \sum_{\substack{i=1 \\ i\neq k(\mathbf{x}_0)}}^{\text{nn}} \phi_i \tilde{\gamma}(\mathbf{x}_i, \mathbf{x}_0) I_i \\ & - \sum_{\substack{i=1 \\ i\neq k(\mathbf{x}_0)}}^{\text{nn}} \sum_{\substack{j=1 \\ j\neq k(\mathbf{x}_0)}}^{\text{nn}} \phi_i \phi_j \tilde{\gamma}(\mathbf{x}_i, \mathbf{x}_j) I_i I_j + 2\gamma\left(\mathbf{x}_{k(\mathbf{x}_0)}, \mathbf{x}_0\right) \end{aligned} \tag{18.8}$$

where

$$\tilde{\gamma}(\mathbf{x}_i, \mathbf{x}_0) = \gamma(\mathbf{x}_i, \mathbf{x}_0) - \gamma\left(\mathbf{x}_i, \mathbf{x}_{k(\mathbf{x}_0)}\right) - \gamma\left(\mathbf{x}_0, \mathbf{x}_{k(\mathbf{x}_0)}\right)$$

and

$$\tilde{\gamma}(\mathbf{x}_i, \mathbf{x}_j) = \gamma(\mathbf{x}_i, \mathbf{x}_j) - \gamma\left(\mathbf{x}_i, \mathbf{x}_{k(\mathbf{x}_0)}\right) - \gamma\left(\mathbf{x}_j, \mathbf{x}_{k(\mathbf{x}_0)}\right)$$

Equation (18.8) represents the conditional variance for a mixed distribution between the natural neighbor interpolated distribution and the Voronoi zone structure. To avoid confusion, the conditional variance in (18.8) should be distinguished from the kriging variance because the basis functions ϕ_j are not the kriging weights in this study. For the NN interpolation method ($I_j = 1, \forall j$), the conditional variance is

$$\sigma^2_{\text{NN}} = 2 \sum_{\substack{i=1 \\ i\neq k(\mathbf{x}_0)}}^{\text{nn}} \phi_i \tilde{\gamma}(\mathbf{x}_i, \mathbf{x}_0) - \sum_{\substack{i=1 \\ i\neq k(\mathbf{x}_0)}}^{\text{nn}} \sum_{\substack{j=1 \\ j\neq k(\mathbf{x}_0)}}^{\text{nn}} \phi_i \phi_j \tilde{\gamma}(\mathbf{x}_i, \mathbf{x}_j) + 2\gamma\left(\mathbf{x}_{k(\mathbf{x}_0)}, \mathbf{x}_0\right) \tag{18.9}$$

For the Voronoi zone structure, the conditional variance is the variogram $\sigma^2_{\text{VT}} = 2\gamma$. One can show that the conditional variance of Indicator GP is bounded between that of the NN and VT methods, i.e., $\sigma^2_{\text{NN}} \leq \sigma^2_{\text{IGP}} \leq 2\gamma$.

18.4 Identification of Optimal Indicator Values by Genetic Algorithm

An inverse procedure has to be implemented to identify the optimal conditional mean of hydraulic conductivity such that the misfit between simulated and observed groundwater heads is minimized. Traditional geostatistical inversion emphasizes the parameter identification in the semivariogram model (Kitanidis and Vomvoris, 1983). However, many studies have indicated that the semivariogram model parameter identification has little improvement on the modeling agreement to groundwater observations. The semivariogram model parameters can be reasonably estimated merely using the experimental semivariogram. In this study, we will show that the data indicator identification is able to significantly reduce the misfit to groundwater observations. The optimal values of the data indicators are identified through the minimization of the estimated standard deviation of groundwater heads:

$$\min_{I_p \in \{0,1\}} \quad \left[\frac{1}{L-m} \sum_{j=1}^{L} \left(h_j(I_p) - h_j^{\text{obs}} \right)^2 \right]^{1/2} \tag{18.10}$$

where the decision variables are indicator values $I_p,\ p = 1,2,\ldots,m$; L is the number of groundwater head observations h_j^{obs}; and $h_j(I_p)$ is the calculated groundwater heads through a groundwater model. The estimated standard deviation in (18.10) considers the m degrees of freedom due to estimating m unknown I_p (Yeh and Yoon, 1981).

The inverse problem posed in (18.10) is a binary integer nonlinear programming (BINLP) problem, which involves groundwater modeling and which leads to a complicated combinatorial optimization problem. Although the cutting plane algorithm and branch-and-bound method have been proved to be very efficient in integer linear programming (ILP) (Papadimitriou and Steiglitz, 1998), these two methods are anticipated to be not efficient in solving (18.10) because a great number of gradient evaluations through groundwater flow forward modeling are necessary. In addition, many local optima may exist. From our experience, a GA will be suitable for solving this BINLP because a GA is a derivative-free heuristic method and is able to search for a global optimum with multiple searching points (Salomon, 1998).

To adopt GA to our problem, a binary chromosome is designed to represent a feasible solution of (18.10). A chromosome includes m parameters, each of which has 1 bit representing a data indicator with a value of zero or one. Therefore, a total number of possible solutions are 2^m. A GA searches for a global optimal solution using a set of processes and operators analogous to bio-evolution processes (e.g., selection, crossover, mutation, reproduction, and replacement) to improve/maximize the fitness of chromosomes generation by generation (Goldberg, 1989). To our inverse problem, the fitness is given as the negative value of the objective function in (18.10) for the maximization purpose:

$$\text{fitness} = -\left[\frac{1}{L-m}\sum_{j=1}^{L}\left(h_j(I_p) - h_j^{\text{obs}}\right)^2\right]^{1/2} \tag{18.11}$$

Another interpretation of the inverse problem in (18.10) is to search for the optimal number and location of the sample data with the value $I_p = 1$ among the m sample locations. As presented in Section 18.3, the more the indicators with $I_p = 1$, the smaller the conditional variances. Understanding the impact of the number of $I_p = 1$ to the misfit value and to the estimation uncertainty according to the conditional variances is important. Therefore, in addition to obtaining the global optimal solution of data indicators using GA in one shot, this study also transforms the original inverse problem into a stepwise procedure where we systematically increase the number of $I_p = 1$. Given the number of $I_p = 1$, the location optimization problem for the indicators with $I_p = 1$ is posed as the following:

$$\min_{I_p \in \{0,1\}} \quad \left[\frac{1}{L-m}\sum_{j=1}^{L}\left(h_j(I_p) - h_j^{\text{obs}}\right)^2\right]^{1/2} \tag{18.12}$$

subject to

$$1 \le \ell_j \le m;\ j = 1,2,\ldots,n \le m \text{ and } \ell_j \text{ is an integer} \tag{18.13}$$

$$I_p = 1 \quad \text{for } p \in \ell_j \text{ and } I_p = 0 \quad \text{for } p \notin \ell_j \tag{18.14}$$

where the decision variables are $\ell_j, j = 1,2,\ldots,n$, the location indices of the indicators with $I_p = 1$; and n is the number of $I_p = 1$. Again, using traditional combinatorial optimization methods to solve this integer nonlinear programming (18.12–18.14) may not be efficient. Instead, a GA can cope with this problem very easily. In this study, we design a new chromosome to consider the location indices $\{\ell_j\}$. The length of the chromosome is n. In summary, a GA tremendously reduces the complexity of the inverse problems in (18.10) and (18.12–18.14) and is able to obtain the global or near-global optimum solution. Specific information about the GA parameters will be given in the following numerical example.

18.5 Conditional Estimation of Hydraulic Conductivity

18.5.1 Synthetic Confined Aquifer

In the numerical example, a true distribution of hydraulic conductivity (range from 2 to 10 m/day) as shown in Fig. 18.3(a) is given in a two-dimensional confined aquifer, which has a bend of high K values extending from northeast to southwest. Hydraulic conductivity decreases from the high K bend to the northwest and the southeast. This numerical example will be used to investigate the

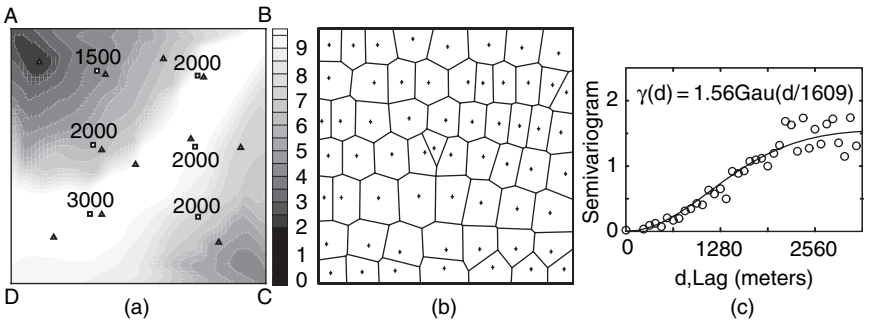


Fig. 18.3 (**a**) The true K distribution, pumping wells (*squares* with pumping rates, m^3/day), and groundwater head boreholes (*triangles*). (**b**) The K sampled locations (*pluses*) and Voronoi tessellation and (**c**) the semivariogram

indicator GP method against the Voronoi zone structure and the NN interpolation method.

The specific information about this numerical experiment is given in Table 18.1. Six pumping wells with corresponding constant pumping rates (the squares and values in Fig. 18.3(a)) are operated for a stress period of 30 days. MODFLOW-2000 (Harbaugh et al., 2000) is used to conduct the groundwater flow simulation, and groundwater heads are collected every day at 11 observation locations. A total of 330 head observations are obtained to which a Gaussian noise term with zero mean and a constant standard deviation is added, where $\sigma_h = 0.01\,\text{m}$ represents the observation error. In addition, 60 hydraulic conductivity values are sampled at the sampled locations shown in Fig. 18.3(b). A Voronoi zone structure is created based on these sampled locations. The experimental semivariogram is obtained using the $60\ln(K)$ values as shown in Fig. 18.3(c) and is described using a Gaussian model: $\gamma(d) = 1.56 \cdot \text{Gau}(d/1609)$, where d is the distance lag (m).

Table 18.1 Characteristics of the synthetic aquifer

Aquifer	Confined
Dimensions	3200 m by 3200 m
Boundaries	$\overline{\text{AD}}$: constant head ($h = 40\,\text{m}$) $\overline{\text{AB}}$, $\overline{\text{BC}}$, and $\overline{\text{CD}}$: impervious
Hydraulic conductivity (K)	2–10 m/day
Specific storage	$10^{-4}\,\text{m}^{-1}$
Number of pumping wells	6 (squares in Fig. 18.3(a))
Number of head observation boreholes	11 (triangles in Fig. 18.3(a))
Number of K measurements	60 (pluses in Fig. 18.3(b))
Discretization	64 rows by 64 columns
Stress period	30 days
Time steps	30

18.5.2 Genetic Algorithm

This study adopts a GA solver (Houck et al., 1995) to search for the best 60 binary values of data indicators I_p such that the misfit to the groundwater head observations is minimized. The fitness in GA is given in (18.11). The total number of head observations is $L = 330$ and the degrees of freedom in the estimation problem are $m = 60$. Given a chromosome, the hydraulic conductivity distribution is calculated by Indicator GP (18.3). We link MODFLOW with GA as a simulation-optimization model to obtain the simulated head observations and to evaluate the fitness. For each GA run, we assign 50 chromosomes in one population to be evolved for 100 generations. The probability for the simple crossover is 0.6 and for the binary mutation is 0.05. A normalized geometric selection method (Houck et al., 1995) is used with a selection probability 0.08. The optimal indicator values are obtained when the GA fitness has no further improvement over one GA run in which the chromosomes are evolved over 100 generations.

18.5.3 Results of Conditional Estimation of Hydraulic Conductivity

Before conducting the GA, the Voronoi zone structure and NN interpolated distribution, as shown in Fig. 18.4(a) and (b), are used to obtain the misfit values as the baseline. The degree of freedom is zero in (18.10) because no estimation is made. The misfit value is 0.0823 m and 0.0863 m for the VT and NN methods, respectively.

We use GA to obtain the optimal binary values of data indicators in one shot. After 7 runs of the GA (35,000 fitness evaluations; note: the total number of enumerations is $2^{60} = 1.15 \times 10^{18}$), the best solution of the binary data indicators is selected from the ultimate generation with a minimum misfit value of 0.0355 m, which is much smaller than that by VT and NN. There are 25 data indicators with $I_p = 1$ and 35 data indicators with $I_p = 0$. The conditional estimation of Indicator GP and the locations of the optimal 0–1 data indicators are shown in Fig. 18.4(c). The optimization results show that Indicator GP significantly improves the fitting of the head observations because Indicator GP allows the optimal adjustment of sampled

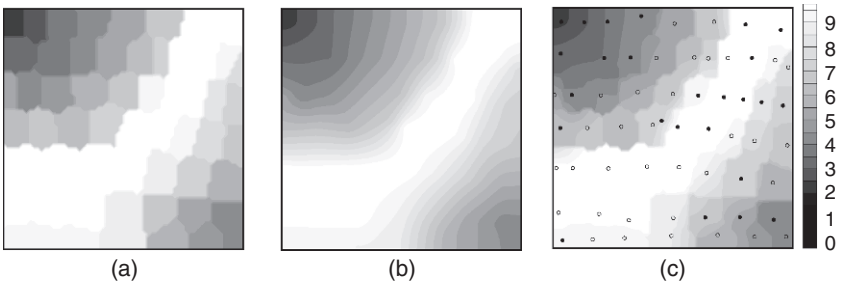


Fig. 18.4 The conditional estimation of hydraulic conductivity using (**a**) VT, (**b**) NN, and (**c**) Indicator GP (*black dots*: $I_p = 1$ and *open circles*: $I_p = 0$)

data influences through the data indicators while the VT and NN methods predetermine the conditional estimations. Again, the conditional estimation of Indicator GP is reasonable because the estimation honors the sampled data and the distribution is between the Voronoi zones and NN interpolated field.

18.5.4 Conditional Variances of Estimation Error and Estimation Uncertainty

The conditional variances of estimation error of the VT, NN, and Indicator GP methods are calculated according to (18.8). As shown in Fig. 18.5, the VT gives the largest conditional variances because only the nearest sampled location is used for estimation. The NN distribution has the smallest conditional variances, where the relatively high variances are present at the boundary. The Indicator GP gives small conditional variances between that of VT and NN.

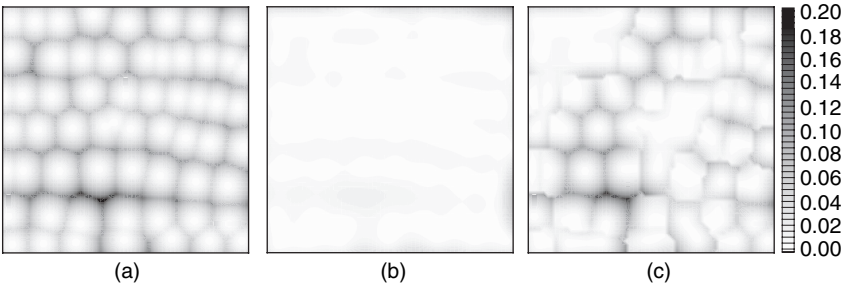


Fig. 18.5 Conditional variance distribution of (**a**) VT, (**b**) NN, and (**c**) Indicator GP

In summary, Indictor GP is able to honor sampled data, minimize the misfit to the observations, and identify a reasonable conditional estimation with small conditional variances.

18.5.5 Misfit and Estimation Uncertainty vs. Quantity of Data Indicators with $I_p = 1$

The stepwise procedure is implemented to evaluate the impact of $I_p = 1$ on the misfit and estimation uncertainty. In the stepwise procedure, the GA optimizes the locations of the $I_p = 1$ data indicators to have the minimum misfit value when the number of $I_p = 1$ is predetermined. As shown in Fig. 18.6, the misfit value of Indicator GP is always smaller than that of the VT and NN methods regardless of the number of $I_p = 1$. The minimum misfit is found when the number of $I_p = 1$ is 25, which is consistent with the GA results in the previous discussion. The overall

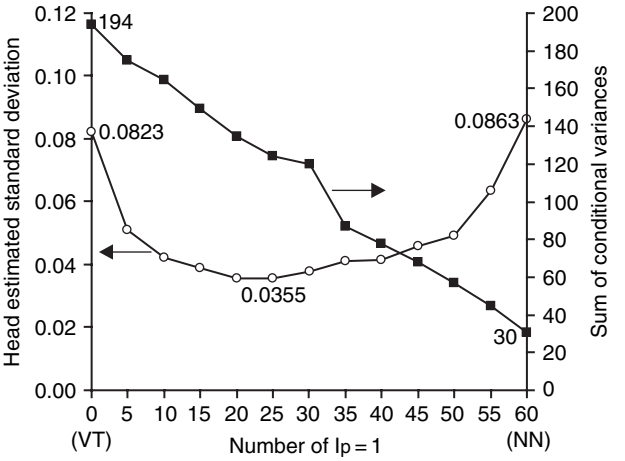


Fig. 18.6 Misfit and estimation uncertainty vs. the number of $I_p = 1$

estimation uncertainty is considered in terms of the sum of the conditional variances $\sum_{i=1}^{64}\sum_{j=1}^{64}\sigma_{ij}^2$. The estimation uncertainty monotonically decreases with the increasing number of $I_p = 1$. Debate on a good representation of conditional estimation may be raised to find a trade-off between the small misfit value and small estimation uncertainty. Some statistical information criteria (e.g., Carrera and Neuman, 1986) can be adopted to determine the best trade-off.

18.6 Conclusions

This study has developed an indicator-generalized parameterization method for a better representation of the conditional estimation of hydraulic conductivity in a random field. We have combined the Voronoi zone structure and a natural neighbor interpolation method to capture non-smoothness of heterogeneity. The conditional variance of estimation error using Indicator GP has been derived for an intrinsic field of log-hydraulic conductivity. The complexity of the inverse problem of identifying the optimal binary values of data indicators has been tremendously eliminated by using a genetic algorithm. It concludes that the Indicator GP is able to find the optimal conditional estimation of hydraulic conductivity between that of the VT and NN methods with the minimal misfit value of groundwater head observations. The estimation error of the Indicator is bounded between that of the VT and NN methods. A trade-off between the small misfit value and the small estimation uncertainty needs further study.

Acknowledgements This research was supported in part by Louisiana Board of Regents under award LEQSF(2005–08)-RD-A-12 and Department of the Interior, U.S. Geological Survey under

grant no. 05HQGR0142 in collaboration between USGS and LSU A&M College. This manuscript is submitted for publication with the understanding that the U.S. government is authorized to reproduce and distribute reprints for governmental purposes. The views and conclusions contained in the document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. government.

References

Carrera J, Neuman SP (1986) Estimation of aquifer parameters under transient and steady-state conditions 3. Application to synthetic and field data. Water Resources Research 22: 228–242

Cherkassky V, Mulier F (1998) Learning from data: Concepts, theory, and methods, John Wiley and Sons Inc., New York, USA

Dagan G (1982) Stochastic modeling of groundwater-flow by unconditional and conditional probabilities 1. Conditional simulation and the direct-problem. Water Resources Research 18: 813–833

Delhomme JP (1979) Spatial variability and uncertainty in groundwater flow parameters: a geostatistical approach. Water Resources Research 15: 269–280.

Goldberg DE (1989) Genetic algorithms in search, optimization, and machine learning. Addison-Wesley, Massachusetts

Harbaugh AW, Banta ER, Hill MC, McDonald MG (2000) MODFLOW-2000, the U.S. Geological Survey modular ground-water model – User guide to modularization concepts and the groundwater flow process. U.S. Geological Survey Open-File Report 00-92

Houck CR., Joines JA, Kay MG (1995) A genetic algorithm for function optimization: A Matlab implementation. Technical Report, NCSU-IE TR 95-09, North Carolina

Kitanidis PK, Vomvoris EG (1983) A geostatistical approach to the inverse problem in groundwater modeling (steady state) and one-dimensional simulations. Water Resources Research 19: 677–690

McLaughlin D, Townley LR (1996) A reassessment of the groundwater inverse problem. Water Resources Research 32: 1131–1161

Papadimitriou CH, Steiglitz K (1998) Combinatorial optimization: Algorithms and complexity. Dover Publications Inc., Mineola, New York

Salomon R (1998) Evolutionary algorithms and gradient search: similarities and differences. IEEE Transactions on Evolutionary Computation 2: 45–55

Sambridge M, Braun J, Mcqueen H (1995) Geophysical parameterization and interpolation of irregular data using natural neighbors. Geophysical Journal International 122: 837–857

Sibson R (1981) Chapter 2: A brief description of natural neighbour interpolation. In: Barnett V (ed) Interpreting multivariate data, John Wiley & Sons, New York, USA

Tsai FTC, Sun NZ, Yeh WWG (2003) A combinatorial optimization scheme for parameter structure identification in ground water modeling. Ground Water 41: 156–169

Tsai FTC, Sun NZ, Yeh WWG (2005) Geophysical parameterization and parameter structure identification using natural neighbors in groundwater inverse problems. Journal of Hydrology 308: 269–283

Tsai FTC, Yeh WWG (2004) Characterization and identification of aquifer heterogeneity with generalized parameterization and Bayesian estimation. Water Resources Research 40: W10102, doi:10.1029/2003WR002893

Yeh WWG, Yoon YS (1981) Aquifer parameter-identification with optimum dimension in parameterization. Water Resources Research 17: 664–672

Voronoi G (1908) Nouvelles applications des parametres continus a la theroie des formes quadratiques, deuxieme memoire, recherches sur les parallelloedres primitifs. Journal fur die Reine und Angewandte Mathematik 134: 198–287

Chapter 19
Fitting Hydrological Models on Multiple Responses Using the Multiobjective Evolutionary Annealing-Simplex Approach

A. Efstratiadis and D. Koutsoyiannis

Abstract Most complex hydrological modeling schemes, when calibrated on a single observed response (e.g., river flow at a point), provide poor predictive capability, due to the fact that the rest of the variables of basin response remain practically uncontrolled. Current advances in modeling point out that it is essential to take into account multiple fitting criteria, which correspond to different observed responses or to different aspects of the same response. This can be achieved through multiobjective calibration tools, thus providing a set of solutions rather than a single global optimum. In addition, actual multiobjective optimization methods are rather inefficient, when real-world problems with many criteria and many control variables are involved. In hydrological applications there are some additional issues, due to uncertainties related to the representation of complex processes and observation errors. The multiobjective evolutionary annealing-simplex (MEAS) method implements an innovative scheme, particularly developed for the optimization of such problems. Its features and capabilities are illustrated by solving a challenging parameter estimation problem, dealing with hydrological modeling and water resource management in a karstic basin in Greece.

Keywords Parameter estimation · conjunctive hydrological models · evolutionary multiobjective optimization · irregular pareto front · model uncertainty

19.1 Introduction

The parameter estimation procedure of hydrological models aims toward a faithful reproduction of observed outputs, in addition to establishing "behavioral" (i.e.,

A. Efstratiadis
Department of Water Resources, School of Civil Engineering, National Technical University of Athens, Heroon Polytechneiou 5, 157 80 Zographou, Greece, e-mail: andreas@itia.ntua.gr

D. Koutsoyiannis
Department of Water Resources, School of Civil Engineering, National Technical University of Athens, Heroon Polytechneiou 5, 157 80 Zographou, Greece, e-mail: andreas@itia.ntua.gr

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

realistic, reliable and stable) parameters. In addition, it is recognized that modeling schemes with more than five to six parameters, when calibrated on a single observed response (typically, a river flow time series at a single point), often provide poor predictive capacity (Wagener et al., 2001). This is due to the fact that the rest of the basin responses remain practically uncontrolled, thus leading to non-realistic representation of the physical processes (Rozos et al., 2004). Given that hydrological models tend to be too complex and, thus, non-parsimonious in parameters, it is essential to consider multiple fitting criteria, corresponding to different observed responses (a "joint-calibration" approach, according to Kuczera and Mroczkowski, 1998) or to various aspects of the same response, by means of multiple error measures, data split, etc. Current advances review the calibration problem from a multiobjective point-of-view, where various fitting criteria are introduced as elements of a vector objective function, thus leading to a set of optimal solutions rather than a single global optimum (Yapo et al., 1998; Madsen, 2000; Khu and Madsen, 2005). Such an analysis provides insight into the manner in which the model needs to be improved and into the confidence that can be ascribed to the model predictions (Gupta et al., 1998).

In this chapter, a short presentation of a new multiobjective optimization method, the multiobjective evolutionary annealing-simplex (MEAS) algorithm, is given, focused on hydrological problems. The methodology is applied within a complex case study, involving the parameter estimation procedure for a combined hydrological–water management model. Through this study, characteristic issues of multiobjective calibration are highlighted, and its practical use is emphasized, as guidance to the best-compromise parameter set.

Apart from the introduction (Sect. 19.1), the chapter contains five sections. Section 19.2 is a synoptic review of current advances of multiobjective optimization. Section 19.3 explains the fundamentals of the MEAS algorithm. Section 19.4 deals with the case study, whereas Sect. 19.5 summarizes the conclusions and discusses the applicability of the multiobjective calibration approach in complex hydrological applications.

19.2 Overview of Multiobjective Optimization Techniques

19.2.1 Problem Formulation and Definitions

A multiobjective calibration problem can be stated as the simultaneous optimization (for convenience and without loss of generality, minimization) of m numerical criteria (objective functions) with respect to a vector of control variables $\mathbf{x} \in X$, i.e.,

$$\text{minimize } \mathbf{F}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})] \tag{19.1}$$

where $X \in \mathscr{R}^n$ is the feasible control space.

Generally, the criteria represented by the components $f_i(\mathbf{x})$ of the objective function are conflicting and, therefore, a feasible point cannot simultaneously optimize all of them. Thus, we look for acceptable trade-offs rather than a unique solution, according to the fundamental concept of *Edgeworth–Pareto optimality* (commonly referred to as *Pareto optimality*), introduced within the theory of welfare economics at the end of the nineteenth century. In particular, we define a vector of control variables $\mathbf{x}^*$ to be *Pareto optimal* if there does not exist another feasible vector $\mathbf{x}$ such that $f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*)$ for all $i = 1, \ldots, m$ and $f_i(\mathbf{x}) < f_i(\mathbf{x}^*)$ for at least one i. The above definition implies that $\mathbf{x}^*$ is Pareto optimal if there is no feasible vector that would improve some criterion without causing a simultaneous deterioration of at least one other criterion.

The concept of Pareto optimality leads to a set of feasible vectors, called the Pareto set and symbolized $X^* \subset X$; all Pareto-optimal vectors $\mathbf{x}^* \in X^*$ are called *non-inferior* or *non-dominated*. The image of the non-dominated set in the objective space is called the Pareto front, symbolized F^*. From a mathematical point-of-view, and in the absence of further information, all non-dominated solutions are assumed equivalent.

19.2.2 Classical Approaches

Optimization problems involving multiple and conflicting objectives have been traditionally handled by combining the objectives into a scalar function and, next, solving the resulting single-optimization problem. The combination schemes, usually referred to as aggregating functions, are the oldest mathematical programming approaches, since they are derived from the Kuhn–Tucker conditions for non-dominated solutions. Aggregation functions may be linear (the well-known weighting method) or non-linear (e.g., the goal-programming method); both may be used within any global optimization algorithm to provide a unique point of the Pareto set. By changing the arguments of the aggregating function (e.g., the weighting coefficients), one can obtain representative points of the Pareto set. But this procedure is computationally inefficient; moreover, when optimization criteria are non-commeasurable, it is necessary to provide some information on the range of objectives, to avoid having one of them dominate the others. For a comprehensive review of classical methods and their application in water resource problems, the reader may refer to the book of Cohon (1978).

19.2.3 Multiobjective Evolutionary Algorithms

The main advantage of evolutionary methods is their ability to provide multiple Pareto-optimal solutions in a single run. Since evolutionary algorithms work with a population of points, they can be adapted to maintain a diverse set of solutions.

The first pioneering studies appeared in the mid-1980s; the most representative example is the vector-evaluated genetic algorithm (VEGA) of Schaffer (1984), where the selection mechanism of a typical genetic algorithm was modified to spread the search toward different regions of the Pareto set, by dividing the population into subsets and switching objectives. But the fundamental concept of Pareto dominance was incorporated later, in the mid-1990s (Fonseca and Fleming, 1993; Srinivas and Deb, 1994; Horn et al., 1994). These methods employ ranking procedures, where individuals are evaluated according to the principle of Pareto optimality, thus conducting search toward the Pareto front; in addition, fitness sharing techniques are used to maintain population diversity, thus favoring the generation of well-distributed sets. We should classify to the same category the MOCOM-UA method of Yapo et al. (1998), which is an extension of the well-known SCE-UA algorithm of Duan et al. (1992), employing a Pareto ranking procedure within a simplex-based searching pattern. More recent advances (e.g., Zitzler and Thiele, 1999; Knowles and Corne, 2000; Deb et al., 2002; Zitzler et al., 2002) give further emphasis on efficiency, by using faster ranking techniques, clustering methods and, primarily, elitism mechanisms to retain the non-dominated solutions found so far. Coello (2005) presents a brief evaluation of the relevant research in multiobjective evolutionary optimization. An extended database regarding relative references and software tools can be found at http://www.lania.mx/∼ ccoello/EMOO/.

19.2.4 Some Drawbacks in Calibration Problems

Currently, there is an increasing interest in applying multiobjective evolutionary algorithms in hydrological applications, most of them referring to calibration studies (Ritzel et al., 1994; Cieniawski et al., 1995; Yapo et al., 1998; Madsen, 2000; Reed et al., 2001; Erickson et al., 2002; Khu and Madsen, 2005). It is recognized that the task is computationally demanding, especially in the case of complex models with many parameters. Moreover, the performance of evolutionary methods with continuous parameters is rather inefficient, due to the generating schemes that are adopted from genetic algorithms. For instance, it has been detected that crossover operators based on the exchange of coordinate values of "parents" usually lead to redundant "offspring" (Solomatine, 1998).

It is also important to notice that within a multiobjective calibration procedure, not all trade-offs may be acceptable from the engineering point-of-view. For example, one should reject parameter sets providing extreme performance (i.e., too good against some criteria, but too bad against some other), albeit being mathematically acceptable according to the concept of dominance. Finally, despite the efforts of some researchers to incorporate users' preferences as to narrow the search (e.g., using constraints or penalty functions), a systematic procedure helping to "capture" the best-compromise solution is missing. Note that such a unique parameter set might be necessary in real-world problems for operational purposes (i.e., forecasting).

19.3 The Multiobjective Evolutionary Annealing-Simplex Algorithm

19.3.1 Main Concepts

The multiobjective evolutionary annealing-simplex (MEAS; Efstratiadis, 2008; Efstratiadis and Koutsoyiannis, 2005) method briefly presented here is a general-purpose tool that places emphasis on the specific features of hydrological models. The algorithm embodies two phases: in the evaluation phase a fitness measure is assigned to all population members, whereas in the evolution phase new individuals are generated on the basis of their fitness values.

The algorithm seeks behavioral parameter sets, by introducing the concept of "suitability", denoting a physically reasonable Pareto subset. Particularly, the user specifies acceptable thresholds for fitting criteria, to avoid the generation of extreme solutions. Hence, only part of the Pareto front is approximated in an attempt to "surround" the best-compromise set. The ranking procedure, which guarantees the preservation of representative non-dominated solutions, is based on the strength-Pareto approach (Zitzler and Thiele, 1999; Zitzler et al., 2002), where some modifications are employed to better handle problems with more than two criteria.

The population is evolved according to a simplex-annealing approach (Press et al., 1992, pp. 451–455); offspring generation is implemented on the basis of a downhill simplex pattern, whereas an adaptive annealing cooling schedule is used to control the degree of randomness during evolution. Most of the generating mechanisms are adapted from an earlier single-optimization version of the method, which has been proved effective and efficient for a broad class of water resource problems (Efstratiadis and Koutsoyiannis, 2002; Rozos et al., 2004).

19.3.2 Initialization

For each control variable (parameter), the user specifies two ranges: an "external" range that may correspond to the physical limits of parameters and an "internal" range that corresponds to more realistic limits, according to the modeler's experience and intuition. The initial population is uniformly sampled within the internal range, whereas search can be expanded to the external one. Thus, the optimization procedure is protected from being trapped in the boundaries of an arbitrarily specified "reasonable" search space. The population size N is a user-defined constant and, due to the necessity of building a simplex, must be at least equal to $n + 1$, where n is the dimension of the search space (i.e., the number of parameters). It is obvious that the larger the population size, the better the approximation of the Pareto front.

19.3.3 Evaluation Procedure

At each generation, a penalty value $p(i)$ is assigned to each individual i, according to its "relative" position in the objective space, i.e., its position against all the other individuals. It includes two components: (a) a dominance term and (b) a suitability term.

The *dominance term* is a real positive number with integral and fractional parts computed separately as explained below. For each individual, the integral part (rank) is computed by taking into account both dominating and dominated points, through a two-step procedure. First, a "strength" value is assigned that is equal to the number of dominated points by the evaluated point. Next, the rank is computed by summing the strength values of all dominators. An example is given in Fig. 19.1. By definition, non-dominated points have zero rank. It can be proved that this scheme, adapted from the SPEA and SPEA-II methods, not only provides a large variety of rank values (larger than any other known ranking algorithm), but also incorporates a sort of "niching" mechanism, thus preserving population diversity (Zitzler and Thiele, 1999).

According to the Pareto optimality definition, if one point outperforms another one against some but not all criteria, then the two alternatives are indifferent; thus they should have the same rank value. Yet, in high-dimensional problems, it is necessary to review the concept of dominance as the only evaluation principle; otherwise the optimal front becomes too extended, even almost identical to the objective space, as reported by Coello (2005). To avoid this, a "discard" mechanism is implemented among indifferent solutions, by comparing them on the basis of each specific criterion. According to this, a fractional term, proportional to the average number of

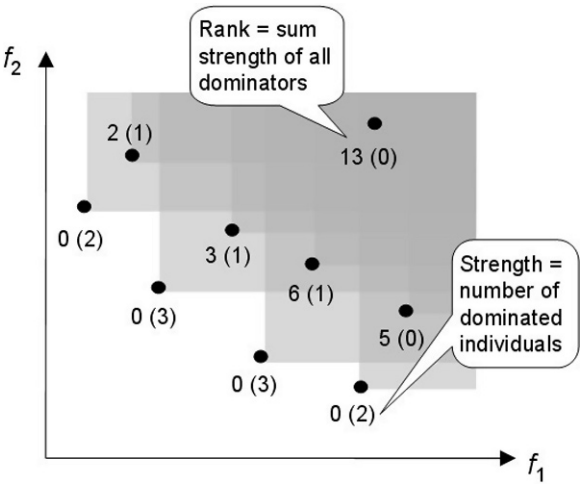


Fig. 19.1 An example of assigning strength (in *parenthesis*) and rank values

criteria against which the individual is superior, is added to the rank measure, thus resulting in an increasing variety of performance values.

On the other hand, the *suitability term* is assigned to favor the generation of solutions with intermediate performance that are of significant interest in case of some practical applications (including model calibration problems). This is done by introducing a constraint vector $\mathbf{e} = (e_1, \ldots, e_m)$, standing for the boundaries of a desirable (physically acceptable) region of the objective space. The computational procedure is as follows. First, the maximum dominance value, $d_{\max}$, within the population is computed. Next, each individual is checked to see whether it lies within the desirable region. If $f_{ij} > e_j$ for the jth criterion, a distance penalty is added to the dominance term. All non-suitable individuals are further penalized by adding $d_{\max}$, thus becoming worse than any other suitable individual, either dominated or not.

19.3.4 Selection Through a Simulated Annealing Strategy

An evolutionary scheme is employed, where the population is changed when a single "offspring" is produced. The individual to "die" is picked from a randomly selected "mating" sub-population comprising $n+1$ points (a simplex), on the basis of a simulated annealing strategy. The latter is implemented by adding to the penalty measure, $p(i)$, a stochastic component, $s(i) = rT$, where r is a uniform random number in the range [0,1] and T is the system's state of search known as "temperature" (large temperature means dominance of the random component). The sum $p(i)+s(i)$ stands as an objective function to minimize, which is updated at each generation. A similar scheme was successfully implemented within the single-objective version of the method.

The temperature, which is initially set equal to the difference between the maximum and minimum fitnesses among the population, is automatically regulated to ensure convergence toward the Pareto front. Each time a non-dominated point is found, it is slightly reduced by a factor λ, whereas it is not allowed to be less than a limit β, thus avoiding early convergence. The larger the temperature, the most probable the acceptance of an "uphill" transition, i.e., a movement in the opposite direction of the current non-dominated front, which makes search more flexible, thus enabling a detailed exploration of the search space.

19.3.5 Evolution Procedure

After selecting the individual to die, a recombination procedure is implemented, following a simplex-evolving pattern. It is based on the well-known local optimization algorithm of Nelder and Mead (1965), in which some modifications were made to handle the peculiarities of multiobjective search. Specifically, four transitions are provided, i.e., reflection, multiple expansion, outside contraction and inside

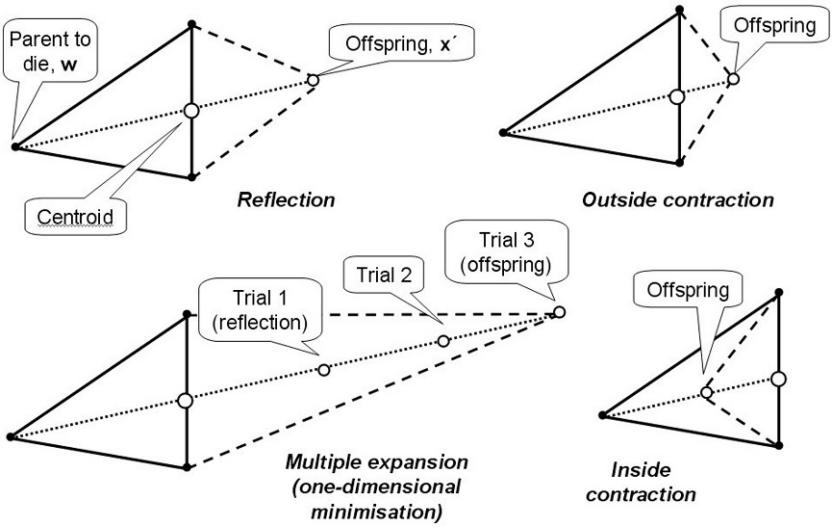


Fig. 19.2 Feasible simplex transitions in the 2D search space

contraction (Fig. 19.2). The multiple expansion transition is a line-minimization procedure, implemented through subsequent trials, thus ensuring a fast exploration of the convex regions of the search space. On the other hand, the shrinkage transition of the original Nelder–Mead scheme is abandoned, because the question is to maintain a well-distributed set, instead of converging to the unique global extreme.

If any of the possible simplex transitions are successful, the offspring is generated through mutation. The algorithm provides two types of mutation: a quasi-random scheme, on the basis of the statistical characteristics of the actual population, and a pure random scheme, based on a uniform sampling of the entire feasible space. The search is interrupted when the number of function evaluations exceeds a user-specified limit, provided that all population members are non-dominated and lie within the desirable objective region.

19.4 Case Study

19.4.1 Description of the Study Area

The Boeoticos Kephisos river basin lies on the Eastern Sterea Hellas, north of Athens, and drains a closed area (i.e., without an outlet to the sea) of 1956 km^2 (Fig. 19.3). The catchment geology comprises heavily karstified limestone, mostly developed on the mountains, and alluvial deposits, covering the plain areas . Due to its karstic background, the watershed has a significant groundwater yield. The main discharge points are large springs in the upper and middle parts of the basin that

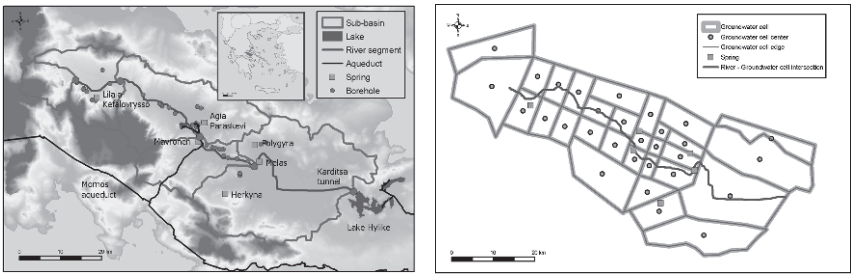


Fig. 19.3 The surface (*left*) and groundwater (*right*) schematization of the Boeoticos Kephisos basin

account for more than half of the catchment runoff. Moreover, an unknown amount of groundwater is conducted to the sea.

The hydrosystem serves multiple water uses. Specifically, through an extended drainage network, the entire surface resources are diverted to the neighboring Lake Yliki, one of the major water storage projects of Athens, which is characterized by significant losses to the sea due to its karstic background. In addition, important supply boreholes are located at the middle course, just upstream of the Mavroneri springs; these are activated in case of emergency and affect significantly the flow regime of the groundwater system. In addition to drinking water, the surface and groundwater resources of the basin are used for irrigation; the total annual demand is more than 220 hm^3.

19.4.2 The Modeling System

The hydrosystem has been extensively studied in the past (Rozos et al., 2004), but the simulation presented here was implemented through a newly developed computer-based tool, named HYDROGEIOS, which integrates a conjunctive (i.e., surface and groundwater) hydrological model within a systems-oriented water management scheme (Efstratiadis et al., 2007) and implements multicriteria model fitting. The modeling scheme follows a semi-distributed approach, aiming at estimating the available water resources at characteristic sites (nodes) of the river basin and the underlying aquifer. Regarding the methodological concept, parameters are assigned on the basis of the main physical characteristics affecting the hydrological processes. This favors the maintenance of a parsimonious structure, since the parameterization is consistent with the available data.

Geographical input data include the river network, the sub-basins upstream of each river node and the aquifer discretization in polygonic cells, representing conceptual groundwater tanks. Additional layers of distributed geographical information, such as geology, land cover and terrain slope, are used to define the hydrological response units (HRUs); the latter are spatial components that correspond

to areas of homogenous hydrological characteristics. On the other hand, input data for artificial components include water abstraction facilities, aqueducts and demand points. Dynamic data include precipitation and potential evapotranspiration series, given at a sub-basin scale, and target demand series, referring to water needs as well as to various water management constraints (e.g., flow preservation).

Various modules are combined to represent the key processes in the watershed, i.e., (a) a conceptual soil moisture accounting model with six parameters per HRU; (b) a groundwater model, based on a modified finite-volume numerical method, where two parameters are assigned to each groundwater tank (Rozos and Koutsoyiannis, 2006); and (c) a water management model, inspired from graph theory, which estimates the optimal hydrosystem fluxes, satisfying both physical constraints and target priorities and simultaneously minimizing costs (Efstratiadis et al., 2004). Model outputs include discharges through the river network, spring flows, groundwater levels and water abstractions. The program provides a variety of goodness-of-fitting criteria, which may be combined following both a single- and a multi-objective calibration approach, through the MEAS algorithm.

19.4.3 Input Data and Schematization

As illustrated in Fig. 19.3 (left), the river network comprises a main branch, divided into four segments, and five sub-basins upstream of or in between the corresponding nodes. HRUs are produced as the union of two geographic layers; the first represents three categories of geological formations, whereas the second represents two categories of terrain slope. Finally, 35 cells represent the groundwater flow field (Fig. 19.3, right). Some are virtual cells, simulating either in-basin outflow sites (springs) or accumulating tanks, draining the basin leakages to the sea.

For the above schematization, the total number of unknown parameters is more than 100. Thus, it is essential to use multiple criteria within calibration, to avoid model over-parameterization and properly representing important characteristics of the physical system that are reflected in the observations. The latter refer to systematic (daily) discharge measurements at the basin outlet (Karditsa tunnel) and sparse (one to two per month) measurements downstream of the six main karstic springs; these raw data were used to construct the monthly hydrographs at seven discharge points, for a 10-year period (Oct. 1984–Sep. 1994).

19.4.4 Model Setup

Initially, model parameters were estimated through a single-optimization approach, based on a weighted objective function, where the coefficients of efficiency (CE) for the seven hydrographs (usually referred as Nash–Sutcliffe measure), as well as other fuzzy performance measures, were aggregated. The latter refer to penalty measures, assigned to reproduce flow intermittencies and to prohibit abnormal trends

regarding the simulated groundwater levels. A combined strategy was employed, coupling both automatic and manual calibration methods. This was significantly time-consuming, whilst leading to a satisfactory best-compromise parameter set. Rozos et al. (2004) provide a comprehensive explanation of this hybrid calibration procedure, employed within an earlier version of the model, applied for the same study area. The model was calibrated on the first 6 years (Oct. 1984–Sep. 1990) and validated on the other 4 years (Oct. 1990–Sep. 1994). One full simulation run accounted for about 0.5 s, on a Pentium IV computer.

Next, a multiobjective calibration approach was implemented, by separating the seven CEs (each representing a fitting measure of simulated and observed time series) and selecting the most important parameters to optimize; these were the soil moisture capacity and recession rates for percolation, for the six HRUs, and the conductivities of the virtual cells representing spring dynamics. The rest of the parameters were assumed known, by assigning the optimal values obtained from the aforementioned single-objective calibration scenario. Hence, a challenging multiobjective optimization problem was formulated, comprising 18 control variables and 7 error functions to minimize, defined as $f_i = 1 - \mathrm{CE}_i$. A population size of 100 and a limit of 5000 trials were set. Two multiobjective calibration scenarios were examined. In the first one, the objective space was left unrestricted, whereas in the second one the objective space was restricted, by accepting only positive values of the CE for all measurement sites.

19.4.5 Results and Discussion

Figure 19.4 shows some characteristic trade-off curves for the first multiobjective calibration scenario, representing specific cross-sections of the Pareto front (note that by taking into account more than two criteria, i.e., $m > 2$, the Pareto front is a hypersurface in the m-dimensional objective space; thus, non-dominated points do not lie on a common curve). Some of the cross-sections have a particularly irregular shape, either very steep, thus leading to an almost right angle, or asymmetrically spread. These are strong indications of unacceptable trade-offs, since a small improvement of some criteria results in significant degradation of other ones. Indeed, as shown in Fig. 19.6, the range of some determination coefficients is too extended, containing even high negative values, which obviously correspond to non-behavioral parameter sets. The inadmissible trade-offs mainly refer to the significant karstic springs of Melas and Polygyra, for which there is too little knowledge about the flow generation mechanism (however, it was important to include these springs in the calibration, even for reproducing their average hydrological regime). For instance, negative correlations between the spring runoff and the precipitation highlight the complexity of the related physical processes. Uncertainties are also due to observation errors, e.g., spring hydrographs that were estimated on the basis of sparse measurements. Therefore, the irregularities of the Pareto front, which made

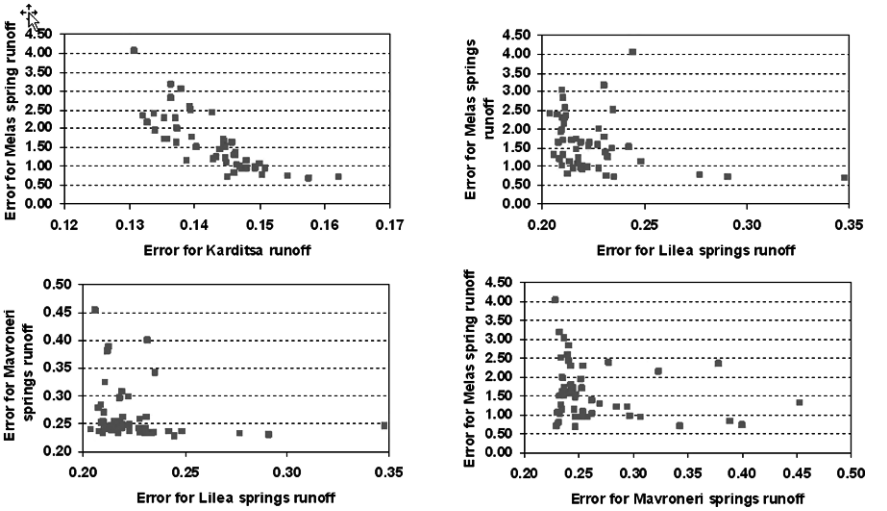


Fig. 19.4 Characteristic trade-offs for multiobjective calibration scenario 1 (unrestricted objective space)

it so difficult to detect the best-compromise parameter set, are explained by the ill-posed model structure and data.

The assumption of a bounded objective space, by accepting only positive determination coefficient values (scenario 2), resulted in a much narrower range of optimal trade-offs, as illustrated in Fig. 19.5. Interestingly, these bounds enclose the

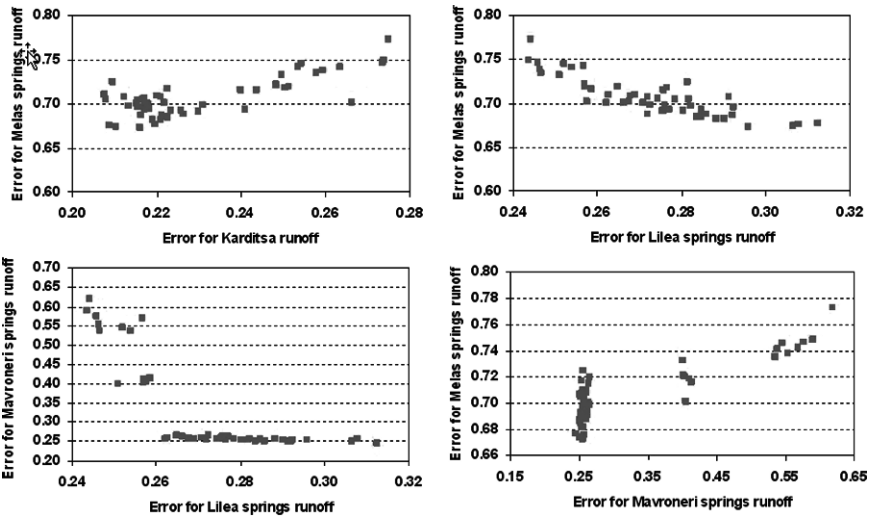


Fig. 19.5 Characteristic trade-offs for multiobjective calibration scenario 2 (restricted objective space)

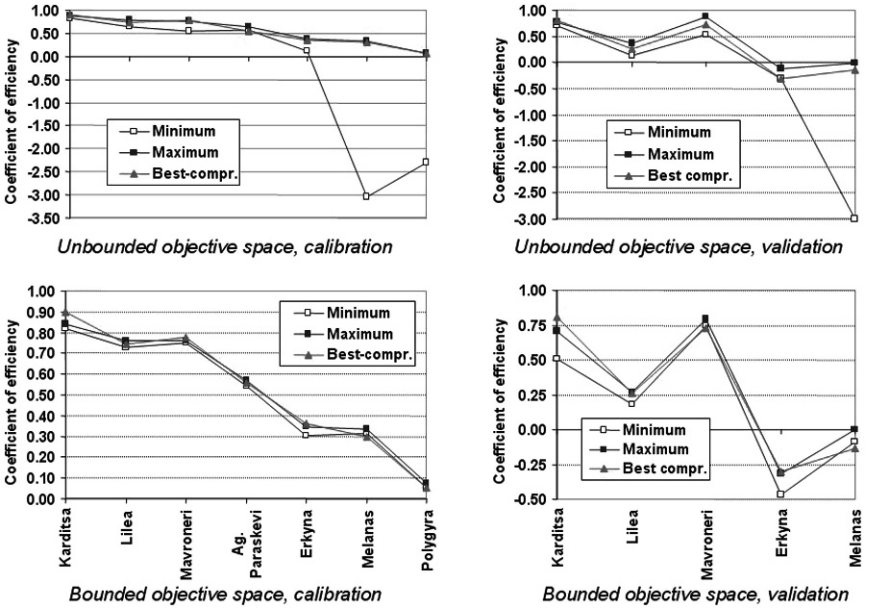


Fig. 19.6 Minimum and maximum coefficients of efficiency obtained by the two multiobjective calibration scenarios and best-compromise values obtained by the single-objective optimization scenario

best-compromise values obtained by the single-objective optimization scenario for all control sites except the basin outlet, where the optimal value found is slightly better than the extreme of the Pareto front. The picture is similar for both the calibration and validation periods (Fig. 19.6).

An important issue to mention is that, in contrast to the Pareto front, the Pareto set, i.e., the set of the optimal parameter vectors, was much less scattered. This indicates the sensitivity of the parameters, especially those representing groundwater conductivities, whose physical limits fluctuate in a wide range of orders of magnitude, in contrast to the surface model parameters that are of the same order of magnitude. It is a well-known problem of conjunctive hydrological modeling, also appearing in the multi-response calibration approach.

19.5 Conclusions

The MEAS algorithm is an innovative technique, suitable for hydrological problems, which combines (a) a fitness evaluation procedure based on a strength-Pareto approach and a suitability concept, (b) an evolving pattern based on the downhill simplex method and (c) a simulated annealing strategy that controls randomness during evolution. It provides innovations, such as the double boundaries of

the search space and the concept of feasibility in the objective space, thus helping experts to effectively handle problems of irregularly conflicting criteria.

When calibrating hydrological models that generate distributed fluxes, it is necessary to take advantage of all available data reflecting the basin responses, by fitting their parameters on multiple observations. The application of the HYDROGEIOS model to the Boeoticos Kephisos basin is a characteristic example of how to treat such problems. The case study indicated that multiobjective calibration may be a useful tool for (a) exploring structural and data uncertainties that are encapsulated in the Pareto front irregularities, (b) investigating acceptable trade-offs between the fitting criteria and (c) guiding the search toward promising (from a hydrological point-of-view) areas of both the objective and the parameter space. The easy implementation of the method and the limited computational effort (5000 trials were sufficient to capture the best-compromise solution, earlier detected after a not inconsiderable number of single-objective optimizations) were a real surprise. A next step may be the incorporation of the multiobjective search routines into a generalized methodological framework, where the parameter estimation procedure will be implemented in an interactive manner, to better account for the hydrological experience.

Acknowledgements The research of this article was performed within the framework of the scholarship project "Heracleitos", elaborated by the Ministry of Education and co-funded by the European Social Fund (75%) and National Resources (25%). The authors are grateful to R. Abrahart and D. Solomatine for the invitation, and the two anonymous reviewers, for their positive comments that were very helpful for an improved presentation of the manuscript.

References

Cieniawski SE, Eheart JW, Ranjithan S (1995) Using genetic algorithms to solve a multiobjective groundwater monitoring problem. Water Resour Res 31(2): 399–409

Coello CA (2005) Recent trends in evolutionary multiobjective optimization. In: Abraham A, Jain L, Goldberg R (eds) Evolutionary Multiobjective Optimization: Theoretical Advances and Applications. Springer-Verlag, London, pp 7–32

Cohon JI (1978) Multiobjective Programming and Planning. Academic Press, NY

Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comp 6(2): 182–197

Duan Q, Sorooshian S, Gupta V (1992) Effective and efficient global optimization for conceptual rainfall-runoff models. Water Resour Res 28(4): 1015–1031

Efstratiadis A (2008) Non-linear methods in multiobjective water resource optimization problems, with emphasis on the calibration of hydrological models. Ph.D. thesis, Department of Water Resources and Environmental Engineering – National Technical University of Athens

Efstratiadis A, Koutsoyiannis D (2002) An evolutionary annealing-simplex algorithm for global optimisation of water resource systems. In: Falconer RA, Lin B, Harris EL, Wilson CA, Cluckie ID, Han D, Davis JP, Heslop S (eds) Hydroinformatics 2002 (Proc. Fifth Int. Conf. on Hydroinformatics, Cardiff, UK). International Water Association Publishing, 2, pp 1423–1428

Efstratiadis A, Koutsoyiannis D (2005) The multiobjective evolutionary annealing-simplex method and its application in calibrating hydrological models. In: 2nd General Assembly of the European Geosciences Union, Geoph Res Abstr 7, Vienna, 04593, European Geosciences Union

Efstratiadis A, Koutsoyiannis D, Xenos D (2004) Minimising water cost in the water resource management of Athens. Urb Water J 1(1): 3–15

Efstratiadis A, Nalbantis I, Koukouvinos A, Rozos E, Koutsoyiannis D (2007) HYDROGEIOS: A semi-distributed GIS-based hydrological model for disturbed river basins. Hydrol Earth System Sci Disc 4(3): 1947–1997

Erickson M, Mayer A, Horn J (2002) Multi-objective optimal design of groundwater remediation systems: application of the niched Pareto genetic algorithm (NPGA). Adv Water Resour 25: 51–65

Fonseca CM, Fleming PJ (1993) Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. In: Proc. Fifth Inter. Conf. on Genetic Algorithms. Morgan Kaufmann Publishers, San Mateo, CA, pp 416–423

Gupta HV, Sorooshian S, Yapo PO (1998) Toward improved calibration of hydrologic models: Multiple and non-commensurable measures of information. Water Resour Res 34(4): 751–763

Horn J, Nafpliotis N, Goldberg DE (1994) A niched Pareto genetic algorithm for multiobjective optimization. In: Proc. First IEEE Conf. on Evolutionary Computation, IEEE World Congress on Computational Intelligence, 1, pp 82–87

Khu ST, Madsen H (2005) Multiobjective calibration with Pareto preference ordering: An application to rainfall-runoff model calibration. Water Resour Res 41: W03004 (doi: 10.1029/2004WR003041)

Knowles JD, Corne DW (2000) Approximating the nondominated front using the Pareto archived evolution strategy. Evol Comp 8(2): 149–172

Kuczera G, Mroczkowski M (1998) Assessment of hydrologic parameter uncertainty and the worth of multiresponse data. Water Resour Res 34(6): 1481–1489

Madsen H (2000) Automatic calibration of a conceptual rainfall-runoff model using multiple objectives. J Hydrol 235(4): 276–288

Nelder JA, Mead R (1965) A simplex method for function minimization. Comp J 7(4): 308–313

Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1992) Numerical Recipes in C. 2nd edition, Cambridge University Press, Cambridge, UK

Reed P, Minsker BS, Goldberg DE (2001) A multiobjective approach to cost effective long-term groundwater monitoring using an elitist nondominated sorting genetic algorithm with historical data. J Hydroinf 3: 71–89

Ritzel BJ, Eheart JW, Ranjithan S (1994) Using genetic algorithm to solve a multiobjective groundwater pollution containment problem. Water Resour Res 30(5): 1589–1603

Rozos E, Koutsoyiannis D (2006) A multicell karstic aquifer model with alternative flow equations. J Hydrol 325(1–4): 340–355

Rozos E, Efstratiadis A, Nalbantis I., Koutsoyiannis D (2004) Calibration of a semi-distributed model for conjunctive simulation of surface and groundwater flows. Hydrol Sci J 49(5): 819–842

Schaffer J (1984) Some experiments in machine learning using vector evaluated genetic algorithms. Ph.D. thesis, Vanderbilt University, Nashville

Solomatine DP (1998) Genetic and other global optimization algorithms – comparison and use in calibration problems. In: Hydroinformatics 1998 (Proc. Third Int. Conf. on Hydroinformatics, Copenhagen, Denmark), Intern. Water Assoc. Publishing, pp 1021–1028

Srinivas N, Deb K (1994) Multiobjective optimization using nondominated sorting in genetic algorithms. Evol Comp 2(3): 221–248

Wagener T, Boyle DP, Lees MJ, Wheater HS, Gupta HV, Sorooshian S (2001) A framework for development and application of hydrological models. Hydrol Ear Sys Sci 5(1): 13–26

Yapo PO, Gupta HV, Sorooshian S (1998) Multi-objective global optimization for hydrologic models. J Hydrol 204: 83–97

Zitzler EK, Thiele L (1999) Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. IEEE Trans Evol Comp 3(4): 257–271

Zitzler E, Laumanns M, Thiele L (2002) SPEA 2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In: Giannakoglou K, Tsahalis D, Periaux J, Papailiou K, Fogarty T (eds) Evolutionary Methods for Design, Optimization and Control. Barcelona, Spain, pp 19–26

Chapter 20
Evolutionary-based Meta-modelling: The Relevance of Using Approximate Models in Hydroinformatics

S.-T. Khu, D. Savic and Z. Kapelan

Abstract This chapter examines various applications of evolutionary computation (EC)-based meta-models to augment or replace the conventional use of numerical simulation and optimisation within the context of hydroinformatics. Evolutionary computation-based optimisation techniques are increasingly used in a wide range of water and environmental applications either as optimisation, analysis or design tools. However, despite the advances in computer power, it may still be impractical to rely exclusively on computationally expensive (time-consuming) simulation for many real-world complex problems. The meta-model investigated in this chapter can take various forms and, when coupled with a genetic algorithm, forms a fast and effective hybridisation. Three examples, including calibration of a rainfall–runoff model, modified Monte Carlo sampling of a kinematic wave model and the design and robust rehabilitation of water distribution models, are then used to illustrate the concept of EC-based meta-models. The proposed meta-model reduces the number of simulation runs required in the numerical model considerably, thus making the optimisation and statistical analysis of computationally intensive simulation models viable.

Keywords Evolutionary computation · meta-model · genetic algorithm · artificial neural network · calibration · uncertainty analysis

S.-T. Khu
Centre for Water Systems, School of Engineering, Computer Science and Mathematics, University of Exeter, UK EX4 4QF, e-mail: s.t.khu@exeter.ac.uk

D. Savic
Centre for Water Systems, School of Engineering, Computer Science and Mathematics, University of Exeter, UK EX4 4QF

Z. Kapelan
Centre for Water Systems, School of Engineering, Computer Science and Mathematics, University of Exeter, UK EX4 4QF

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

20.1 Introduction

Population-based evolutionary computational (EC) methods such as evolutionary algorithms (EA) (which include genetic algorithms, evolutionary strategies, evolutionary programming, etc.), shuffled complex algorithms, and simulated annealing are powerful search algorithms that can be used for optimisation. These algorithms are increasingly used as optimisation tools for a wide range of water and environmental simulation models. Examples of EC applications can be found in the design and operation of pipe network systems (Dandy et al., 1996; Savic and Walters, 1997); groundwater monitoring/containment (Ritzel and Eheart, 1994; Cieniawski et al., 1995); design and modelling of urban drainage systems (Rauch and Harremoes, 1999; Liong et al., 2001); calibration of river basin and urban water models (Duan et al., 1992; Liong et al., 1995; Savic and Walters, 1997) and many others. However, the main weakness in using EC-based search methods for design, calibration or operational control is that they require a large number of fitness evaluations, thereby rendering them unsuitable when computationally intensive simulation models are required.

It is not uncommon for large environmental simulation models to run for several hours or longer, such as that of running extended period simulation for water distribution networks, 3D flood inundation models or 3D coastal/tidal models and simulation models for integrated surface and sub-surface flows in a large catchment. With a typical EC run requiring thousands (if not tens of thousands) of model evaluations, it is currently infeasible or impracticable to perform full optimisation of these computationally intensive models.

To this end, there are basically three approaches to resolve the problem of using EC for computationally intensive optimisation. They are (i) using faster algorithms; (ii) utilising more computing power; and (iii) using approximate fitness functions. The first approach exploits the flexibility of EC to develop more efficient techniques requiring less function evaluations and, hence, less model evaluations. Typical methods of this approach are hybridisation of EA with some form of heuristics or local search (Deb and Beyer, 2001; Keedwell and Khu, 2003, Kapelan et al. 2003) and enhancement to EA operators (reproduction and selection) (Liong et al., 2001; Salami and Hendtlass, 2003). The second approach uses the inherent parallel computing capability of EA and allows simultaneous multiple model simulations on multiple processors (Kohlmorgen et al., 1999; Rivera, 2001). Hence the computational time required can be reduced proportionally to the number of co-processors used for parallel computing (subject to limitations of the slowest processor). The third method is the use of an approximate fitness function coupled or integrated with EA. To reduce the computational cost of model evaluations/simulations, surrogate evaluation tools such as fastest (but less accurate) models or simpler performance estimation through proxies may be used in place of the time-consuming simulations. These approximate or surrogate tools are commonly known as meta-models (Kleijnen, 1975).

This chapter reviews the different types of meta-modelling techniques currently available and different schemes of coupling meta-models with evolutionary computation in the wider engineering context. A generic framework for an EC-based

meta-model (ECMM) is formulated and their potential to replace a conventional scheme of numerical simulation and optimisation is investigated. The meta-model investigated in this chapter is an artificial neural network that when coupled with a genetic algorithm forms a fast and effective hybridisation. Three examples, including calibration of a rainfall–runoff model, modified Monte Carlo sampling of a kinematic wave model and the design of a water distribution network model under uncertainty, are used to illustrate a range of applications of EC-based meta-models in the field of hydroinformatics.

This chapter starts by reviewing different meta-models and EC-based meta-model frameworks, followed by illustrations of the concept of EC-based meta-modelling using three examples. Finally Sects. 20.5–20.7 will give concluding remarks and some discussions on current research issues related to EC-based meta-modelling and ways to enhance the acceptance of ECMM in the field of hydroinformatics.

20.2 Meta-models

Meta-models have been in existence for a long time (Kleijnen, 1975) and are widely used by the engineering design community to reduce the time required for full simulation (or testing of a prototype). For example, Gu (2001) reported that in Ford Motor Company, one crash simulation on a full passenger car may take up to 160 hours and that they had successfully implemented a meta-model to reduce the crash simulation time significantly.

Meta-models, otherwise known as surrogate or approximate models, are essentially a "model of the model" which may be used to approximate the simulation model. In general, a meta-model utilises some form of simplified (multivariate) function to approximate the underlying system using points that have already been evaluated and is considered to be a fast surrogate model compared to the exact evaluation model. The simplified function may be (i) linked to the processes to be modelled; (ii) some simplified solution(s) to the complex problem; (iii) a mathematical model of the input and output causal relationship; and/ or (iv) a surrogate generated due to inherency or proximity (Jin et al., 2001).

In general, the formulation of a meta-model involves the following three steps (Jin et al., 2001):

a) Experimental design – a means to generate data points for evaluation. Some examples are design of experiments (DoE), Latin hypercube importance sampling (LHS), and random selection
b) Model choice – the selection of a surrogate function as an approximator. Some examples are polynomial/or regression models, radial basis functions (RBF), kernel smoothing, kriging, rules and decision trees
c) Model fitting – tuning of the surrogate function to match the data points

Each of the steps may have many options and the choice of option at each step gives rise to different meta-models. For example, generating data using fractional factorial design and fitting the data onto a second order polynomial function using the method of least squares regression gives rise to the meta-model known as "response surface methodology" (RSM), while measured data fitted onto a network of artificial neurons using least squares with back-propagation gives rise to an "artificial neural network" (ANN) as a meta-model. The reader interested in meta-models which use approximate fitness functions can refer to Simpson et al. (2001) for a complete discussion.

The basic approach of constructing a meta-model may be generalised as follows:

- Select a surrogate function (meta-model) which can be used to approximate the "Simulator";
- run the "Simulator" for a small number of runs;
- construct the meta-model and adjust the variables within the model to fit the run results from the "Simulator";
- make necessary adjustments by performing additional "Simulator" runs; and
- update the meta-model either periodically or heuristically (based on certain rules).

Meta-modelling has been widely used in mechanical and aerospace design (Simpson et al., 2001), structural optimisation (Barthelemy and Haftka, 1993), manufacturing quality control (Li et al., 2003) and many others. Jin et al. (2001) compared four types of meta-models (RSM, RBF, kriging and multivariate adaptive regression splines) and presented the advantages and disadvantages of each technique using multi-criteria and multiple test problems. RBF was found to be the best performing algorithm, but data sampling has significant impact on the construction of the meta-model. If the data are not sampled correctly, the meta-model will not be a good representation of the simulation model it is trying to emulate, and therefore, the results of meta-models will not be accurate. This can be overcome, to a certain extent, by meta-models that are robust to noise.

Meta-models have also been successfully applied to model a variety of water and environmental problems. RSM has been applied to predict numerical geophysical models (Tatang et al., 1997), reconstruction and interpolation of effluent plumes in an estuary (Riddle et al., 2004) and the calibration of an urban drainage model (Liong et al., 1995). Kriging has been used to model the spatio-temporal pollutant deposit trend through the atmosphere (Haas, 1998), the spatial distribution of heavy metals in a river basin (Ouyang et al., 2002) and shallow water waves in an estuary (Gorman and Neilson, 1999). Artificial neural networks are probably the most widely used meta-models in all fields of hydrology and environmental engineering. ANNs have been used in hydrologic modelling (see Part II of this book for further examples), as well as any other applications such as modelling the input–output behaviour of wastewater treatment plants (Belanche et al., 1999); deforestation simulation (Mas et al., 2004); prediction of pollutant trends in urban areas (Lu et al., 2004); algae growth and transport modelling (Whitehead et al., 1997).

20.3 EC-Based Meta-models

As stated in the introduction, one possible way of overcoming the problem of time-consuming simulations in optimisation applications is to use meta-models in place of the simulation model. Many researchers, especially in engineering design, have examined strategies to integrate different meta-models with GAs. Emmerich et al. (2002) used kriging as the meta-model and found that kriging provided local error estimation which enabled assessment of the solution reliability. Giannakoglou et al. (2001) used a RBF network as a meta-model coupled with a GA to optimise an airfoil shape design. Poloni et al. (2000) used a hybridisation of a GA, ANN and local search method to optimise the design of a sailing yacht fin keel. The ANN acted as a surrogate model for the 3D Navier–Stokes simulation of the fin keel while cruising. Wilson et al. (2001) used RSM and kriging as a meta-model to explore the design space and capture the Pareto front during multiobjective design of an innovative grasper for minimal invasive surgical operations.

The concept of approximation in optimisation is not new (Barthelemy and Haftka, 1993) and recently, there has been a comprehensive survey of fitness approximation in evolutionary computation used by the engineering simulation and design community (Jin, 2005). According to Jin (2005), there are at least three categories of such EC-based meta-models and they are formulated through (i) problem approximation; (ii) function approximation; and (iii) evolutionary approximation. In problem approximation, the original problem statement is replaced by one which is approximately the same as the original problem, but which is easier to solve. An example would be to replace dynamic wave routing with kinematic wave routing or to replace 3D solvers with 1D-(quasi)2D solvers to enhance the computational efficiency. In function approximation, an alternate and explicit method of evaluating the objective function is constructed. Examples of function approximation are kriging (commonly used in geo-hydrology) and artificial neural networks (commonly used in surface hydrology). Evolutionary approximation is specific for evolutionary algorithms whereby the fitness of offspring is estimated based on the fitness of their parents without the need to run the simulation software again. Among these three fitness approximation methods function approximation is the most commonly encountered approach in meta-models and in some areas (such as aerospace engineering), function approximation is synonymous to meta-modelling. Problem approximation is usually very context specific hence difficult to generalise and requires the complete understanding of the different types of models available for the particular application. Application of evolutionary approximation to hydroinformatics is at its infancy stage, with most of the research work very much restricted to computer science applications at the moment. However, Yan and Minsker (2004) and Kapelan et al. (2003) have started to combine evolutionary approximation with fitness approximation to improve the performance of their meta-models on engineering optimisation problems.

The most direct way of integrating meta-models with a GA is to replace the "Simulator" with the meta-model completely during evaluation of the objective function in the GA. However, in order to construct the meta-model, a small number of

runs of the "Simulator" are required. This is the experimental design mentioned in Sect. 20.2 and can be performed either using the Taguchi method, Design of Experiments, the response surface methodology or even using a GA. Liong et al. (2001) detailed one such method using fractional factorial design with central composite design to provide an initial population for the GA.

Figure 20.1 gives an outline framework of how EC may be integrated with meta-models. The solid links refer to essential or basic linkages that are present in most EC-based meta-models and the dotted lines refer to optional linkages. In the next section, three examples of EC-based meta-models developed by the authors will be briefly discussed. Continual reference will be made to Fig. 20.1 to highlight the essential linkages.

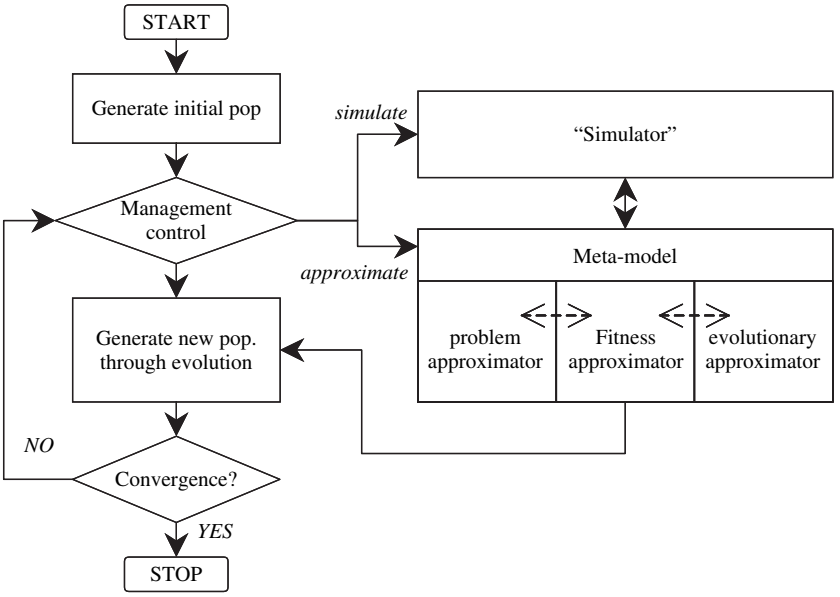


Fig. 20.1 Framework of an evolutionary computation-based meta-model (ECMM)

20.4 Application Examples

20.4.1 Example 1: Calibration of a Rainfall–Runoff Model

Khu et al. (2004) proposed an EC-based meta-model for the calibration of a rainfall–runoff model. This example demonstrates a scheme where a GA–ANN meta-model is used to calibrate the MIKE11/NAM model and the meta-model is constantly updated with the latest information, with the interval of updating governed by

heuristics. A GA is used to search for the optimal objective function in much the same way as any optimisation routine. The ANN chosen for this implementation is the radial basis function (RBF). The reason for choosing a RBF is that training and retraining (updating) the network is very efficient and simple. With sufficient training and updating, RBF can be used to map (and adapt) to the response surface of the objective function and used as a fast surrogate for the NAM model at regular intervals. As the GA search progresses, the response surface of the objective function tends to change and therefore, the RBF model needs to be self-adapting to the changing landscape. The updating strategy is based on probabilistic elitism where the best solutions have higher chances of being incorporated as updating points for the RBF. In this manner, the ANN is trained using the latest good solutions after each GA generation. A sensitivity analysis was conducted to determine the training sample, sampling frequency and updating size and frequency. It was found that a training sample of 300 individuals, a sampling frequency of every generation and an updating size of 5 individuals every generation performed best.

The above EC-based meta-model was used to calibrate the MIKE11/NAM rainfall–runoff model which simulates runoff from the Danish Tryggevaelde catchment. The objective was to determine suitable parameter values to simulate runoff from 1979 to 1993 from continuous daily rainfall during this period (Fig. 20.2). This period is further divided into three 5-year periods: calibration period, 1984–1988; validation period 1, 1979–1983; and, validation period 2, 1989–1993. A simple GA was used to determine the optimal parameter values while the RBF was used to map the objective function.

The EC-based meta-model was tested using two different objective functions with each objective function designed to examine a feature of the runoff:

(i) Average root mean square error for peak flow events, $\mathrm{RMSE}_{\mathrm{peak}}$, given by

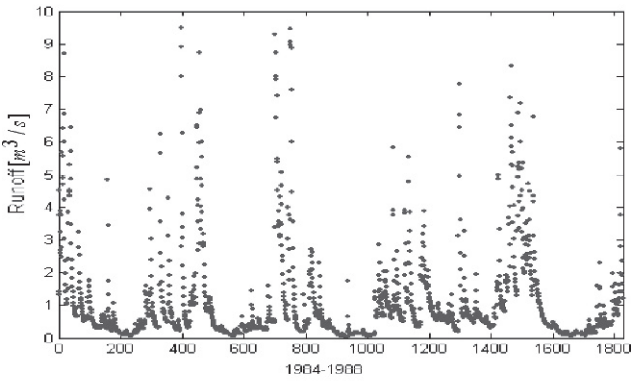
$$\mathrm{RMSE}_{\mathrm{peak}} = \frac{1}{M_{\mathrm{p}}} \sum_{j=1}^{M_{\mathrm{p}}} \left[\frac{1}{N} \sum_{i=1}^{n_j} [Q_{\mathrm{obs},i} - Q_{\mathrm{sim},i}(\theta)]^2 \right]^{1/2}$$

(ii) Root mean square error for low flow events, $\mathrm{RMSE}_{\mathrm{low}}$, given by

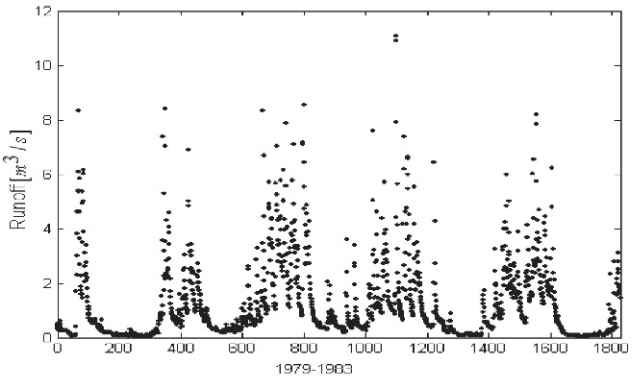
$$\mathrm{RSME}_{\mathrm{low}} = \frac{1}{M_{\mathrm{l}}} \sum_{j=1}^{M_{\mathrm{l}}} \left[\frac{1}{N} \sum_{i=1}^{n_j} [Q_{\mathrm{obs},i} - Q_{\mathrm{sim},i}(\theta)]^2 \right]^{1/2}$$

$Q_{\mathrm{obs},i}$ is the observed discharge at time i, $Q_{\mathrm{sim},i}$ is the simulated discharge, M_{p} is the number of peak flow events, M_{l} is the number of low flow events, n_j is the number of time steps in peak/low event no. j and θ is the set of model parameters to be calibrated. Peak flow events were defined as periods with flow above a threshold value of $4.0\,\mathrm{m}^3/\mathrm{s}$, and low flow events were defined as periods with flow below $0.5\,\mathrm{m}^3/\mathrm{s}$.

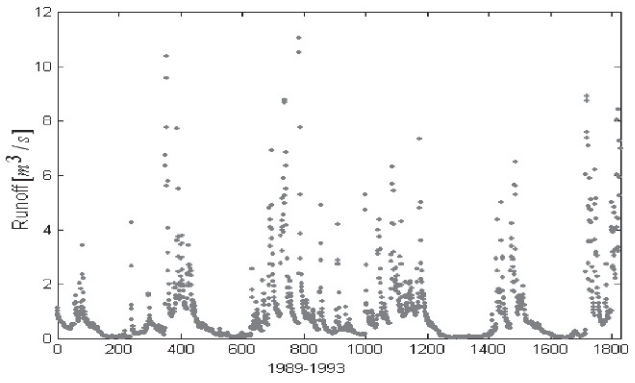
Figure 20.3 and Table 20.1 show the results of the proposed ECMM compared to those obtained from running the simple GA without the RBF. As can be seen



(a)



(b)



(c)

Fig. 20.2 Observed hydrographs for (**a**) calibration period; (**b**) validation period 1; and (**c**) validation period 2

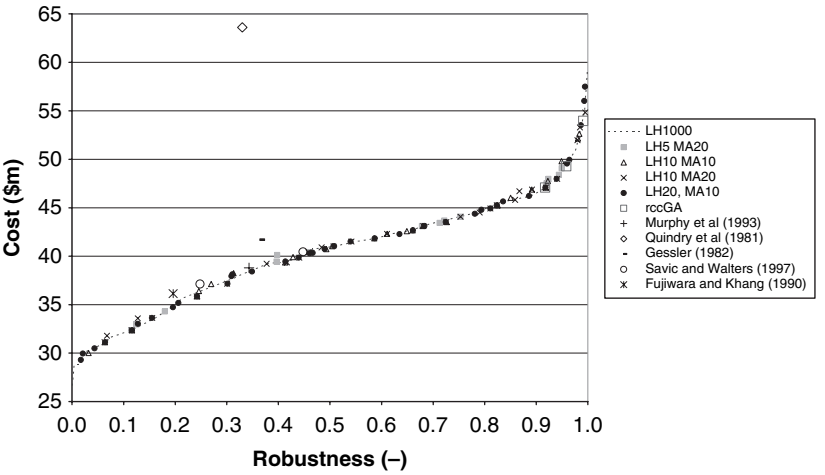


Fig. 20.3 LH1000 denotes the "full sampling" case solution, i.e. the solution obtained using the standard NSGAII (Deb et al. 2000) with 1000 Latin hypercube (LH) samples. Solutions labelled "LH *x* MA *y*" are the rNSGAII Pareto optimal solutions with *x* samples and a minimum chromosome age MA equal to *y* (MA is the rNSGAII algorithm parameter – see Kapelan et al. (2005) for more details). The referenced solutions are the corresponding, single-objective, deterministic problem solutions from the research literature (hence this representation with a single point)

from Table 20.1, the ECMM managed to outperform the GA during calibration runs for both peak flow events and low flow events on all accounts. When it comes to validation, the performance of the ECMM is comparable with those of the GA for both data sets. From Fig. 20.2, we noted that some of the peak flow events were of slightly higher magnitude compared with those during calibration. Thus good performance by the ECMM provides the modeller with reasonable confidence in the use of the ECMM for predicting previously unseen and slightly out-of-range data.

Table 20.1 Calibration and validation results for the average RMSE of peak and low flow events

		Calibration data (1984–1988)		Validation data set 1 (1979–1983)		Validation data set 2 (1989–1993)	
		GA	ECMM	GA	ECMM	GA	ECMM
RMSE (m^3/s)	Best RMSE	1.1750	1.1687	1.1724	1.1836	1.0485	1.1325
	Worst RMSE	1.2378	1.2178	1.2564	1.2516	1.3945	1.3672
	Mean	1.2016	1.1966	1.2303	1.2165	1.2005	1.2386
	STD	0.0184	0.0175	0.0292	0.0227	0.0964	0.0798
RMSE (m^3/s)	Best RMSE	0.1345	0.1323	0.2065	0.1752	0.0986	0.1043
	Worst RMSE	0.1782	0.1697	0.2491	0.2516	0.1470	0.1519
	Mean	0.1543	0.1451	0.2215	0.2196	0.1217	0.1164
	STD	0.0119	0.0110	0.0140	0.0210	0.0166	0.0159

20.4.2 Example 2: Modified GLUE Uncertainty Analysis

Another potential usage of ECMMs is the evaluation of risk and uncertainty. Currently, different sampling approaches have been devised to perform fast and effective sampling. Monte Carlo sampling (MCS) is commonly regarded as the most accurate approach but it requires thousands, if not tens of thousands, of model evaluations. Importance sampling, Metropolis algorithms, the Latin hypercube method, etc. are fast alternatives but they approximate the statistical properties of the MCS samples. Recently, Khu and Werner (2003) proposed the use of an EC-based meta-model (GA–ANN) to select regions of interest for sampling. Their method required only about 10% of the samples compared to the MCS method. A brief description of their approach is given below.

The method proposed by Khu and Werner (2003) was targeted specifically at reducing the number of simulation runs required under the generalised likelihood uncertainty estimation (GLUE) framework (Beven and Binley, 1992). GLUE utilises Monte Carlo sampling to estimate the uncertainty level and confidence limits of estimated parameters. In GLUE, the likelihood that a given parameter set is a good simulator of the system is a function of the model performance expressed in terms of the objective function chosen. All parameter sets performing below a pre-determined threshold are considered non-behavioural and removed from further analysis. This threshold is linked with the objective function and thus can only be evaluated through model simulation.

A practical problem with the GLUE procedure is that for models with a large number of parameters, the sample size from the respective parameter distribution must be very large to achieve a reliable estimate of model uncertainties. Moreover, depending on the level of acceptance of model behaviour, there may be a large number of redundant model runs. This posed two problems: (a) it is not feasible to perform GLUE if the simulation run is time consuming and (b) even when time is not a constraint, the efficiency of the sampling (measured according to the number of redundant runs) may be low for high threshold levels.

One possible remedy is of course to use meta-models in place of the actual simulation, but this means that the meta-model should be very accurate in order to estimate uncertainty in the model. Another possible remedy is to use an EC-based meta-model to estimate model performance of all sampling points from the MCS and select only those within the behavioural threshold for further evaluation. In Khu and Werner (2003), a GA was used to generate sufficiently good samples using the parameter space instead of random or stratified sampling. It also ensured that promising areas of the parameter space were sampled more densely. The typical GA search requires only a small fraction of those required by MCS. After that, an ANN was applied to map the response surface generated by the points selected by the GA. The MCS sampling points were projected onto the response surface and only those points that fell within the behavioural threshold were selected. Therefore, the number of model simulations can be reduced substantially.

The above methodology was applied to two rainfall–runoff models: the Nash–Cascade model and the Storm Water Management Model (SWMM). It was found

Table 20.2 Comparison of the number of model simulations required in applying GLUE and ECMM procedure

	Gamma function model		SWMM model	
	GLUE procedure	Procedure based on ECMM	GLUE procedure	Procedure based on ECMM
Initial runs required, N_{mc}	2500	250	25000	2500
Number of extra runs required, N_{sel}	–	571–723	–	2425–2503
Total number of runs required, N_{tot}	2500	821–973	25000	4925–5003
Number of runs generating behavioural points, N_{beh}	628 (average)	502–575	2586	2306–2354

that the ECMM required only 40% and 20% of the simulation runs for the Nash–Cascade model and SWMM, respectively, when compared to GLUE (Table 20.2). It was also shown that the number of redundant simulations was low compared to GLUE. Moreover, as seen in Table 20.2, the number of model runs that generated behavioural results (which are essential model runs) was very similar for both the ECMM and GLUE, indicating that the quality of the ECMM is comparable to that of GLUE. This study showed that the EC-based meta-model can also be effective in risk and uncertainty estimation.

20.4.3 Example 3: Optimal Design/Rehabilitation of Water Distribution Systems Under Uncertainty

In this example, the water distribution system (WDS) design/rehabilitation problem is solved as a multiobjective optimisation problem under uncertainty. The two objectives are (1) minimise the total design/rehabilitation cost and (2) maximise WDS robustness. The WDS robustness is defined as the probability of simultaneously satisfying minimum pressure head constraints at all nodes in the network. Decision variables are the alternative design/rehabilitation options for each pipe in the network. Two sources of uncertainty considered are future water consumption and pipe roughness coefficients. Uncertain variables are modelled using any probability density function (PDF) assigned in the problem formulation phase.

A conventional approach to solving the above optimisation problem would be to use some standard multiobjective optimisation method, e.g. the NSGAII (Deb et al., 2000) and a large number of Monte Carlo samples to accurately estimate fitness of each potential solution (i.e. WDS design). This approach, however, is extremely computationally demanding. To overcome this problem, a new

methodology was developed. The new methodology is based on a modified NSGAII algorithm named rNSGAII (robust NSGAII). In rNSGAII, each solution's fitness is estimated using the Latin hypercube sampling technique (McKay et al., 1979) with a very small number of samples (typically 5–50). Obviously, such a small number of samples allows only for a very rough fitness approximation to be made. However, at the same time, the rNSGAII algorithm enables breeding of the solutions (i.e. chromosomes) which are robust enough to survive over multiple generations. As a consequence, each chromosome's fitness can be calculated as the average of all past fitness values over that chromosome's age. Therefore, even if a small number of samples are used for each fitness evaluation, the fitness is effectively evaluated using a larger number of samples (100–1000 if the chromosome survived for example for 20 generations). Once the rNSGAII run is stopped, each point on the non-dominated Pareto front is re-evaluated using a large number of Monte Carlo samples (100,000 in the case study shown here).

Actually, the effective number of samples is even larger because the rNSGAII exploits the fact that the GA search process is of a stochastic nature with a population of solutions evaluated at each generation. It is a well-known fact that GA determines (near) optimal solution by combining highly fit building blocks of population chromosomes. As the search progresses, the population is likely to have more and more chromosomes containing highly fit building blocks. As a consequence, a relatively large number of indirect evaluations of these building blocks are likely to be found in the population even if a small number of samples are used to evaluate each chromosome's fitness.

The rNSGAII methodology was tested and verified on a New York tunnels reinforcement problem (Kapelan et al., 2005). The following four cases were analysed: (1) uncorrelated, normally distributed demands with 10% coefficient of variation; (2) uncorrelated, normally distributed demands with 30% coefficient of variation; (3) uncorrelated, normally distributed demands with 10% coefficient of variation and uncorrelated, uniformly distributed pipe roughness values and (4) correlated, normally distributed demands with 10% coefficient of variation (correlation coefficient between any two demands assumed equal to 0.50). The results obtained

Table 20.3 The rNSGAII algorithm computational times (minutes)[1,2]

Number of samples N_s	Cases 1 and 2	Case 3	Case 4
5	4.2	5.1	N/A
10	7.8	9.1	N/A
20	15	17	27
50	33	37	44
1000	340	400	500

[1] Optimisation runs performed on a PC with a 2.6 GHz AMD FX-55 processor, 1 GB RAM and the MS Windows XP Pro operating system; [2] Total number of deterministic (i.e. Epanet2 simulation) model evaluations in each rNSGAII run is equal to $2 \cdot N_{pop} \cdot N_{gen} \cdot N_s$ where N_{pop} is the GA population size (200 here) and N_{gen} is the number of GA generations before convergence (500 here).

indicate that the new methodology is capable of identifying accurately the robust Pareto optimal solutions (see Fig. 20.3 and Kapelan et al., 2005 for more details) whilst significantly reducing the overall computational effort (when compared to the "full sampling" case with 1000 samples, see Table 20.3).

20.5 Challenges in Current Implementation

All the above application examples utilise some form of surrogate model to approximate the simulation model. In the case of applications (1) and (2), an ANN was used, but it is well known that an ANN requires a minimum number of training data for a good response surface mapping. One would expect that there is an optimal number of training data points required at the beginning of the EC optimisation routine to train the ANN, thus leading to a minimum number of total simulations. There has yet to be some guidelines on this minimum number of training data points even though the work of Yan and Minsker (2004) has addressed some related issues. In the case of application (3), sampling was performed using the Latin hypercube method. Even though we used a very small number of samples and it worked well in the example, there needs to be guidelines on the optimal sample size. The sample size is one of the most critical parameters in the proposed rNSGAII method since it directly affects the computational efficiency of the method.

Despite the extensive works in evolutionary-based meta-models, little effort is spent on overcoming the problem of "changing landscape". During the process of optimisation, the region of the GA search will constantly change and it is reasonable to assume that the meta-model will have to be suitably modified to account for such changes. As the search progresses more information on the objective function will be obtained and a suitable mechanism should be implemented to utilise this additional information and update the meta-model. Quite recently, researchers have started working with EC-based meta-models to investigate the usage of evolution control as a mechanism to overcome the problem of "changing landscape". Evolution control (Jin and Branke, 2005) refers to the setting of some fixed rules that regulate the use of approximate models and compares them with the original simulation model. The first application example seemed to indicate that elitist sampling and updating holds promising results for dealing with this problem but more investigation is required.

20.6 Future Research Directions

Evolutionary computation-based meta-modelling as an optimisation technique for computationally intensive models is a research area that has not yet attracted sufficient attention in the water resources and environmental modelling community. The

authors would like to highlight several promising topics for future research, some of which were highlighted by Jin (2005):

(i) Develop meta-models for problems with nonlinear constraints. Many water systems applications, especially design problems, are heavily constrained. In such problems the objective function space will be highly undulating with pockets of infeasible solutions. Many existing meta-models are not able to map surfaces with discontinuities and voids. Hence, there is a need to develop meta-models capable of dealing with such problems.
(ii) Extend meta-modelling to be used within multiple objective optimisation and multi-criteria decision support approaches. Recently Yang et al. (2002) presented an interesting and novel EC-based meta-model known as adaptive approximation in single and multi-objective optimisation (AASO-AAMO) for the design of a 10-bar truss problem. The AASO-AAMO method differed from the framework so far in that adaptive updating of the meta-model is only carried out after the EC optimisation has converged. Their results indicate that the amount of computational savings could be as high as 99%. It would be extremely interesting to apply AASO-AAMO to a number of calibration and design problems in water and environmental systems.
(iii) Risk-based optimisation/dynamic stochastic optimisation. The third example in this chapter demonstrated a feasible way to use EC-based meta-modelling for uncertainty estimation. However, it remains a challenge to the research community to extend this concept to other risk-based or stochastic optimisation problems.

20.7 Conclusions

This chapter discusses the concept of meta-models and the integration of evolutionary algorithms and meta-models. It can be seen that there is a significant advantage in using meta-models for water and environmental system simulation, design and calibration. The chapter also explores various ways of linking optimisation routines with meta-models to form EC-based meta-models. The meta-model investigated in this chapter is the artificial neural network and, when coupled with a genetic algorithm, forms an effective and efficient optimisation routine. Three examples were then discussed.

This chapter also highlighted one major problem for evolutionary-based meta-modelling: that is how to ensure that the meta-model is constantly relevant as the search progresses. To overcome this problem, a number of strategic and periodic schemes of updating the meta-model may be used. The first example illustrated one such possible updating strategy that is unique to EC-based meta-models and showed that they are indeed feasible. Other methods of updating are also possible as indicated by examples 2 and 3. Furthermore, this chapter highlights several challenges encountered in the three examples and also lists several future research directions

for EC-based meta-modelling. In conclusion, we hope to have demonstrated that ECMM is very relevant to the field of hydroinformatics.

Acknowledgements The authors of this chapter wish to express their gratitude to the co-authors of various papers referenced in this chapter. They are: Dr. Yang Liu (former PhD student at the Centre for Water Systems), Dr. Henrik Madsen (who has assisted in interpreting the NAM model results) and Dr. Micha Werner (currently at Delft Hydraulics).

References

Barthelemy J-FM, Haftka RT (1993) Approximation concept for optimum structural design – a review. Struct. Optim. 5: 129–144.

Belanche LA, Valdes JJ, Comas J, Roda IR, Poch M (1999) Towards a model of input-output behaviour of wastewater treatment plants using soft computing techniques. Environ. Modelling & Software 14: 409–419.

Beven K, Binley A (1992) The future of distributed models: Model calibration and uncertainty prediction. Hydrological Processes 6: 279–298.

Cieniawski SE, Eheart JW, Ranjithan J (1995) Using genetic algorithms to solve groundwater monitoring problem. Water Resour. Res. 31(2): 399–409.

Dandy GC, Simpson AR, Murphy LJ (1996). An improved genetic algorithm for pipe network optimisation. Water Resour. Res. 32(2): 449–458.

Deb K, Beyer HG (2001) Self-adaptive genetic algorithm with simulation binary crossover. Evolutionary Comp. 9(2): 197–221.

Deb K, Pratap A, Agarwal S, Meyarivan T (2000) A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-objective Optimisation: NSGA-II, Report No. 200001, Indian Institute of Technology, Kanpur Genetic Algorithms Laboratory, p. 20.

Duan QY, Sorooshian S, Gupta V (1992) Effective and efficient global optimisation for conceptual rainfall-runoff models. Water Resour. Res. 28(4): 1015–1031.

Emmerich M, Giotis A, Ozdemir M, Back T, Giannakoglou K (2002) Metamodel-assisted evolution strategies. Lecture Notes in Computer Sci. 2439: 361–370.

Giannakoglou KC, Giotis AP, Karakasis M (2001) Low cost genetic optimisation based on inexact pre-evaluations and the sensitivity analysis of design parameters. Inverse Problems in Engrg. 9: 389–412.

Gorman RM, Neilson, CG (1999) Modelling shallow water wave generation and transformation in an intertidal estuary. Coastal Engrg. 36(3): 197–217.

Gu L (2001) A comprison of polynomial based regression models in vechicle safety analysis. In: Diaz, A. (ed.) ASME Design Engineering Technical Conferences – Design Automation Conference, Pittsburgh, PA, Paper No. DETC2001/DAC-21063. New York: ASME.

Haas T.C. (1998) Statistical assessment of spatio-temporal pollutant trends and meteorological transport models. Atm. Env. 32(11): 1865–1879.

Jin Y (2005) A comprehensive survey of fitness approximation in evolutionary computation. Soft computing, 9(12): 3–12.

Jin R, Chen W, Simpson TW (2001) Comparative studies of metamodelling techniques under multiple modelling criteria. Struct. Multidisc. Optim. 23: 1–13.

Jin Y, Branke J (2005) Evolutionary optimization in uncertain environments: A survey. IEEE Trans. Evol. Comput. 9(3): 303–317.

Kapelan Z, Savic DA, Walters GA (2003) A Hybrid inverse transient model for leakage detection and roughness calibration in pipe networks. J. Hydraulic Res. 41(5): 481–492.

Kapelan Z, Savic DA, Walters GA (2005) Multiobjective Design of Water Distribution Systems under Uncertainty. Water Resour. Res. 41(11), W11407.

Keedwell E, Khu ST (2003) More choices in water system design through hybrid optimisation. Proc. Int. Conf. on Comput. and Control for the Water Industry, pp. 257–264.
Kleijnen JPC (1975) A comment on Blanning's metamodel for sensitivity analysis: The regression metamodel in simulation. Interfaces 5(1): 21–23.
Khu ST, Werner MGF (2003) Reduction of Monte-Carlo simulation runs for uncertainty estimation in hydrological modelling. Hydrol. Earth Sys. Sci. 7(5): 680–692.
Khu ST, Savic D, Liu Y, Madsen H (2004) A fast evolutionary-based meta-modelling approach for the calibration of a rainfall-runoff model. 2nd International Integrated Environmental Modelling Society Conference, 14–17 June, Osnabruck, Germany.
Kohlmorgen U, Schmeck H, Haase K (1999) Experiences with fine-grained parallel genetic algorithms. Annals of Op. Res. 90: 2003–219.
Li TS, Su CT, Chiang TL (2003) Applying robust multi-response quality engineering for parameter selection using a novel neural-genetic algorithm. Computers in Industry 50: 113–122.
Lu W-Z, Wang W-J, Wang X-K, Yan S-H, Lam JC (2004) Potential assessment of a neural network model with PCA/RBF approach for forecasting pollutant trends in Mong Kok urban air. Hong Kong. Env. Res. 96(1): 79–87
Liong SY, ShreeRam J, Chan WT (1995) Catchment calibration using fractional factorial and central composite designs-based response surface. J. Hydr. Engrg. 121(8): 613–617.
Liong SY, Khu ST, Chan WT (2001) Derivation of Pareto front with genetic algorithm and neural networks. J. Hydrol. Engrg. 6(1): 52–61.
Mas JF, Puig H, Palacio JL, Sosa-Lopez A (2004) Modelling deforestation using GIS and artificial neural networks. Env. Modelling & Software 19: 461–471.
McKay MD, Conover WJ, Beckman RJ (1979) A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics, 211: 239–245.
Ouyang Y, Higman J, Thompson J, O'Toole T, Campbell D (2002) Characterization and spatial distribution of heavy metals in sediment from Cedar and Ortega rivers sub-basin. J. Contaminant Hydrol. 54(1–2): 19–35.
Poloni C, Giurgevich A Onesti L, Pediroda V (2000) Hybridization of a multi-objective genetic algorithm, a neural network and a classical optimizer for a complex design problem in fluid dynamics. Comput. Methods Appl. Mech. Engrg. 186: 403–420.
Rauch W, Harremoes P (1999) On the potential of genetic algorithms in urban drainage modeling. Urban Water 1: 79–89.
Riddle AM, Lewis RE, Sharpe AD (2004) Environmental quality information processor (EQUIP). Env. Modelling & Software 19(1): 57–62.
Ritzel BJ, Eheart JW (1994) Using genetic algorithms to solve a multiple objective groundwater pollution containment problem. Water Resour. Res. 30(5): 1589–1603.
Rivera W (2001) Scalable Parallel genetic algorithm. Artificial Intel. Rev. 16: 153–168.
Salami M, Hendtlass T (2003) A fast evaluation strategy for evolutionary algorithms. Appl. Soft. Comput. 2: 156–173.
Savic DA, Walters GA (1997) Genetic algorithms for the least-cost design of water distribution networks. ASCE J. Water Resour. Plng. Mgmt. 123(2): 67–77.
Simpson TW, Peplinski J, Koch PN, Allen JK (2001) Metamodels for computer-based engineering design: Survey and recommendations. Engrg. Computers 17(2): 129–150.
Tatang MA, Pan W, Prinn RG, McRae, GJ (1997) An efficient method for parametric uncertainty analysis of numerical geophysical models. J. Geophysical Res. 102: 21925–21932.
Wilson B, Cappelleri D, Simpson TW, Frecker M (2001) Efficient Pareto frontier exploration using surrogate approximation. Opt. Engrg. 2: 31–50.
Whitehead PG, Howard A, Arulmani C (1997) Modelling algal growth and transport in rivers: a comparison of time series analysis, dynamic mass balance and neural network techniques. Hydrobiologia 349(1–3): 39–46.
Yan S, Minsker B (2004) A dynamic meta-model approach to genetic algorithm solution of a risk-based groundwater remediation design model. ASCE Water and Environment conference. Salt Lake City, USA, July 2004.
Yang BS, Yeun Y-S, Ruy W-S (2002) Managing approximation models in multiobjective optimisation. Struct. Multidisc. Optim. 24: 141–156.

Chapter 21
Hydrologic Model Calibration Using Evolutionary Optimisation

A. Jain and S. Srinivasulu

Abstract Hydrologic models are key inputs to many water resource system projects. In the past, classical optimisation techniques have been employed for hydrologic model calibration and it is only since the 1990s that evolutionary optimisation is being employed for this purpose. This chapter presents a comprehensive literature review of hydrologic model calibration using classical and evolutionary optimisation and presents a case study of the application of a real-coded genetic algorithm to calibrate a conceptual rainfall–runoff (CRR) model. The rainfall and flow data for a 26-year period derived from the Kentucky River catchment of area in excess of 10,000 km^2 were used for this purpose. The performance of the calibrated CRR model was evaluated using five different standard statistical parameters. The results obtained in this study indicate that the real-coded genetic algorithm can be a very efficient tool for hydrologic model calibration and needs to be explored further by the researchers and hydrologists in catchments of varying hydrologic and climatic conditions.

Keywords Hydrologic modelling · genetic algorithms · model calibration · evolutionary computing · global optimisation

21.1 Introduction

Water is a scarce natural resource. Due to its limited availability, and an increase in water demand due to population and industrial growth, mankind's existing water resources need to be utilised in a sustainable manner. This calls for the efficient planning, design, operation and management of existing and proposed water resource systems, which require mathematical models of the various components of the hydrologic system. One of the important hydrologic processes that need to

A. Jain
Department of Civil Engineering, Indian Institute of Technology Kanpur, Kanpur – 208 016, UP, India, e-mail: e-mail:ashujain@iitk.ac.in

S. Srinivasulu

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

be modelled is the rainfall–runoff process. Many rainfall–runoff models of varying degrees of sophistication and complexity are described in the literature, and the choice of a model normally depends upon the data available and the practical application intended. Regardless of the kind of model, the first step in its use is the determination of its parameters. The model parameters are normally determined through model calibration or field measurements. In model calibration, optimisation techniques can be adopted to determine the optimal set of parameters using known rainfall and streamflow data.

Various sub-components of the rainfall–runoff process, such as infiltration, evapotranspiration, surface and groundwater flows, etc. are highly complex, non-linear and dynamic in nature, which makes the optimisation problem of hydrologic model calibration complex and non-linear. Duan et al. (1992) have reported that the problem of hydrologic model calibration can be ill-posed, highly non-linear, non-convex and multi-modal involving numerous optima. Such optimisation problems need heuristic solutions based on competitive evolution and probabilistic principles. Historically, researchers have employed classical optimisation methods for hydrologic model calibration and it is only since the 1990s that evolutionary techniques are being used for this purpose. The objective of this chapter is to review the literature available in the area of conceptual rainfall–runoff (CRR) model calibration and then present a case study that employs the soft computing technique of a real-coded genetic algorithm (GA) for hydrologic model calibration. The chapter begins with an extensive review of available literature on the calibration of hydrologic models. This is followed by a brief description of the CRR model used in the case study before presenting the results and making concluding remarks.

21.2 Review of Hydrologic Model Calibration

Historically, parameter estimation of hydrologic models has been performed using manual methods of trial and error. In manual methods of hydrologic model calibration, parameters are initially estimated, and time series plots are visually examined to find a match between the estimated and observed runoff hydrographs until all of the parameters are suitably determined. The US National Weather Service's River Forecast System based on soil moisture accounting (SMA-NWSRFS) was probably one of the earliest hydrologic models that employed a manual calibration technique. Another approach to model calibration has been the "automatic calibration method" that employs digital computers. Dawdy and O'Donnel (1965) used an automatic calibration method to calibrate mathematical models of catchment behaviour. Other important examples employing automatic methods for hydrologic model calibration include Jackson and Aron (1971), Clarke (1973) and Johnston and Pilgrim (1976). Johnston and Pilgrim (1976) presented a detailed search technique for parameter optimisation of catchment models. Eight different parameters of the Boughton rainfall–runoff model (Boughton, 1965) were estimated using simplex and Davidon optimisation methods. They concluded that in spite of achieving

rapid initial reductions in the objective function values, solutions approached several widely different apparent optima. James and Burges (1982) presented an extensive study on the selection, calibration and testing of hydrologic models using automatic methods. Sorooshian et al. (1983) reported the effect of data variability and the length of the record on model performance while evaluating the maximum likelihood parameter estimation technique for calibrating the SMA-NWSRFS model. They stressed the need for carefully selecting the objective function type and the quality of data employed for calibration and concluded that a properly chosen objective function can enhance the possibility of obtaining unique and conceptually realistic parameter estimates.

Sorooshian et al. (1993) reported that many earlier attempts of using automatic calibration methods to calibrate the Sacramento soil moisture accounting (SAC-SMA) model of the NWSRFS failed to obtain a unique set of parameters. Duan et al. (1993) proposed a robust, effective and efficient method for global optimisation called shuffled complex evolution (SCE-UA) for hydrologic model calibration. Sorooshian et al. (1993) applied a global optimisation method for automatic calibration of the SAC-SMA model. They investigated the consistency with which the two global optimisation methods, SCE-UA and the multi-start simplex (MSX) methods, were able to find the optimal parameter set during calibration of the SAC-SMA model. The SCE-UA method was found to perform consistently better than the MSX method. Duan et al. (1992) also applied the SCE-UA method to calibrate the SIXPAR rainfall–runoff model using synthetic rainfall–runoff data. Most of the earlier studies used lumped hydrologic models to demonstrate the suitability and efficiency of the various optimisation methods. Refsgaard (1997) emphasised the different requirements for calibration and validation of lumped and distributed models. Different steps in the calibration of the MIKE-SHE distributed hydrologic model were illustrated through a case study on the 440 km^2 Karup catchment in Denmark.

Most of the studies discussed above used classical optimisation problem solution methodologies. The rainfall–runoff process is a highly complex, dynamic and non-linear process and the mathematical models attempting to model it involve high degrees of complexity. As such, the method of estimating the optimal set of model parameters needs to be very robust because of the presence of one or more of the following difficulties in the search for the global optima: (a) there may be several major regions of attraction into which a search strategy may converge; (b) each of these major regions of attraction may contain many local optima that may be either close to or far away from the global solution; (c) the error function surface may not be smooth and could vary in an unpredictable manner; and (d) the model parameters may exhibit varying degrees of sensitivity and large non-linear inter-dependence. Any non-linear optimisation problem with the above characteristics must be solved with a global optimisation strategy that is based on the following concepts: (a) combination of deterministic and probabilistic approaches; (b) systematic evolution of the solution in the direction of global improvement and the concept of competitive evolution (Duan et al. 1993). Wang (1991) was probably one of the first researchers to employ this form of competitive evolution technique using a GA. He used a GA to calibrate the Xinanjiang rainfall–runoff model for the Bird Creek catchment. In

determining the optimal set of model parameters with a GA, 10 different GA runs were carried out. Each run started from different initial randomly selected solutions, and with 5,000 function evaluations, 8 out of the 10 runs were able to locate the global solution. Since the other two runs were also very close to the eight best solutions, it was argued that the GA method provided an efficient and robust means for the calibration of hydrologic models. Kuczera (1997) reported that the estimation of catchment model parameters is a difficult task due to ill-posed questions, the existence of multiple local optima and a heavy computational burden and emphasised the need to confine the search space to a subspace within which the global optimum is likely to be found. He compared four probabilistic search algorithms: SCE-UA, GA, multiple random start using a simplex search algorithm and multiple random start using a quasi-Newton local search method. In the Kuczera (1997) study, SCE-UA was found to be the most robust and the most efficient method for hydrologic model calibration. The GA performed better than SCE-UA initially but floundered near the optimum and could not be relied upon. This may have been due to problems associated with the binary-coded strings that are employed in a traditional GA. Savic et al. (1999) presented a genetic programming approach to hydrologic modelling for the Kirkton catchment in Scotland. The results obtained were compared with an optimally calibrated conceptual model and an artificial neural network (ANN) model. The GA and ANN data-driven approaches were found to be surprisingly similar in their consistency considering the relative size of the models and the number of variables involved. Recently, Ndiritu and Daniel (2001) proposed an improved genetic algorithm (IGA) for rainfall–runoff model calibration and function optimisation. The standard binary-coded GA was improved using three different strategies to deal with the occurrence of multiple regions of attraction. The performance of the proposed IGA was compared with the SCE-UA method using three different types of optimisation problems. The IGA was reported to be about 2 times less efficient, 3 times more efficient and 34 times less efficient than the SCE-UA method for the SIXPAR hydrologic model calibration, the Hartman function and the Griewank function optimisation, respectively. Their study highlighted the fact that in spite of attempts at modifying existing GA procedures, the efficiency of the optimisation of hydrologic model calibration could be improved only marginally. Apart from GAs, some researchers have employed other types of randomised search or global optimisation algorithms, e.g. clustering-based algorithms (Solomatine 1998) and simulated annealing (Skaggs et al. 2001).

There appears to be a shift in the emphasis of researchers from classical methods towards the evolutionary approaches for the calibration of hydrologic models. However, most of the studies reported earlier employed the binary-coded GA in which binary strings are used to represent possible solutions. The GA that employs real-valued strings is called a real-coded GA. The real-coded GA uses the decision variables directly to compute the fitness values while the binary-coded GA uses a mapping of binary digits on to real space to calculate the fitness values. This chapter presents a case study which employs a real-coded GA for the calibration of a CRR model. A brief description of the CRR model used in the case study is provided first.

21.3 Conceptual Rainfall–Runoff Model

In a CRR model, a simplified conceptual representation of the underlying physics is adopted instead of using the equations of mass, energy and momentum to describe the process of water movement. The development of a CRR model is a two-step process, which is responsible for modelling the non-linear, dynamic and complex nature of the rainfall–runoff process. The first step is the calculation of infiltration and other losses and the estimation of effective rainfall, and the second step is the transformation of the effective rainfall into runoff through an operator which simulates the behaviour of the catchment being considered. A schematic of the hydrologic system adopted in this study is shown in Fig. 21.1.

Total rainfall, represented by *P*, is considered as an input to the hydrologic system. A portion of the total rainfall infiltrates into the soil and appears at the outlet as a base flow (QG) after passing through subsurface storage. The remaining portion of the total rainfall, effective rainfall (ER), runs through the surface storage and appears at the outlet as surface flow (QS). The CRR model consists of the following components: (a) a base flow component, (b) an infiltration component, (c) a soil moisture accounting (SMA) component and (d) a surface flow component. The base flow component was modelled using the concept of flow recession in a catchment; infiltration was modelled using the Green–Ampt method; and the SMA component was modelled using a simple mass balance to update the soil moisture continuously.

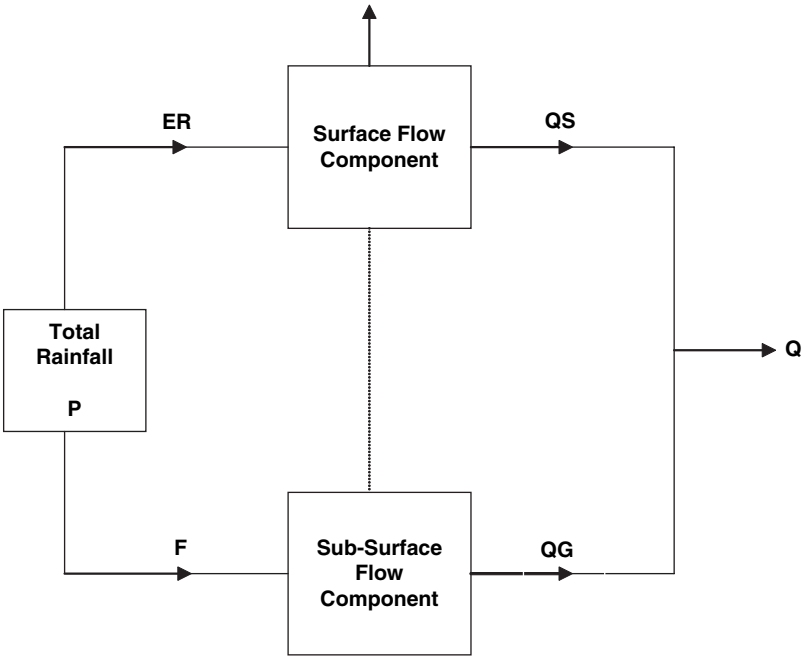


Fig. 21.1 Schematic of the simplified hydrologic system

The details of these three components are not included here for brevity and can be found in Jain and Srinivasulu (2004). The surface flow component is described in detail in the following section.

21.3.1 Surface Flow Model Component

The total observed flow at the outlet of a catchment can be thought of as the sum of surface flow and subsurface flow components. This can mathematically be represented as follows:

$$\mathrm{QO}_t = \mathrm{QS}_t + \mathrm{QG}_t \tag{21.1}$$

where QO_t is the observed flow at time t, QS_t is the surface flow component at time t and QG_t is the base flow component at time t.

The surface flow (QS_t) from the catchment can be derived by simulating the catchment to be a linear reservoir. The inflow (ER_t) to the surface flow system can be routed through the linear reservoir to compute QS_t using a modified version of Singh (1988) as follows:

$$\mathrm{QS}_t = C_1 \left(\frac{\mathrm{ER}^*_{t-1} + \mathrm{ER}_t}{2} \right) + C_2 \, \mathrm{QS}^*_{t-1} \tag{21.2}$$

$$K_1 = \frac{Q_{t-2}}{\Delta Q_{t-2}} \tag{21.3}$$

$$\Delta Q_{t-2} = \frac{Q_{t-1} - Q_{t-3}}{2} \tag{21.4}$$

$$C_1 = \frac{2\Delta t}{2K_1 + \Delta t} \tag{21.5}$$

$$C_2 = \frac{2K_1 - \Delta t}{2K_1 + \Delta t} \tag{21.6}$$

where C_1 and C_2 are the linear reservoir routing coefficients, ER^*_{t-1} is the updated amount of inflow coming into the surface storage at time $(t-1)$, ER_t is the effective rainfall coming into the surface storage at time t, QS^*_{t-1} is the updated amount of surface flow at time $t-1$, K_1 is an attenuation constant that can be determined from historical flow data using Clark's method (Singh 1988) and Δt is the time interval of the model. QS^*_{t-1} can be computed from past observed stream flow and base flow, using the following equation:

$$\mathrm{QS}^*_{t-1} = \mathrm{QO}_{t-1} - \mathrm{QG}_{t-1} \tag{21.7}$$

where QO_{t-1} is the observed streamflow at time (t–1) in m^3/s and QG_{t-1} is the computed base flow at time (t–1) in m^3/s. Further, the updated value of ER^*_{t-1} to be used in (21.2) can be estimated by using the same equation in inverse direction written at a previous time step as follows:

$$\mathrm{ER}^*_{t-1} = 2\frac{(\mathrm{QS}^*_{t-1} - C_2\,\mathrm{QS}^*_{t-2})}{C_1} - \mathrm{ER}^*_{t-2} \tag{21.8}$$

The surface flow on the rising limb was modelled using (21.2) through (21.8). The total flow on the rising limb was calculated in accordance with (21.1). The falling limb of the flow hydrograph can be modelled using an adaptive decay model, which can mathematically be expressed as follows:

$$Q_t = \mathrm{Kf}_t \mathrm{QO}_{t-1} \tag{21.9}$$

$$\mathrm{Kf}_t = \frac{\mathrm{QO}_{t-1}}{\mathrm{QO}_{t-2}} \tag{21.10}$$

where Q_t is the modelled streamflow at time t on the falling limb (m^3/s), Kf_t is the decay coefficient on the falling limb at time step t and QO_{t-2} is the observed stream flow at time $t-2$ on the falling limb (m^3/s).

The CRR model employed in this study consists of a total of nine parameters: one parameter in the base flow model; four parameters in the infiltration model (K, ψ, η and S); two parameters in the SMA model; and two parameters in the surface flow model (K_1 and Kf). The parameters KG_t, K_1 and Kf_t can be determined using past flow data adaptively and need not be calibrated. Thus, only six parameters (four for infiltration and two for SMA) need to be calibrated.

21.4 Model Application and Results

The CRR model presented above was applied to the Kentucky River Basin, USA. The daily rainfall and river flow data derived from the Kentucky River Basin (see Fig. 21.2) were employed to calibrate and validate the CRR model. The drainage area of the Kentucky River at Lock and Dam 10 (LD10) near Winchester, KY, is approximately 10,240 km^2. The data used include average daily streamflow (m^3/s) for the Kentucky River at LD10 and daily average rainfall (mm) from five rain gauges scattered throughout the Kentucky River Basin. The total length of the available rainfall–runoff data was 26 years. The data were divided into two sets: a calibration data set of 13 years (1960–1972) and a validation data set of 13 years (1977–1989). The statistical properties (mean and standard deviation) of the calibration and validation sets were comparable.

The two parameters of the SMA model were estimated using regression analysis and their details can be found in Jain and Srinivasulu (2004). The four Green–Ampt parameters (K, ψ, η and S) were determined using a real-coded GA in this study. The real-coded GA offers certain advantages over the traditional binary-coded GA: (a) it allows the search for an optimal solution in a continuous real-valued search space; (b) high precision can be easily achieved in the real-coded GA without having to increase the size of the population; (c) it avoids positional bias in carrying out the single-point crossover operator; and (d) it is more flexible and robust in searching

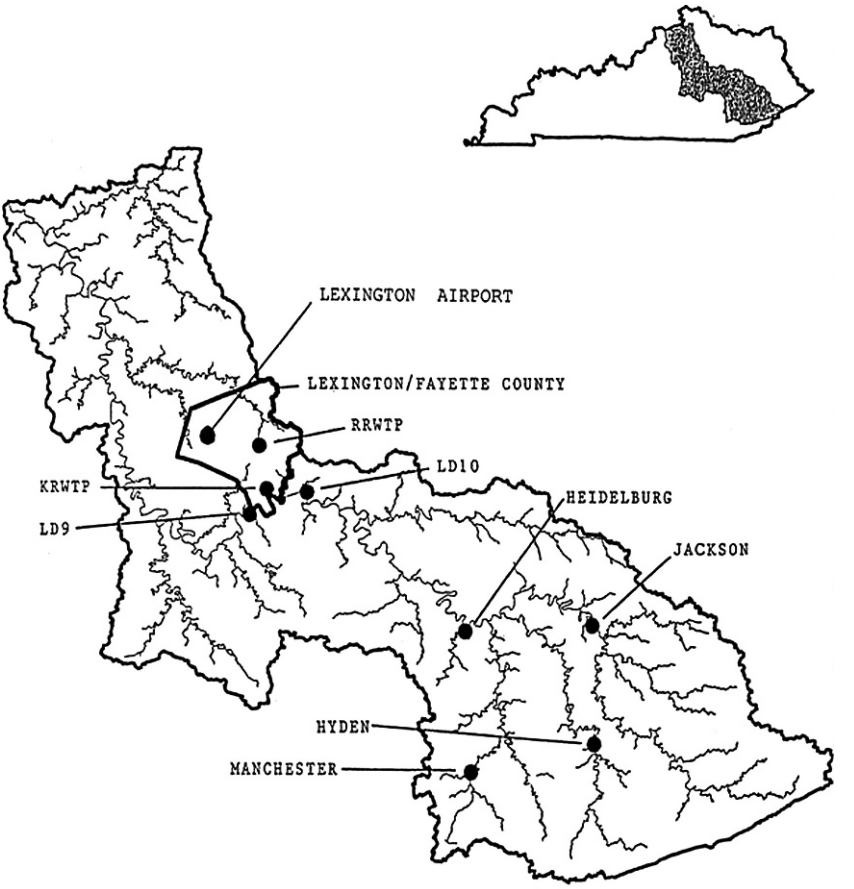


Fig. 21.2 Kentucky River Basin

for a global solution around the Hamming Cliff and creating feasible children solutions. More details of real-coded GAs can be found in Deb and Agarwal (1995) and Deb (2000).

In determining the Green–Ampt parameters using a real-coded GA, an objective function in the form of some error function that needs to be minimised can be formulated. In the present study, the objective function was the mean squared error (MSE) as follows:

$$E = \frac{1}{N}\sum_{t=1}^{N}(Q(t) - \mathrm{QO}(t))^2 \tag{21.11}$$

where E is the error function to be minimised in the optimisation formulation, $Q(t)$ is the estimated flow at time t, $\mathrm{QO}(t)$ is the observed flow at time t and N is the total number of data points in the observed flow series during calibration. The real-coded

GA parameters $\eta_c = 2$, $\eta_m = 20$; crossover probability $P_c = 0.9$; and mutation probability $P_m = 0.01$ and a population size of 40 were used. While carrying out the search for global solutions, the GAs were continued until fitness values converged to the specified tolerance level or the maximum number of function evaluations was reached. In an attempt to achieve the global optimal solution, 20 different runs starting with different initial populations were carried out and all of them resulted in similar solutions indicating that a near-global solution was reached. The best solution out of the 20 solutions is reported here. The parameters of the Green–Ampt equations corresponding to the best solution are $K = 0.1999$ mm/hour, $\psi = 201.267$ mm, $\eta = 0.09686$ and $S = 305.42$ mm.

Once the CRR model has been calibrated, it can be employed to calculate daily streamflow in the Kentucky River during both calibration and validation periods. Then, performance evaluation indices can be computed for the two 13-year calibration and validation periods. Five different performance evaluation parameters were used in this study for this purpose. These are average absolute relative error (AARE), correlation coefficient (R), Nash–Sutcliffe coefficient of efficiency (E), normalised root mean squared error (NRMSE) and normalised mean bias error (NMBE). The equations to calculate these statistical measures and their overview can be found in Jain and Srinivasulu (2004). The computed values of the performance evaluation measures are presented in Table 21.1.

It can be noted from Table 21.1 that values of R in excess of 0.93 and E in excess of 0.83 obtained for both calibration and validation data sets indicate a very good model performance. The NMBE values around +8.9% indicate that the model tends to over-predict the flow slightly, which may be due to the structure of the surface flow component that is linear in nature. AARE values around 20% for both calibration and validation data sets also indicate good model performance. These results confirm that the real-coded GA is able to provide a very good calibration of the hydrologic process in the Kentucky River Basin. The performance of the real-coded GA-determined parameters is represented graphically in the form of a scatter plot of the observed and estimated flows during validation in Fig. 21.3. The scatter plot indicates a uniform departure from the ideal line at all magnitudes of flow barring a few outliers.

Table 21.1 Performance statistics from the CRR model

Statistic	Calibration	Validation
AARE	23.57	24.68
R	0.9363	0.9332
E	0.8436	0.8344
NRMSE	0.639	0.649
NMBE	8.951	9.462

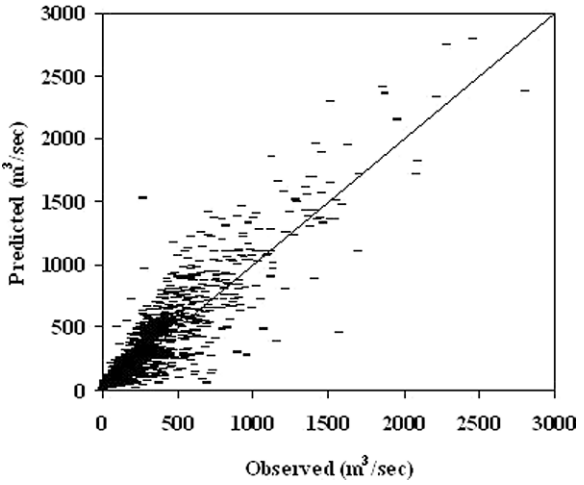


Fig. 21.3 Scatter plot of observed and computed flows for the validation period

21.5 Summary and Conclusions

The calibration of hydrologic models has been a difficult task for researchers and practicing hydrologists. The efforts at determining the parameters of a CRR model have ranged from manual methods of trial and error in earlier days to the very sophisticated methods of soft computing today. This chapter presents a brief review of the relevant literature and a case study of the use of a real-coded GA for hydrologic model calibration. A CRR model consisting of a total of nine parameters was employed, of which the four Green–Ampt parameters were calibrated using a real-coded GA. The performance of the calibrated CRR model was evaluated on the calculated flows for the two 13-year calibration and validation data sets using five different statistical measures.

The performance of the hydrologic model calibrated using a real-coded GA was found to be excellent in terms of R and E and very good in terms of AARE, NRMSE and NMBE statistics. The results obtained in this study indicate that the real-coded GA offers a suitable alternative to the problem of hydrologic model calibration. The real-coded GA needs to be further exploited by researchers working in hydrology and closely related areas. The CRR model employed in the present study was a lumped model. It would be interesting to investigate the performance of a real-coded GA in the calibration of a distributed hydrologic model. It would also be interesting to compare the performance of a real-coded GA with the traditional binary-coded GA and its other variations in hydrologic model calibration. Also, the use of different error functions in the GA optimisation may lead to improved performance in hydrologic model calibration. A sensitivity analysis that evaluates the effect of changes in each calibrated parameter on the model performance also needs to be carried out. It is hoped that future research efforts will focus in some of these

directions and thereby help to improve the performance of both existing and planned water resource systems.

References

Boughton WC (1965) A mathematical model for relating runoff to rainfall with daily data. Civil Engg Trans Inst Engrs Australia CE8(1):83–97

Clarke RT (1973) A review of some mathematical models used in hydrology with observations on their calibration and use. J Hydrol 19(1):1–20

Dawdy DR, O'Donnel T (1965) Mathematical models of catchment behaviour. J Hydrl Divn ASCE 91(HY4):123–137

Deb K (2000) An efficient constraint handling method for genetic algorithms. Comput Methods Appl Mech Engg 186:311–338

Deb K, Agarwal RB (1995) Simulated binary crossover for continuous search space. Complex Sys 9:115–148

Duan Q, Gupta VK, Sorooshian S (1993) Shuffled complex evolution for efficient and efficient global optimization. J Opt: Theory & Appl 76(3):501–521

Duan Q, Sorooshian S, Gupta VK (1992) Effective and efficient global optimization for conceptual rainfall–runoff models. Wat Resour Res 28(4):1015–1031

Jackson DR, Aron G (1971) Parameter estimation in hydrology. Wat Resour Bull 7(3):457–472

Jain, A, Srinivasulu, S (2004) Development of effective and efficient rainfall–runoff models using integration of deterministic, real-coded GA, and ANN techniques. Wat Resour Res 40(4) W04302, doi:10.1029/2003WR002355

James LD, Burges SJ (1982) Selection, calibration, and testing of hydrologic models. In: Haan CT (ed) Hydrologic modelling of small watersheds. Monograph 5 ASAE Michigan, USA, pp 437–472

Johnston PR, Pilgrim DH (1976) Parameter optimization of watershed models. Wat Resour Res 12(3):477–486

Kuczera G (1997) Efficient subspace probabilistic parameter optimization for catchment models. Wat Resour Res 33(1):177–185

Ndiritu JG, Daniel TM (2001) An improved genetic algorithm for rainfall–runoff model calibration and function optimization, Math & Comp Modelling 33:695–706

Refsgaard JC (1997) Parameterisation, calibration, and validation of distributed hydrologic models. J Hydrol 198:69–97

Savic DA, Walters GA, Davidson JW (1999) A genetic programming approach to rainfall–runoff modelling. Wat Resour Mgmt 13:219–231

Singh VP (1988) Hydrologic systems: Vol 1. Prentice Hall, Englewood Cliffs, New Jersey, USA

Skaggs RL, Mays LW, Vail LW (2001) Application of enhanced annealing to ground water remediation design. J Am Wat Resour Assoc 37(4):867–875

Solomatine DP (1998) Genetic and other global optimisation algorithms – comparison and use in calibration problems. Proc. 3rd International Conference on Hydroinformatics. Copenhagen, Denmark, pp 1021–1028

Sorooshian S, Duan Q, Gupta VK (1993) Calibration of rainfall–runoff models: Application of global optimization to the Sacramento Soil Moisture Accounting Model. Wat Resour Res 29(4):1185–1194

Sorooshian S, Gupta VK, Fulton JL (1983) Evaluation of maximum likelihood parameter estimation technique for conceptual rainfall–runoff models: Influence of calibration data variability and length on model credibility. Wat Resour Res 19(1):251–259

Wang QJ (1991) The genetic algorithm and its application to calibrating conceptual rainfall–runoff models. Wat Resour Res 27(9):2467–2471

Chapter 22
Randomised Search Optimisation Algorithms and Their Application in the Rehabilitation of Urban Drainage Systems

D.P. Solomatine and Z. Vojinovic

Abstract Urban drainage systems constitute a very significant portion of all assets in urban areas. Their structural integrity and functional efficiency represent key parameters for the safe transfer and disposal of surface runoff and domestic/trade discharge. Hydroinformatics tools can help in dealing with the optimal rehabilitation of such systems. An approach that links the hydrodynamic model of a drainage system with the multi-criteria global evolutionary optimisation engine that takes into account the performance indicators relevant for rehabilitation decisions is being developed. This paper presents the tools and the optimisation algorithms used, and a simple case study demonstrating the effectiveness of the approach.

Keywords Urban drainage asset management · system rehabilitation · global optimisation

22.1 Introduction

22.1.1 New Possibilities in Optimisation

For many years, in the 1970s and 1980s, water engineers were mainly using optimisation techniques in reservoir optimisation and, to some extent, in real-time control. Other problems where optimisation could have been used were posed but in practice only small-scale problems could be solved with the techniques and tools that were at the disposal of water engineers and managers.

D.P. Solomatine
UNESCO-IHE Institute for Water Education, 2601 DA Delft, The Netherlands

Z. Vojinovic
UNESCO-IHE Institute for Water Education, 2601 DA Delft, The Netherlands

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

The information technology revolution has changed the situation dramatically. Modern methods of optimisation and the available inexpensive computing power make it possible to solve problems that were not even posed 10–15 years ago. In a water context many such methods are being developed and used by the hydroinformatics community. The problems where an objective function cannot be expressed analytically (and indeed there are many of them) are nowadays solved by the use of direct optimisation methods. Such methods, instead of using analytical expressions for gradients are based on direct calculations of the function and in some sense the search for an optimum generally follows the direction of the gradient vector. These methods require a large number of objective function evaluations, where each evaluation involves a run of a mathematical model, and their application in engineering practice became possible only after the arrival of inexpensive powerful computers. A group of methods generally referred to as *randomised search* methods is the most popular group of direct optimisation methods.

Note also that many problems are typically multi-extremum and so a straightforward application of gradient-based methods (searching for a local optimum) is not adequate. Methods oriented at finding the global optimum (contrasted to finding a local one) are called *global optimisation methods*. Randomised search methods belong to this group.

A family of randomised search methods called evolutionary and genetic algorithms (abbreviated as GA, EA or EGA) has become quite popular during the last 15 years (Michalewicz, 1999). Their success is fully deserved and can be explained by their methodological appeal, relative simplicity, robustness and the existence of a well-organised community (see the chapter by Savic in this volume). The recent developments in hybrid EAs, in particular memetic algorithms, have led to considerable improvements in the effectiveness of this class of algorithms.

There are many other algorithms following the idea of randomised search: simulated annealing, particle swarm optimisation, ant colony optimisation, adaptive cluster covering and others. There are many examples of their successful use in many areas, and the choice of a particular algorithm is often based on the experience and preferences of a researcher or an engineer and not on a detailed investigation of effectiveness (accuracy) and efficiency (speed) of a particular algorithm. Due to the power of modern computers such an approach is often justified since the additional research costs much more than running a computer for several extra hours. (For example, the relative inefficiency of standard GAs is often ignored by practitioners since "GAs are popular and work relatively well anyway", and, indeed, this is true). This of course does not mean that there is no need to build better algorithms since for complex problems running time can be very significant (days and weeks), and, obviously, a more efficient and reliable algorithm should be typically preferred to a slower one (even if the latter is well known and being used by many people).

In this chapter we have used the adaptive cluster covering algorithm (ACCO) (Solomatine, 1995, 1999), which has proved to be an effective and efficient tool for solving many global optimisation problems.

22.1.2 Two Important Optimisation Problems in Water Management

Two important problems that are considered in this chapter can be mentioned in the context of optimisation: automatic calibration of complex models and optimisation of pipe networks.

The calibration problem is a typical problem where the objective function cannot be expressed analytically (since its value is always a result of running a model), and which can only be solved by using direct optimisation (Wang, 1991; Babovic et al., 1994; Cieniawski et al., 1995; Solomatine, 1995; Savic and Walters, 1997; Franchini & Galeati, 1997).

The problem of water distribution pipe network optimisation was posed long ago and a number of approaches based on gradient-based optimisation were suggested to solve it (Alperovits & Shamir, 1977; Yates et al., 1984; Kessler & Shamir, 1989). However, the complexity of the problem does not allow for neat analytical formulation of the objective function and constraints so that its accurate solution became possible only in the 1990s when computers became powerful enough to run randomised search algorithms (Simpson et al., 1994; Dandy et al., 1996; Savic & Walters, 1997; Abebe & Solomatine, 1998).

During the last years more and more problems are posed as *multi-objective optimisation* problems, and the two aforementioned problems are not exceptions. In water management there are many conflicting interests and objectives, and this leads to natural formulations with several criteria. There are specialised methods for solving such problems (e.g. Deb et al., 2002), but often there is a tendency to update them to make them more effective in solving water-related engineering problems (see, e.g. Tang et al., 2005). An increasing number of papers is appearing lately on multi-objective optimisation of water distribution networks (Prasad and Park, 2004; Kapelan et al., 2005). In this chapter we use both single- and multi-objective approaches.

22.1.3 Motivation

The main motivation for writing this chapter was the growing understanding of the second author that the problems of urban drainage need the application of adequate optimisation methods and tools. The basis for this chapter was the experience of the first author with optimisation methods in solving water-related problems including pipe network optimisation, and the experience of the second author in solving practical problems of urban water management using hydrodynamic models and his initial experience in posing and solving problems of optimal rehabilitation of drainage networks using traditional branch-and-bound methods.

The problem of the multi-objective approach to optimal rehabilitation of drainage networks has not yet received adequate attention in the scientific and engineering community. Such an approach was indeed discussed by Rauch and Harremoes (1999), but it was not actually applied to a drainage network. Diogo et al. (2000) consider the

(single-objective) optimal layout and design of urban drainage systems for sewage and storm water; they used dynamic programming, alongside simulated annealing and genetic algorithms to solve the discrete combinatorial optimisation problem.

This chapter makes a contribution to solving the optimal rehabilitation problem. Single- and multi-objective approaches to its solution based on the use of randomised search optimisation methods employed earlier in other hydroinformatics-related studies are demonstrated.

22.2 Wastewater Pipe Networks and Their Optimisation

Wastewater system rehabilitation is usually implemented based on several structural and hydraulic indicators that centre around the concept of levels of service. As presented in Vojinovic and Solomatine (2005a; 2006), the methodology to do that should consider the following aspects: full utilisation of the existing system's capacity prior to undertaking improvement works, reduction of pollution affecting local receiving waters as well as main watercourses and groundwater, reduction of infiltration/inflow in the network and its adverse impact on wastewater treatment, prevention of local system surcharges affecting properties and the environment, prevention of structural collapses and damage to other subsurface infrastructures and the prevention of the pollution of water supply systems, due to infiltration of wastewater and minimisation of expenditure.

All the aforementioned aspects are almost equally important for sustainable urban water management; therefore, finding the right balance between the extent of remedial works requirements (i.e. the target level of the system's performance) and capital expenditure is one of the greater challenges for those concerned with urban water systems.

In the approach discussed in this chapter, the above problem is posed as a fully fledged multi-objective optimisation problem, taking into account most of the aforementioned factors. It deals with determining the optimal pipe diameters for a network with a predetermined layout that needs remediation given the specific constraints. This problem belongs to the group of multi-extremum (global) optimisation methods mentioned in the Introduction. The MOUSE system (developed by DHI Water & Environment) is used as the wastewater pipe network modelling tool.

The objective of selecting an optimal set of remedial measures (i.e. remedial works optimisation) of a pipe network system is to identify the vector of some values $x = \{x_1, \ldots, x_n\}$, which are not known a priori (x could be a set of options characterising the possible remedial works, for example pipes upgrades). This is achieved by feeding the model input data, and subsequently calculating the total cost of the works required. The remedial works optimisation problem can be posed in two ways: (1) as a single-criterion optimisation problem, in the case of a single output variable (e.g. the total cost of remedial works) and (2) as a multi-criteria optimisation problem – when there are several output variables (e.g. the total cost of remedial works and the total system surcharge/overflow volume, etc.).

22.2.1 Multi-Objective Setting of an Optimisation Problem

A problem characterised by several variables $(x_1,\ldots,x_n)$ is considered. In most real-life optimisation (we will consider minimisation) problems a solution $x^* = \{x_1{}^*,\ldots,x_n{}^*\}$ is sought such that several objective functions (criteria) $f_1, f_2, \ldots, f_m$ are minimised. Typically these objectives are conflicting and there are no such solutions, so a solution is sought that would at least bring low values to the objectives. Finding a compromise between different objectives is the central problem in multi-objective optimisation.

In the context of drainage system rehabilitation, vector $(x_1,\ldots,x_n)$ could represent various system improvements (upgrades of particular pipes, new storage tanks, etc.), and objectives could be costs of remedial works, flood damage, risk of failure, etc. Since such objectives are conflicting, the set of necessary remedial works identified with respect to one objective (e.g. costs) would not necessarily optimise another objective (e.g. the total surcharge in the system or the potential flood damage).

There are several ways of dealing with multi-objective problems; we will mention the three most widely applied:

- A number of "good" solutions are found that bring low (but not minimum) values of several objectives, and these solutions are presented to a decision maker who has to choose one solution, a "compromise" (see e.g. Deb et al., 2002).
- Several objective criteria are combined into one, for example in the form of a weighted sum, or by minimising the distance in the objective space from the solution to the "ideal point", and a single-criterion problem is solved.
- Optimisation is performed with respect to one objective, whereas constraints are imposed on the values of all other objectives (for minimisation problems, upper bounds) (as was done, e.g. by Solomatine & Torres (1996)). An optimal solution is the one minimising the selected objective given all constraints are satisfied (in the case of minimisation, this means that the objective value is below the chosen upper bound).

22.2.2 Single-Objective Setting of an Optimisation Problem

In the case of one objective the problem becomes a standard optimisation problem studied already for dozens of years. Very often such a problem has constraints and one of the ways of taking them into account is using a penalty function $p(x)$ with a high value outside the specified constraints, so that the function to be minimised is $f(x)+p(x)$ (for the simplicity of narration the function to be minimised will still be denoted as $f(x)$). Typically the exact value of the optimum vector cannot be found exactly, or this requires too much computer time, so it is reasonable to be satisfied with finding its *estimate*, and correspondingly, the *minimum estimate*. Typically, in the wastewater system optimisation problem, the function f characterises the total cost of remedial works.

In the wastewater system remedial works optimisation problem, the formula for the system performance criterion includes the values of the modelled output which is a result of running a computer program and typically is not analytically defined, so known efficient gradient-based methods cannot be used. There is also no guarantee that $f(x)$ has a single extremum, and therefore such a problem has to be solved by the methods of global (multi-extremum) optimisation.

22.3 Global Search and Methods of Computational Intelligence

22.3.1 Multistart of Local Searches

The basic idea of the family of *multistart* methods is to apply a procedure oriented at searching for a single extremum (referred to as local search) several times, and then to choose an assessment for the global optimiser. There are various ways of choosing the initial points for the local search procedures, for example to choose them randomly, or to apply the Multi-level Single Linkage (MLSL) by Rinnooy Kan and Timmer (1987). Multistart can also be based on the clustering beforehand, i.e. creating groups of mutually close points that hopefully correspond to relevant regions of attraction (Torn and Zilinskas, 1989) of potential starting points. The *region (area) of attraction* of a local minimum x^* is the set of points in X starting from which a given local search procedure P converges to x^*. In an ideal case, the multistart methods aim at starting this local search procedure exactly once in every region of attraction. However, there is no guarantee for that, and it may be necessary to apply the global, rather than local search in each area.

In the GLOBE tool used in this study (Solomatine, 1999) three methods of multistart are implemented – Multis that uses the local derivative-free optimisation method of Powell–Brent (Press et al., 2003), M-Simplex that uses the simplex descent method (Nelder and Mead, 1965) (both use random selection of the starts) and ACCOL, which combines multistart with clustering.

22.3.2 Randomised Search Methods

The most popular approach to global optimisation is, however, based on randomised search. Since derivatives cannot be calculated, global search methods are said to employ *direct search*, i.e. the straightforward calculation of the function values at different points of the search space. The procedure of generating these points often (but not always) uses methods of randomisation. Such an approach needs much more computational power than traditional gradient-based methods and became popular only due to increased capacity of widely available personal computers and workstations. Several representatives of the global search methods are covered below.

For an extensive coverage of various methods the reader is referred to Torn and Zilinskas (1989) and Pinter (1995).

The simplest scheme of obtaining an optimiser assessment, a *pure direct random search* (pure random search, Monte Carlo search or direct search) draws N points from a uniform distribution in X and evaluates f in these points; the smallest function value found is the minimum f^* assessment. If f is continuous, then there is an asymptotic guarantee of convergence, but the number of function evaluations grows exponentially with n. An improvement would be to generate evaluation points in a sequential manner so that the already known function values are taken into account when the next point is chosen.

Price (1983) proposed several versions of the algorithm called *controlled random search* (CRS) in which the new trial point in search (parameter) space is generated on the basis of a randomly chosen subset of previously generated points. The basic idea is that at each iteration, a simplex (in three-dimensional space, it is a tetrahedron) is formed from a sample and a new trial point is generated as a reflection of one point in the centroid of the other points in this simplex; if the worst point in the initially generated set is worse than the new one, it is replaced then by the latter. The ideas of CRS algorithms have also been further extended by Ali and Storey (1994) producing CRS4 and CRS5.

A popular method of direct search was proposed by Nelder and Mead (1965), which we used as the base procedure in the M-Simplex multistart method; however, this method was found by many researchers to be quite effective in finding global minima as well. It is also used as the base procedure in the shuffled complex evolution (SCE) method (Duan et al., 1992), which is popular in water-related applications.

The family of *evolutionary algorithms* is based on the idea of modelling the search process of natural evolution, although these models are crude simplifications of biological reality. Evolutionary algorithms are variants of randomised search and use terminology from biology and genetics. These are covered in more detail by Savic in this volume. Applications of evolutionary algorithms, especially GAs, in water-related optimisation have been widely reported, see e.g. early papers by Wang (1991), Cieniawski et al. (1995) and Savic and Walters (1997).

22.3.3 Adaptive Cluster Covering

The adaptive cluster covering (ACCO) algorithm (Solomatine (1995, 1999) is a workable combination of generally accepted ideas of reduction, clustering and covering:

1. *Clustering*. Clustering (identification of groups of mutually close points in search space) is used to identify the promising subdomains to continue the global search (by active space covering) in each of them.
2. *Covering the shrinking subdomains*. The random covering of each subdomain is applied, i.e. the values of the objective function are assessed in the points drawn

from a uniform or some other distribution. Covering is repeated multiple times and each time the subdomain is progressively reduced in size.
3. *Adaptation*. Adaptive algorithms are updating their algorithmic behaviour depending on the new information revealed about the problem under consideration. In ACCO, adaptation is in *shifting* the subregion of the search, *shrinking* it and changing the density (number of points) of each covering – depending on the previous assessments of the global minimiser.
4. *Periodic randomisation*. Due to the probabilistic character of point generation, any strategy of randomised search may simply miss a promising region for search (similar to the situation when local search procedures may miss the global minimum). In order to reduce this danger, it is reasonable to re-randomise the initial population, i.e. to solve the problem several times, and/or to re-randomise some populations at intermediate steps.

The version of ACCO used here is the ACCOL algorithm – it combines ACCO with multiple local searches using the Powell–Brent procedure.

22.4 Problem Formulation

The constraints in the problem can be grouped into the following: the dynamic nature of the system, uncontrolled surcharge (i.e. overflows) and economic drivers (the total cost of remedial works involved). The system dynamics constraint is handled by the hydrodynamic network simulation model. The optimisation package handles only box-type constraints on the parameters, i.e. upper and lower bounds on each parameter, and therefore, the penalty functions are used to handle the system surcharge constraint. The economic constraint imposes the reduction of the parameter space to a discrete one. GLOBE has an option to fix the resolution of the parameter space to be searched. This can be adjusted by the number of available pipe sizes and each parameter can take values from one of the pipe sizes. This number is used as an index for the choice of diameters; therefore, the search algorithms will search for the optimal set of pipe indices instead of the optimal set of diameters. This approach has the following technical advantages: (a) search algorithms will not waste computational time looking for diameters in a real parameter space and (b) the solutions obtained will not be split pipe solutions.

The problem is posed as a multi-objective problem with the two objective functions:

1. Minimisation of the total flood damage related to surcharge
2. Minimisation of the total network rehabilitation costs

The corresponding equations are given below:

$$f_1 = \mathrm{Min}\,\Delta\left(\sum_{i=1}^{n}\left(Q/Qf\right)_i\right)\Big/\,\Delta\left(\sum_{i=1}^{n}\left(Q/Qf\right)_i^{\max}\right) \tag{22.1}$$

$$f_2 = \text{Min}\,\Delta\left(\sum_{i=1}^{n}(L_i\,{}^*C_i)\right)\Big/\,\Delta\left(\sum_{i=1}^{n}(L_i\,{}^*C_i^{\max})\right) \tag{22.2}$$

where the expression *Q/Qf* is a surcharge indicator, *Q* is the actual maximum discharge and *Qf* is the capacity of the pipe at full section; values greater than one indicate the grade of surcharge in the pipe. In (22.2) (L^*C) represents the cost of the pipe replacement. The *max* superscript means "maximum value expected" – this ensured normalisation of the functions. No additional constraints have been imposed; however, constraints due to hydraulic conditions are internally imposed in the hydrodynamics model.

In the single-objective formulation the objective function to be minimised was expressed as the weighted sum of the two cost components:

$$C = wf_1 + (1-w)f_2 \tag{22.3}$$

In the test cases considered in this study we used $w = 1$ (note that the contributing functions are normalised). Other ways to combine the multiple objectives mentioned above will be considered in future studies.

22.5 Tools Used

The tools used in the proposed approach are MOUSE for hydrodynamic modelling (Danish Hydraulic Institute), GLOBE for single-objective optimisation and NSGAX for multi-objective optimisation.

Optimisation tool GLOBE. In order to apply global optimisation techniques to the present problem, a PC-based system GLOBE has been utilised. GLOBE (Solomatine 1995, 1999, http://www.ihe.nl/hi) can be configured to use an external program as the supplier of the objective function values. The number of independent variables and the constraints imposed on their values are supplied by the user in the form of a simple text file. The three modules (i.e. programs) such as the MOUSE model and two transfer programs (i.e. processors) are activated from GLOBE in a loop. It iteratively runs an executable program that receives potential solutions generated by the search algorithms and returns a corresponding value of the objective function. Currently, GLOBE includes the following nine algorithms: several versions of controlled random search (Price, 1983; Ali and Storey, 1994) GA, Multis, M-Simplex, ACCO, ACCOL, adaptive cluster descent (ACD) and adaptive cluster descent with local search (ACDL) (Solomatine, 1999).

Optimisation tool NSGAX. This tool implements NSGA-II, the multi-objective optimisation algorithm of Deb et al. (2002), and uses C code downloaded from the web site of the Kanpur Genetic Algorithms Laboratory (KanGAL). They were modified to call an external function to calculate the objective function values based on the model run.

Pipe network simulation tool MOUSE. The MOUSE pipe flow model used here (a product of DHI Water & Environment) is a computational tool for simulations of unsteady flows in pipe networks. The computation is based on an implicit, finite difference numerical solution of basic one-dimensional, free surface flow equations (Saint-Venant). This algorithm provides a solution in multiple connected branched and looped pipe networks. The computational scheme is applicable to vertically homogeneous flow conditions, which occur in pipes ranging from small-profile collectors for detailed urban drainage, to low-lying, often pressurised sewer mains, affected by the varying water level at the outlet. Both subcritical and supercritical flows are treated by means of the same numerical scheme that adapts according to the local flow conditions.

22.5.1 Algorithmic and System Setup

Based on the ideas outlined by Vojinovic and Solomatine (2005a,b; 2006), the following steps are used to calculate the cost of one set of remedial works (Fig. 22.1):

1. The values generated by GLOBE (or NSGAX) are read from the parameter file and converted to indices of pipe sizes that represent one network.
2. The actual rehabilitation cost of the network f_2 is calculated based on the length and cost per unit length corresponding to the diameter of each pipe.

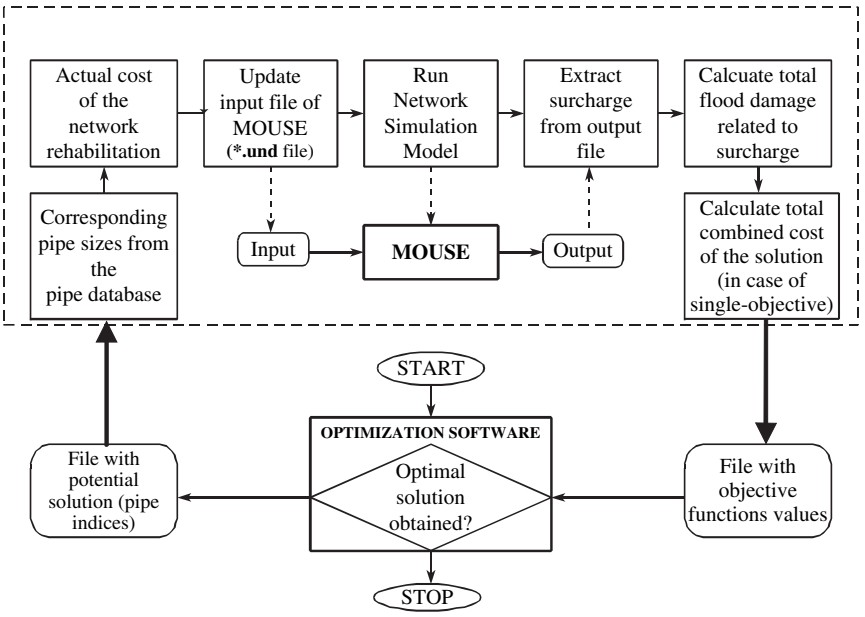


Fig. 22.1 Problem setup using optimisation software GLOBE and NSGAX

3. The MOUSE input file is updated (only diameters are changed).
4. The MOUSE model is run.
5. From the output file of the simulation, the nodal surcharges are extracted and summed representing the total system surcharge.
6. The surcharge cost f_1 is calculated based on the degree of the total system surcharge (if any).
7. The total objective function $C = g(f_2, f_2)$ value is calculated (in the case of a single-objective approach and using GLOBE), using, e.g. (22.3).
8. The values f_2, f_2 (in the case of the multi-objective approach), or the values of C (in the case of a single-objective approach) are written to the response file, which is then read by GLOBE or NSGAX.

Additionally, in this study, in order to reduce the problem dimension and hence the computational burden, we run MOUSE prior to optimisation a number of times to determine the critical pipes where a surcharge threshold is exceeded. Only such pipes are subsequently included in the optimisation process.

22.6 Case Study

The methodology has been tested on several real networks. One of them, a simplified combined sewer subsystem of an urban catchment in Japan, is reported in this chapter (Fig. 22.2). The considered system contains 12 pipes, 12 nodes and 1 outlet. A design rainfall event of a 1 in 5 year return period was applied for evaluation of remedial options and the system's upgrade. One hydrodynamic simulation for this system using this particular event was found to last around 3 minutes. A 1.4 GHz PC was used.

The problem of remediation was posed as indicated above. In the single-objective setting the composite objective function combined the flood damage and the rehabilitation costs by weighting them (22.3) and the ACCO algorithm was used. In the multi-objective setting two objectives were considered: total flood damage related to surcharge and the total network rehabilitation costs (both to be minimised).

The single-objective optimisation served as a sort of "feasibility study" for the subsequent multi-objective optimisation and helped to identify pipes corresponding to the exceeded surcharge threshold. Only those pipes were included in the optimisation process, so that out of 12 pipes only 4 pipes were optimised.

22.7 Results

Since one MOUSE model run takes around 3 minutes, the algorithm was first tested in the single-objective setting on a limited number of model runs (between 40 and 50), and for that purpose its parameters had to be set accordingly (see Solomatine,

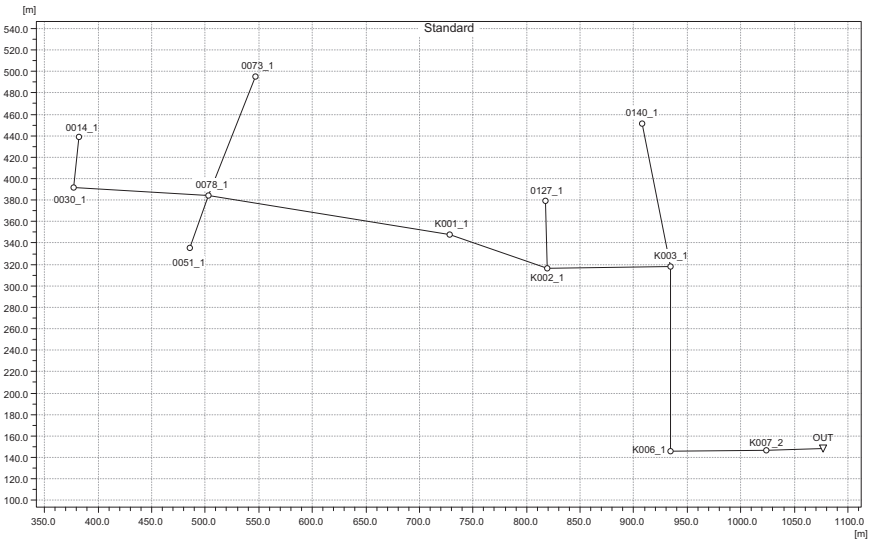


Fig. 22.2 Network under study

1999). Even when such limited optimisation was employed, the total cost dropped 50% from the initial value.

A limited investigation of the effectiveness (how well it solves the problem) and efficiency (how fast it is solving the problem) of the optimisation algorithm was undertaken. Figure 22.3 presents the plot showing how fast (i.e. after how many model runs) the total cost associated with the network remediation is going down (the first cost value is recorded after six initial model runs by the ACCO algorithm). The full investigation of comparative effectiveness of various single-objective optimisation algorithms is yet to be done.

In a multi-objective setting, using NSGA-II a more comprehensive study was undertaken. We tested several population sizes – 20, 32 and 64. Crossover probability was 1, and the mutation probability was chosen to be 1/*np*, where *np* is the number of pipes to optimise.

For each population set four runs were performed; using different seeds for the random number generator, the selected indicators were averaged for each population size and each optimiser. Table 22.1 shows the averaged performance indicators for various population sizes.

Table 22.1 Summary of the NSGA-II results

Population size	# Solutions in the Pareto layer	# Function evaluations	Runtime (min)
20	18.00	440	26.4
32	27.25	468	25.3
64	44.25	785	45.8

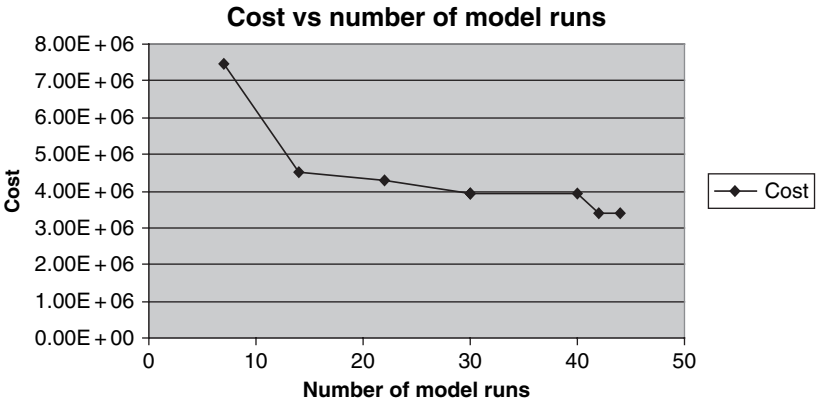


Fig. 22.3 Composite cost value (non-normalised) as a function of the number of model runs in single-objective optimisation

Two solutions taken from the NSGA-II Pareto front are presented in Table 22.2. It can be seen how the change of diameter influences the trade-off between the two objective functions.

We have also calculated and analysed the metrics traditionally used in multi-objective optimisation – hypervolume and ε-indicator, which will be reported elsewhere due to lack of space.

Multi-objective optimisation does not answer the question of which solution is the best, but only provides a decision maker with a set of alternatives. If one compares these two solutions using a combined criterion (22.3) then solution 2 would be the preferred choice. If, however, one is interested in the minimum-cost solution, which does not lead to flood damage larger than a predefined threshold, then solution 1 could be chosen since its implementation is more than five times cheaper than that of solution 2.

Table 22.2 Values of variables and objective functions for the two different Pareto vectors

Solution	Pipe ID	Catalogue #	Diameter (mm)		Objectives	
			Initial	Final	Cost	Surcharge
1	0078_111	1	225	225	0.0095	0.7190
	K003_111	1	225	225		
	K006_111	1	225	225		
	K007_211	2	225	470		
2	0078_111	2	225	470	0.1160	0.0081
	K003_111	4	225	692		
	K006_111	3	225	575		
	K007_211	2	225	470		

22.8 Conclusions

In this chapter, an approach to optimisation of wastewater system remedial works was presented. In this approach, the search for an optimal solution was performed by either a single-objective method (e.g. ACCO or other methods implemented in the global optimisation tool GLOBE) or a multi-objective optimisation method (NSGA-II). The dynamic nature of the wastewater pipe network system was replicated with commercially available hydrodynamic software, MOUSE. The method was tested on simple systems and this (still preliminary) work suggests a strong potential for this approach to find the optimal solution and achieve economic benefits when compared to the approaches traditionally applied by engineers.

Further research will be aimed at testing the methodology on a larger system, and at designing algorithms that would be computationally less demanding. Attention should also be given to using practitioners' knowledge in formulating the objective functions (e.g. rehabilitation phased in time), and explicitly taking into account specific urban drainage system features during optimisation.

References

Abebe AJ, Solomatine DP (1998) Application of global optimization to the design of pipe networks. Proc. 3rd Intern Conf. Hydroinformatics, Copenhagen, 1998, pp 989–996.

Alperovits E, Shamir U (1977) Design of optimal water distribution systems. Water Resources Research 13(6): 885–900.

Ali MM, Storey C (1994) Modified controlled random search algorithms. International Journal of Computer Mathematic 53: 229–235.

Babovic V, Wu Z, Larsen LC (1994) Calibrating hydrodynamic models by means of simulated evolution, Proc. Int. Conf. on Hydroinformatics, Delft, The Netherlands. Balkema, pp 193–200.

Cieniawski SE, Eheart JW, Ranjithan S (1995) Using genetic algorithms to solve a multiobjective groundwater monitoring problem. Water Resources Research 31: 399–409.

Dandy GC, Simpson AR, Murphy LJ (1996) An improved genetic algorithm for pipe network optimization. Water Resources Research 32(2): 449–458.

Deb K, Pratap A, Agarwal S, Meyarivan, T (2002) A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Transaction on Evolutionary Computation 6(2): 181–197.

Diogo AF, Walters GA, de Sousa ER, Graveto VM (2000) Three-dimensional optimization of urban drainage systems. Computer Aided Civil and Infrastructure Engineering 15: 409–426.

Duan Q, Sorooshian S, Gupta VK (1992) Effective and efficient global optimization for conceptual rainfall-runoff models. Water Resources Research 28(4): 1015–1031.

Franchini M, Galeati G (1997) Comparing several genetic algorithm schemes for the calibration of conceptual rainfall-runoff models. Hydrological Sciences Journal 42(3): 357–379.

Kapelan Z, Savic DA, Walters GA, Babayan AV (2005) Risk- and robustness-based solutions to a multi-objective water distribution system rehabilitation problem under uncertainty. Water Science & Technology 53(1): 61–75.

Kessler A, Shamir U (1989) Analysis of the linear programming gradient method for optimal design of water supply networks. Water Resources Research 25(7): 1469–1480.

Michalewicz Z (1999) Genetic algorithms + data structures = evolution programs. Springer, Berlin.

Nelder JA, Mead R (1965) A simplex method for function minimization. Computing Journal 7: 308–312.

Pinter J (1995) Global optimisation in action. Kluwer, Amsterdam.

Prasad TD, Park N (2004) Multiobjective genetic algorithms for design of water distribution networks. ASCE Journal of Water Resources Planning and Management 130(1): 73–82.

Press WH, Flannery BP, Teukolsky SA, Vetterling WT (2003) Numerical recipes in C++. The art of scientific computing. Cambridge University Press, Cambridge, USA.

Price WL (1983) Global optimisation by controlled random search. Journal of Optimistic Theory Application 40: 333–348.

Rauch W, Harremoes P (1999) On the potential of genetic algorithms in urban drainage modeling. Urban Water 1: 79–89.

Rinnooy Kan, AHG, Timmer GT (1987) Stochastic global optimization methods. Part 1: clustering methods. Mathematical Programming: Series A and B 39(1): 27–56.

Savic DA, Walters GA (1997) Genetic algorithms for the least-cost design of water distribution networks. ASCE Journal of Water Resources Planning and Management 123(2), 67–77.

Simpson AR, Dandy GC, Murphy LJ (1994) Genetic algorithms compared to other techniques for pipe optimization. ASCE Journal of Water Resources Planning and Management 120(4), 423–443.

Solomatine DP (1995) The use of global random search methods for models calibration, Proc. XXVIth Congress of the International Association for Hydraulic Research (IAHR), vol. 1, pp 224–229.

Solomatine DP, Torres LA (1996) Neural network approximation of a hydrodynamic model in optimizing reservoir operation. Proc. 2nd Intern. Conf. on Hydroinformatics, Zurich, September 1996, pp. 201–206.

Solomatine DP (1999) Two strategies of adaptive cluster covering with descent and their comparison to other algorithms. Journal of Global Optimization 14 (1): 55–78.

Tang Y, Reed P, Wagener T (2005) How effective and efficient are multiobjective evolutionary algorithms at hydrologic model calibration? Hydrology Earth System Science Discussion 2, 2465–2520.

Torn A, Zilinskas A (1989) Global optimisation. Springer-Verlag, Berlin.

Wang QJ (1991) The genetic algorithm and its application to calibrating conceptual rainfall-runoff models. Water Resources Research 27: 2467–2471.

Vojinovic Z, Solomatine DP (2005a) Dynamic Least-Cost Optimisation of Wastewater Systems Remedial Works Requirements, 10th International Conf. on Urban Drainage, Copenhagen, Denmark, August 2005.

Vojinovic Z, Solomatine DP (2005b) Multi-criteria global evolutionary optimisation approach to rehabilitation of urban drainage systems, EGU General Assembly, Vienna, April 2005. Geophysical Res. Abstracts, Vol. 7, p. 10720.

Vojinovic Z, Solomatine DP (2006) Dynamic least-cost optimisation of wastewater system remedial works requirements. Water Science and Technology 54(6–7): 467–475.

Yates DF, Templeman AB, Boffey TB (1984). The computational complexity of the problem of determining least capital cost designs for water supply networks. Engineering Optimization 7(2): 142–155.

Chapter 23
Neural Network Hydrological Modelling: An Evolutionary Approach

A.J. Heppenstall, L.M. See, R.J. Abrahart, and C.W. Dawson

Abstract Neural networks are now extensively used for rainfall–runoff modelling. Considered a "black box" approach, they allow non-linear relationships to be modelled without explicit knowledge of the underlying physical processes. This method offers many advantages over earlier equation-based models. However, neural networks have their own set of issues that need to be resolved, the most important of which is how to best train the network. Genetic algorithms (GAs) represent one method for training or breeding a neural network. This study uses JavaSANE, a system that advances on traditional evolutionary approaches by evolving and optimising individual neurons. This method is used to evolve good performing neural network rainfall–runoff solutions for the River Ouse catchment in the UK. The results show that as lead times increase, the JavaSANE networks outperform conventional feedforward networks trained with backpropagation.

Keywords Genetic algorithms · neural networks · rainfall–runoff modelling

23.1 Background

Hydrological systems are complex entities containing numerous non-linear processes which are often interrelated at different spatial and temporal scales. This complexity presents substantial problems in both understanding and modelling hydrological processes. This is particularly problematic in the area of rainfall–runoff modelling (Zhang and Govindaraju, 2000). The development of mathematical models to understand

A.J. Heppenstall
School of Geography, University of Leeds, Woodhouse Lane, Leeds, LS2 9JT, UK

L.M. See
School of Geography, University of Leeds, Woodhouse Lane, Leeds, LS2 9JT, UK

R.J. Abrahart
School of Geography, University of Nottingham, Nottingham, NG7 2RD, UK

C.W. Dawson
Department of Computer Science, Loughborough University, Loughborough, LE11 3TU, UK

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

this system occupies a substantial amount of the literature where efforts are concentrated on attempting to capture the characteristics of the underlying physical processes through the use of equations of mass and momentum (Jain and Srinivasulu, 2004). However, many of these models have varying degrees of success being limited by the need for large quantities of high-quality data as well as detailed calibration and optimisation methods.

Due to the constraints of these physically – based models, attention has shifted to the use of "black box" methods such as neural networks. Neural networks can be used to develop relationships between input and output variables that form part of a process without having explicit knowledge of the underlying physics. These models are now extensively used to perform rainfall–runoff modelling (see other chapters in this book on neural network modelling for many good examples). This uptake is due to the advantages that they possess over traditional modelling solutions. These include inherent capabilities such as learning, adapting and generalising to unseen data sets as well as performing distributed parallel processing. Neural networks also offer associative memory storage and exhibit robustness and fault-tolerant characteristics (Kasabov, 1996). However, these tools are not without their disadvantages. These include a need for large data sets, long training times, lack of guidance on architecture and parameter settings and the potential to overfit. Furthermore, there is a tendency to employ the same objective function, generally sum-squared error, which may not always be the most appropriate for rainfall–runoff modelling.

An important element in developing a successful neural network model lies in the method used for training the network. Methods that have been investigated range from the momentum correction factor (Raman and Sunilkumar, 1995) to the commonly used backpropagation training algorithm (Rajurkar et al., 2002). However, research by Hsu et al. (1995) suggested that the results of these training methods were both inconsistent and unpredictable due to their inability to capture the non-linearity in the rainfall–runoff process. This is in part related to the inability of such methods to search for the global optima within a solution space that is both discontinuous and has significant variations over the input space.

This chapter will briefly describe genetic algorithms and how they are implemented in the package called JavaSANE, which has been used to evolve neural network rainfall–runoff models for the River Ouse catchment in the UK. Neural networks trained with backpropagation and multiple regression models were developed in parallel to provide a comparison. The results of applying these three methods are presented as well as a discussion of the potential benefits and problems associated with using a genetic approach.

23.2 Genetic Algorithms

The training of a neural network is essentially a non-linear optimisation problem whereby the objective is to minimise the global error at the output level. A tool that is widely used for solving complex non-linear optimisation problems is the genetic algorithm (GA).

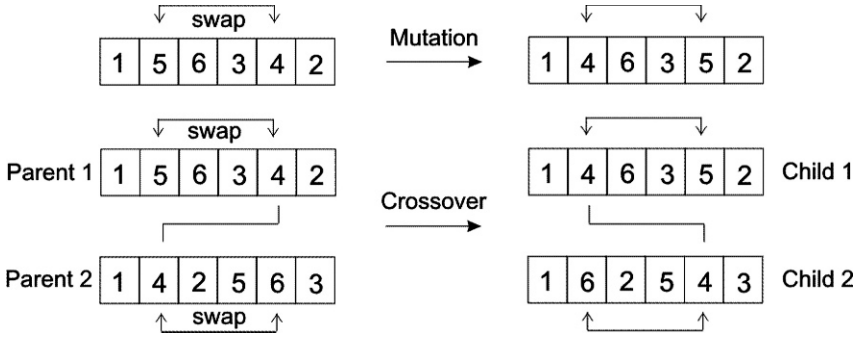


Fig. 23.1 The crossover and mutation operators applied to candidate solutions of a combinatorial problem (after Flake, 2001)

GAs are modelled on natural evolutionary processes, known as genetic operators. These manipulate individuals in a population over several generations to improve their fitness. A detailed introductory survey can be found in Reeves and Rowe (2003).

In a GA, the properties of each individual are represented in an encoded form known as a chromosome (or genome). Chromosomes are combined or mutated to breed new individuals. Crossover of two chromosomes models the sexual reproduction occurring in nature. An offspring's chromosome is created by joining segments chosen alternately from each of two parents' chromosomes which are of fixed length. Selection is the process of choosing the chromosomes to be recombined. Mutation is the alteration of one or more parts of the chromosome with a random probability. These operators are illustrated in Fig. 23.1 while a simple schematic of how a GA operates is provided in Fig. 23.2. The algorithm is very simple with the main functions contained in the innermost loops. These include the process of selection, crossover and mutation. The main difference between GAs and other classical optimisation search techniques is simply that the GA works with a population of possible solutions, whereas the classical optimisation techniques work on a single solution.

```
Initialise the population P
Repeat for some length of time
        Create an empty population, P1
        Repeat until P1 is full:
    SELECT two individuals from P based on some fitness criteria
    Optionally mate and RECOMBINE with offspring
    Optionally MUTATE the individuals
    Add the two individuals to P1
    Let P now be equal to P1
```

Fig. 23.2 Basic structure of a GA (after Flake, 2001)

GAs have previously been combined with neural network solutions in a number of different ways. These include the evolution of a set of neural network weights (with or without further training using a gradient descent mechanism), the evolution of entire neural network solutions and the automatic calibration of rainfall–runoff models (Abrahart et al., 1999; Franchini, 1996; Jain and Srinivasulu, 2004; Shamseldin and O'Connor, 2001; Wang, 1991; Whitley et al., 1990). In these applications the GAs were used to converge a population into a simple individual "best solution".

23.3 Cooperative Coevolution and JavaSANE

The methodology presented here represents an advance on the classic GA approach by evolving a population of neurons rather than a population of neural networks. This is the basis of the SANE (symbiotic adaptive neuro-evolution) system (Moriarty and Miikkulainen, 1998) and is constructed from the notion of *cooperative coevolution* (Horn et al., 1994; Potter, 1997).

Cooperative coevolution emphasises the role of the individual neuron as a partial solution within a population of neurons. To develop a complete solution (i.e. the optimal network), individuals are first evolved via the genetic operators of mutation, crossover and selection. The neurons are then combined to optimise one part of the solution space whilst cooperating with other partial solutions through the development of connections with other neurons.

As a result of applying genetic operators to individual neurons, convergence towards a single type of individual is avoided and diversity is preserved within the population. This avoids convergence towards a suboptimal peak and allows the population to adapt to changes. Other advantages that cooperative coevolution has over traditional GA solutions are speed and efficient search of the solution space.

A Java version of the SANE system has recently been developed called JavaSANE. JavaSANE contains important modifications to the cooperative coevolutionary method. One of these modifications is the creation of records or "blueprints" of the most effective neuron combinations that occur within the current population. These are used to build neural networks in the next generation and concentrate the search efforts towards finding optimal neural network solutions. The main advantage of evolving network blueprints is the potential to exploit the best networks discovered during the evolution process. This results in the best neuron combinations being recombined to form fresh, and potentially better, collections of neurons. Evolution at the blueprint level thus provides a very aggressive search procedure.

JavaSANE therefore maintains and evolves two populations: a population of neurons and a population of network blueprints. Each individual neuron in the population specifies a set of connections that need to be made within a neural network. The neuron evolution searches for effective partial networks, while the blueprint evolution searches for effective combinations of the partial networks. Each individual in the neuron population represents a hidden neuron in a one-hidden-layer feedforward network. Neurons encode the weighted connections and where the

connections should be made and connect to the input and output layer. The basic steps in the algorithm are:

1. Specify the number of network input neurons, network hidden neurons, network output neurons, maximum neuron population, maximum blueprint population and maximum number of iterations (fixed).
2. Select a certain number of neurons from the population using a blueprint.
3. Create a neural network from a mixture of selected neurons.
4. Evaluate the network.
5. Set the blueprint fitness value to be the network evaluation score.
6. Repeat steps 2–4 for each individual in the blueprint population.
7. Assign each neuron the sum of the fitness evaluations of the best five networks in which it participated.
8. Kill weaker performing blueprints and neurons.
9. Breed new members based on crossover and mutation operations.
10. Repeat steps 2–10 until a maximum number of specified iterations has been completed.

23.4 Models of the River Ouse

Three types of models were developed for the River Ouse catchment at Skelton (see Abrahart et al. (2007) for background details to this study area):

- JavaSANE neural network models (JSNN)
- Neural network models trained with backpropagation (BPNN)
- Multiple linear regression models (MLR)

Four lead times were chosen in order to look at the performance over longer time steps: T+6 hours, T+12 hours, T+18 hours and T+24 hours. Each model was developed on a training data set covering one historical winter period (1993/94) and evaluated using two independent test data sets (Test 1 covering a normal winter period (1994/95) and Test 2 covering a drought period in 1995/96). Historical data for model development were available for the gauging station at the prediction point (Skelton, Q) as well as three upstream stations (Crakehill (U1), Skip Bridge (U2) and Westwick (U3)) and five rain gauges (Tow Hill (R1), Arkengartdale (R2), East Cowton (R3), Osmotherly (R4) and Malham Tarn (R5)). The model inputs were determined using correlation analysis and are listed in Table 23.1.

Table 23.1 Model inputs and outputs

Model inputs	Output
$U1_T$, $U2_T$, $U3_{T-6}$, $R1_{T-24}$, $R2_{T-24}$, $R3_{T-24}$, $R4_{T-30}$, $R5_{T-24}$, Q_T	Q_{T+6}
$U1_T$, $U2_T$, $U3_T$, $R1_{T-18}$, $R2_{T-18}$, $R3_{T-18}$, $R4_{T-24}$, $R5_{T-18}$, Q_T	Q_{T+12}
$U1_T$, $U2_T$, $U3_T$, $R1_{T-12}$, $R2_{T-12}$, $R3_{T-12}$, $R4_{T-18}$, $R5_{T-12}$, Q_T	Q_{T+18}
$U1_T$, $U2_T$, $U3_T$, $R1_{T-6}$, $R2_{T-6}$, $R3_{T-6}$, $R4_{T-12}$, $R5_{T-6}$, Q_T	Q_{T+24}

JavaSANE was run on its default settings with six hidden neurons and sum-squared error (SSE) as the optimisation function. The model does not require a cross-validation data set so the best model was selected based on overall performance on the training data set. Neural networks trained with backpropagation were similarly trained on only the training data set and allowed to run for a long time (i.e. 20,000 epochs). Multiple linear regression models were developed using the same set of inputs and again using only the training data set. The resulting models were then applied to the Test 1 and Test 2 data sets.

A suite of goodness-of-fit tests was employed to examine the performance of each solution. These included root mean squared error (RMSE) defined in metres, coefficient of efficiency (CE) (Nash and Sutcliffe, 1970), mean absolute error (MAE) in metres and the *r*-squared (r^2). The results are provided in Table 23.2 and were produced using the HydroTest website (Dawson et al., 2007).

Table 23.2 Training, Test 1 and Test 2 statistics for different models. Shading denotes the best performing model within a given test data set

Lead time	Data set	Model	RMSE	MAE	CE	r^2
T+6	Test1	MLR	0.2597	0.1646	0.9947	0.9496
		BPNN	0.1807	0.1269	0.9731	0.9875
		JSNN	0.2240	0.1513	0.9587	0.9798
	Test2	MLR	0.1362	0.1132	0.9509	0.9683
		BPNN	0.1272	0.1001	0.9572	0.9892
		JSNN	0.1495	0.1135	0.9409	0.9823
T+12	Test1	MLR	0.3075	0.2000	0.9221	0.9270
		BPNN	0.2650	0.1803	0.9422	0.9450
		JSNN	0.3141	0.2169	0.9187	0.9206
	Test2	MLR	0.1534	0.1214	0.9377	0.9563
		BPNN	0.1685	0.1298	0.9248	0.9474
		JSNN	0.1940	0.1552	0.9003	0.9456
T+18	Test1	MLR	0.4313	0.2898	0.8465	0.8488
		BPNN	0.4334	0.2758	0.8450	0.8536
		JSNN	0.4792	0.3529	0.8104	0.8234
	Test2	MLR	0.2387	0.1986	0.8490	0.9031
		BPNN	0.2187	0.1372	0.8733	0.8813
		JSNN	0.2921	0.2089	0.7738	0.8205
T+24	Test1	MLR	0.4789	0.3219	0.8105	0.8127
		BPNN	0.5936	0.4161	0.7089	0.8422
		JSNN	0.5614	0.4083	0.7396	0.8671
	Test2	MLR	0.2817	0.2277	0.7895	0.8542
		BPNN	0.5272	0.4053	0.2629	0.7335
		JSNN	0.2718	0.1760	0.8041	0.9095

23.5 Results and Discussion

Table 23.2 clearly shows that the performance statistics exhibits a general degradation across all models as the lead times increase from T+6 to T+24. There is no overall winning model, a reflection of the increasing difficulties involved in modelling longer lead times. At T+6 and T+12, the BPNN outperforms the other models. The solutions here are simple and linear suiting the more efficient gradient descent method of optimisation. At longer lead times the situation is more mixed and no single model is the best performer overall.

Figure 23.3 shows the behaviour of the models with regard to RMSE for the different lead times and for the Test 1 and Test 2 data sets. For the Test 1 data set the BPNN solution produces its best performance at T+6 and T+12. However, a clear crossover occurs at longer lead times where both the MLR and JSNN models outperform the BPNN at T+24. This reflects the increasing complexity of the solution and the inability of the BPNN to cope. A similar phenomenon can be seen for the Test 2 data set.

Figures 23.4–23.6 show scatterplots of the actual vs predicted values for the Test 1 and Test 2 data sets. The plots on the left correspond to T+6 while those on the right are for a lead time of T+24. Examination of the MLR at T+6 highlights the models' poor performance at the upper ends; this is most likely where the non-linear part of the relationship exists. The situation worsens at T+24. The BPNN handles this non-linearity at T+6 but the widespread nature of the points around the line of perfect agreement at T+24 shows that the model is not performing as well.

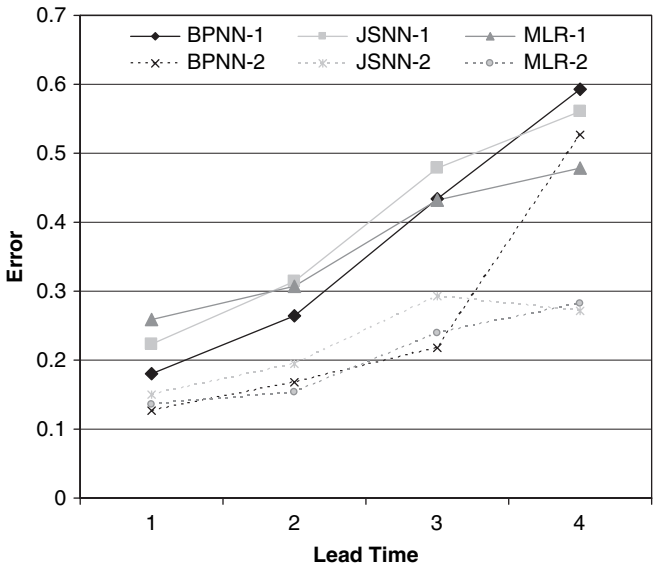


Fig. 23.3 RMSE of different models as the lead time increases

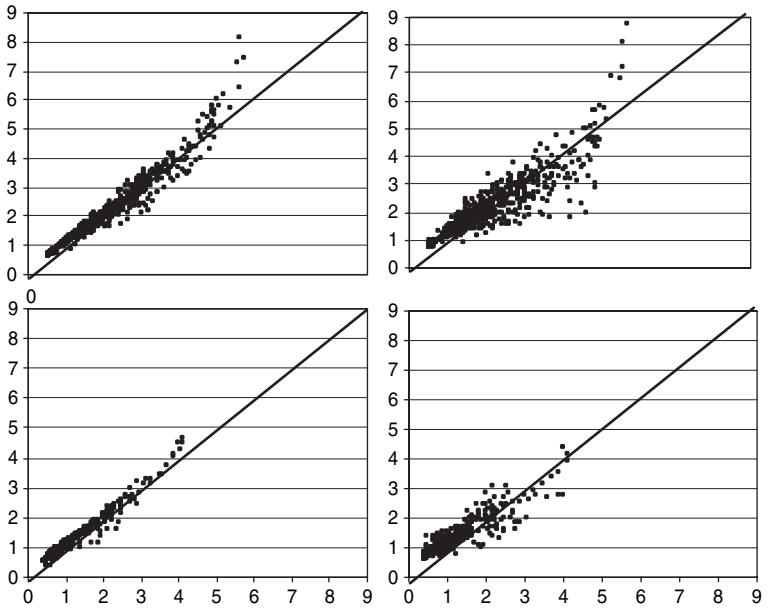


Fig. 23.4 Actual vs predicted normalised river levels from the MLR models for T+6 (*left*) and T+24 (*right*) for the Test 1 (*top*) and Test 2 (*bottom*) data sets

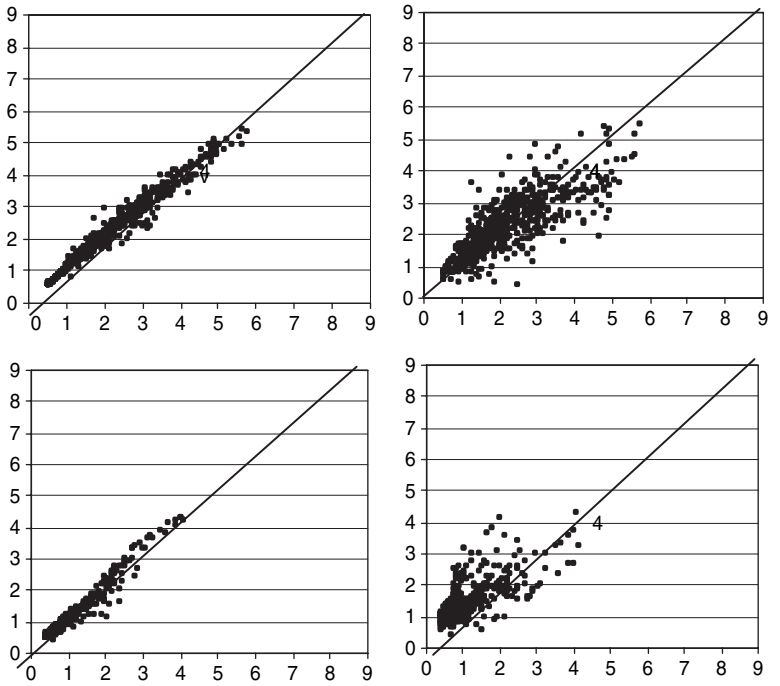


Fig. 23.5 Actual vs predicted normalised river levels from the BPNN models for T+6 (*left*) and T+24 (*right*) for the Test 1 (*top*) and Test 2 (*bottom*) data sets

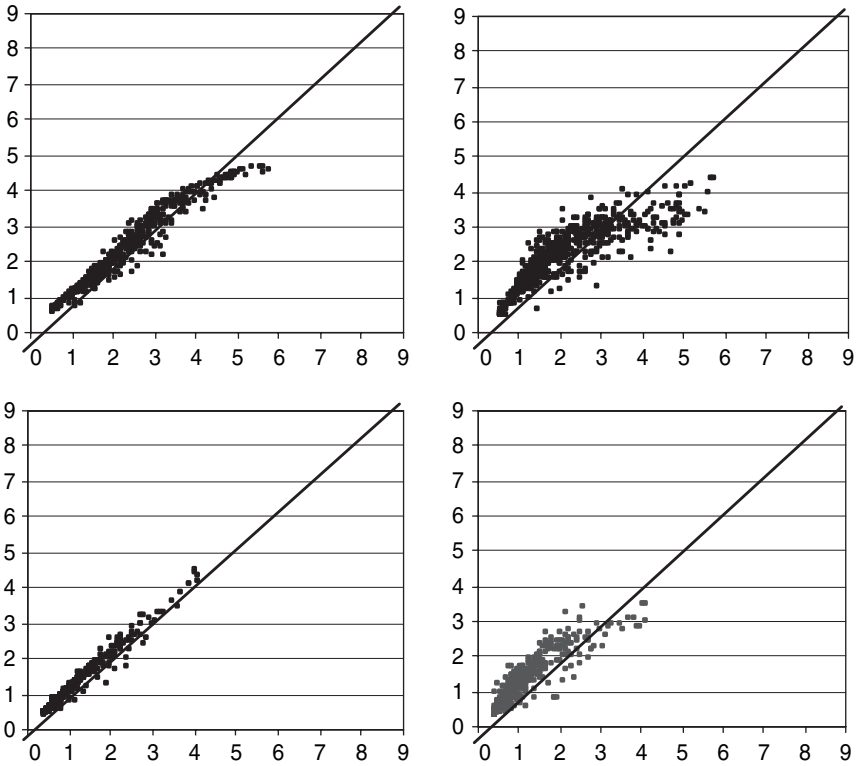


Fig. 23.6 Actual vs predicted normalised river levels from the JSNN models for T+6 (*left*) and T+24 (*right*) for the Test 1 (*top*) and Test 2 (*bottom*) data sets

The JSNN model is generally undershooting at the upper ends but is able to handle the non-linearity of the relationship, especially at T+24 where it is clearly the best performing model.

A final way to examine the behaviour of the model is through looking at a time series plot. Figure 23.7 shows the three model predictions for a lead time of T+6 for the Test 2 data set. Both the MLR and BPNN models overpredict at low flows. The BPNN in particular overpredicts the peaks. However, the JSNN model fits the data better at all levels but tends to underpredict the higher peaks.

23.6 Conclusions

This chapter has presented an adaptive coevolutionary approach for rainfall–runoff modelling. The results presented above demonstrate the potential of the JavaSANE toolbox. In each of the experiments performed, the solutions captured the main patterns of the rainfall–runoff relationship. Degradation in results between T+6 and

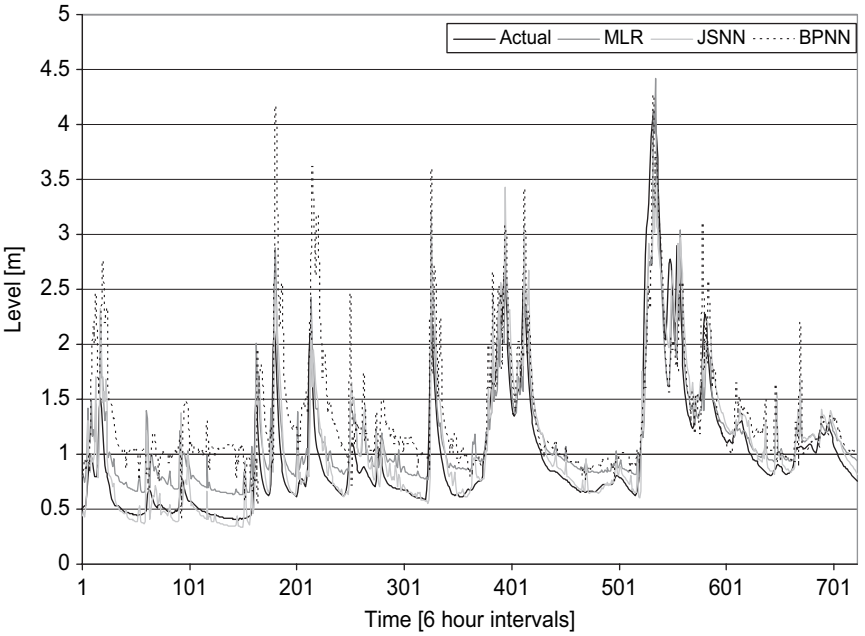


Fig. 23.7 Model performance at T+6 on the Test 2 data set

T+24 hour predictions was found reflecting the difficulty in modelling longer lead times. These results were compared with a conventional feedforward neural network trained using backpropagation. It was found that the backpropagation model did not perform as well as the models developed using JavaSANE. The solutions obtained by JavaSANE were produced using the default settings. There is clearly room for improvement of results with further experimentation of JavaSANE's parameters.

There are several clear benefits to adopting JavaSANE. Within rainfall–runoff modelling, there is a tendency to use the same objective function. JavaSANE has the ability to employ one of several alternative objective functions such as relative error or CE. This diversity strengthens the modelling approach and could (ultimately) result in more accurate predictions/forecasts. Through the use of blueprints storing the most effective neuron combinations, JavaSANE is able to search larger areas of the solution space more quickly than traditional evolutionary algorithms. Furthermore, the approach can be implemented effectively with both large and small data sets, and validation data are not required.

In terms of usability, JavaSANE is extremely simple to operate, individual parameters can be adjusted with ease and new objective functions quickly incorporated. Furthermore, JavaSANE is constructed using the platform-independent language, Java; this allows the program to be run easily on any operating system that has the Java virtual machine installed.

Perhaps one of the more interesting scientific aspects of this method could lie in the hydrological interpretation of hidden neurons. This is an area of growing

interest in which specialist neurons are identified and their respective roles are matched against recognised catchment processes (Wilby et al., 2003; Jain et al., 2004; Sudheer and Jain, 2004). The neurons in the cooperative coevolution method are encouraged to evolve into different individual and collective specialisations; therefore it is important to establish whether or not these specialisations are (a) different from those found in a conventional neural network model and (b) whether or not it is possible for the hidden neurons to provide pertinent information with regard to the real hydrological processes that occur within the actual river basin concerned.

JavaSANE is a powerful addition to the hydrologists' tool box. Its advantages and potential for rainfall–runoff modelling have been outlined here. However, this approach is not limited to this area of hydrological modelling but can be exploited in other application domains such as groundwater modelling, water quality and sediment estimation.

References

Abrahart, R.J., See, L., Kneale, P.: Using pruning algorithms and genetic algorithms to optimise network architectures and forecasting inputs in a neural network rainfall-runoff model. Journal of Hydroinformatics 1 (1999) 103–114.

Abrahart, R.J., See, L.M., Heppenstall, A.J.: Neuroevolution applied to river level forecasting under winter flood and drought conditions. Journal of Intelligent Systems 16 (2007).

Dawson, C.W. Abrahart, R.J., See, L. HydroTest: a web-based toolbox of statistical measures for the standardised assessment of hydrological forecasts. Environmental Modelling and Software 27 (2007) 1034–1052.

Flake, G.W.: The Computational Beauty of Nature: Computer Explorations of Fractals, Chaos, Complex Systems and Adaptation. MIT Press, Cambridge MA, 4th edn. (2001).

Franchini, M.: Use of a Genetic Algorithm Combined with a Local Search Method for the Automatic Calibration of Conceptual Rainfall-runoff Models. Hydrological Sciences Journal. 41 (1996) 21–39.

Horn, J., Goldberg, D.E., Deb, K.: Implicit Niching in a Learning Classifier System: Nature's Way. Evolutionary Computation 2 (1994) 27–66.

Hsu, K.L., Gupta, H.V. and Sorooshian, S.: Artificial neural networks modeling of the rainfall-runoff process, Water Resources Research. 31 (1995) 2517–2530.

Kasabov N.K.: Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering. MIT Press, Cambridge, Massachusetts (1996).

Jain, A., Sudheer, K.P., Srinivasulu, S.: Identification of physical processes inherent in artificial neural network rainfall runoff models. Hydrological Processes 18 (2004) 571–581.

Jain, A., Srinivasulu, S.: Development of effective and efficient rainfall-runoff models using integration of deterministic, real-coded genetic algorithms and artificial neural network techniques. Water Resources Research W04302 (2004).

Moriarty, D.E., Miikkulainen, R.: Forming neural networks through efficient and adaptive coevolution. Evolutionary Computation. 5 (1998) 373–399.

Nash, J.E., Sutcliffe, J.V.: River flow forecasting through conceptual models, I, A discussion of principles. Journal of Hydrology 10 (1970) 282–290.

Potter, M.A.: The Design and Analysis of a Computational Model of Cooperative Coevolution. PhD Thesis, George Mason University (1997).

Rajurkar, M.P., Kothyari, U.C., Chaube, U.C.: Artificial neural networks for daily rainfall-runoff modeling. Hydrological Sciences Journal 47 (2002) 865–877.

Raman, H., Sunilkumar, N.: Multivariate modeling of water resources time series using artificial neural networks. Hydrological Sciences Journal 40(2) (1995) 145–163.

Reeves, C.R., Rowe, J.E.: Genetic Algorithms: Principles and Perspectives. Kluwer Academic Publishers, 1st edn. (2003).

Shamseldin, A.Y., O'Connor, K.M.: A non-linear neural network technique for updating of river flow forecasts. Hydrology and Earth System Sciences 5 (2001) 577–597.

Sudheer, K.P., Jain, A.: Explaining the internal behaviour of artificial neural network river flow models. Hydrological Processes 18 (2004) 833–844.

Wang, Q.J.: The genetic algorithm and its application to calibration of conceptual rainfall-runoff models. Water Resources Research 27 (1991) 2467–2471.

Wilby, R.L., Abrahart, R.J., Dawson, C.W.: Detection of conceptual model rainfall runoff processes inside an artificial neural network. Hydrological Sciences Journal 48(2) (2003) 163–181.

Whitley, D., Starkweather, T., Bogart, C.: Genetic algorithms and neural networks: Optimizing connections and connectivity. Parallel Computing 14 (1990) 347–361.

Zhang, B., Govindaraju, S.: Prediction of watershed runoff using Bayesian concepts and modular neural networks, Water Resources Research, 36(3) (2000) 753–762.

Part V
Emerging Technologies

Chapter 24
Combining Machine Learning and Domain Knowledge in Modular Modelling

D.P. Solomatine

Abstract Data-driven models based on the methods of machine learning have proven to be accurate tools in predicting various natural phenomena. Their accuracy, however, can be increased if several learning models are combined. A modular model is comprised of a set of specialized models each of which is responsible for particular sub-processes or situations, and may be trained on a subset of the training set. This paper presents the typology of such models and refers to a number of approaches to build them. An issue of combining machine learning with domain expert knowledge is discussed, and new approaches are presented.

Keywords Local models · modular models · committees · neural networks · flood forecasting

24.1 Introduction

Most natural phenomena are composed of a number of interacting sub-processes, so a model of a phenomenon should consist of several components, either process (physically-based) models, or data-driven. In the case of data-driven modelling the training examples can be divided into several groups, and separate specialised models built for each of them. Such partitioning can be performed using clustering techniques, or algorithms and rules based on domain expert knowledge. These models we will call *modular models* (MM), and their components will be called modules, or local models. Note that various authors use different terms to denote the combination of models (modular models): committee machines (Haykin, 1999), combining classifiers (Kuncheva, 2004), mixtures of experts (Jordan and Jacobs, 1995), multi-models, or gated networks. Another term, "data fusion", typically refers to the process of combining data from various sources (sensors) in order to arrive at a better understanding of the studied environment. This term is sometimes also used to denote the combination of information flows generated by various models, e.g. Abrahart and See (2002).

D.P. Solomatine
UNESCO-IHE Institute for Water Education, 2601DA, Delft, The Netherlands

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

In the context of hydroinformatics, several researchers have reported the usefulness of combining several models. Typically, however, such models are responsible for the whole process under question, an ensemble of the models is built, and their outputs are combined. See and Openshaw (2000) used a hybrid multi-model approach to river flow forecasting; they combined artificial neural networks (ANNs), fuzzy rule-based systems and ARMA models in an ensemble using several averaging and Bayesian methods. Xiong et al. (2001) used a non-linear combination of the forecasts of rainfall–runoff models using fuzzy logic. Abrahart and See (2002) used six alternative methods to combine data-driven and physically-based hydrologic models. Georgakakos et al. (2004) analysed the advantages of multi-model ensembles where each model is a hydrologic distributed model with the same structure but different parameters. Toth and Brath (2002), and Abebe and Price (2004) combined models in a different way, using ANNs for updating the forecasts made by a physically-based model.

Only very recently, however, has an explicitly modular approach to water-related modelling been undertaken. Among the latest publications in this area the following can be mentioned. Solomatine and Xue (2004) demonstrated an approach where separate data-driven catchment models were built for various hydrologic regimes (identified on the basis of hydrologic domain knowledge); these specialized models were also modular (M5 model trees) but the modules were generated using machine learning algorithms. Wang et al. (2006) used a combination of ANNs for forecasting daily streamflow: different networks were trained on the data subsets determined by applying either a threshold discharge value, or clustering in the space of inputs (several lagged discharges only but no rainfall data, however). Jain and Srinivasulu (2006) used a similar approach – decomposing the flow hydrograph by a certain threshold value and then building separate ANNs for each regime. Corzo and Solomatine (2006, 2007) used a more sophisticated approach by applying specialized algorithms for the hydrograph analysis to separate baseflow from the excess flow and then building a combination of ANN-based models which was optimized by a genetic algorithm. All studies demonstrated the higher accuracy of the modular models if compared to overall "global" models.

The increased attention to the multi-model approach in data-driven modelling calls for some sort of typology of such models and for developing better algorithms where domain knowledge can be taken into account. In this paper a typology of modular models is suggested, ways to optimize the process of building modular models is presented, and the problem of incorporating more domain knowledge into such models is addressed.

24.2 Methods of Combining Models

In the context of data-driven modelling, the model is calibrated (trained) on a particular set of examples, or instances (training data set). This means that the modules of MM have to be trained on particular subsets of this data set (possibly intersecting),

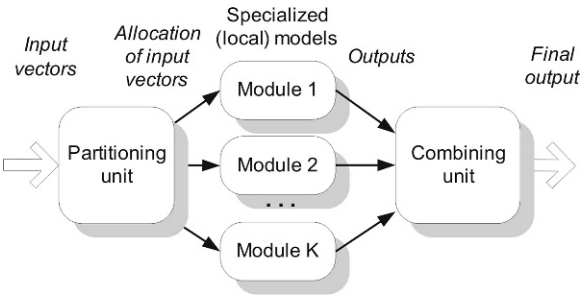


Fig. 24.1 Partitioning data sets and combining the outputs

and there are two major decisions to be made: (A) which module should receive which training example (partitioning problem), and (B) how the outputs of the modules should be combined to form the output of the final output of the system (combining problem). Accordingly, two decision units A and B must be built, or one unit performing both functions. Such a unit can be called an integrating unit, or in the case of using ANNs, the term *gating network* is used. Note that the functioning of units A and B could be different during training and operation. Figure 24.1 illustrates this principle. Note also that such an architecture can be made dynamic and adaptive, i.e. accommodating to the changing environment, e.g. by learning different switching strategies by unit A, or by its periodic retraining as new data are collected.

The following sections introduce five types of MMs with respect to the way that partitioning and combining is performed.

24.2.1 Hard Partitioning

The training set is partitioned into subsets (the input space can be partitioned accordingly), and for each of them an individual local expert model is trained. In operation, the input vector is processed only by one of the models and its output becomes the output of the overall model.

One way to do this is to use information theory to perform such splits and to perform splitting progressively; examples are: decision trees, regression trees (often referred to as CART – classification and regression tree), MARS (Breiman et al., 1984) and M5 model trees (Quinlan, 1992) (considered in Sect. 24.3).

Another method, used in time series analysis, is based on automatic identification of different dynamic regimes in the process, for example by applying hidden Markov models, and then building different expert models for each regime. An example of such an approach is reported by Velickov (2003) (note that the reported version of the method uses the weighted combination of expert models and in this sense belongs to the class of ensemble models as well.)

24.2.2 Hard Partitioning with a Soft Combination of Models (Fuzzy Committee)

Hard partitioning leads to a problem of compatibility at the boundaries between partitions. This issue calls for the introduction of smooth weighting schemes to combine outputs of the local expert models. One of the ways is of course to use the statistical approaches mentioned above. It is however possible to combine hard and soft partitioning by introducing a more transparent combining scheme.

This can be done by weighting the relevant model outputs for the input vectors that are close to the boundary. Such weighting can also be formulated with the use of fuzzy logic and the following framework of a *fuzzy committee* can be proposed (Fig. 24.2):

1. Perform hard partitioning of the training set into subsets and of the input space into regions.
2. Train individual local models for each subset.
3. For each example, assign the values of the membership functions (degree of belonging) corresponding to each model. The functions should be constant over the "central" part of the region, starting to decrease in the proximity of the region boundary, and decreasing to zero beyond the boundary; an option is to use a simple trapezoidal shape.
4. For the new input vector, calculate the output value as the combination of the outputs of all models weighted by the corresponding membership function values.

The weighted combination of global models is quite widely used, but the combination of local models is less common – see e.g., Kasabov and Song (2002). The presented framework would allow for combining the advantage of local modeling with the smooth combination of models at the boundaries between the regions. The shapes of the membership functions have to be optimized, for example by minimizing the overall model error.

Note that the introduced principle of the fuzzy committee approach is to address the problem of "fitting" the local models and in this respect it differs from the combination of classifiers based on the fuzzy integral (Kuncheva 2004).

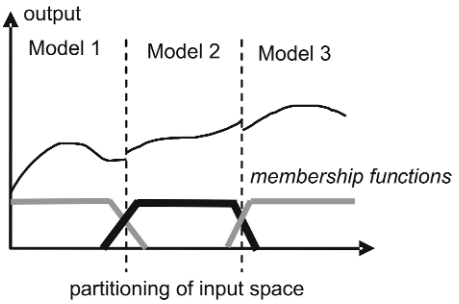


Fig. 24.2 MMs with hard partitioning have incompatibilities when switching between models. A solution is to assign models to membership functions and implement a fuzzy committee

24.2.3 Ensembles

MM becomes an *ensemble* model if the data are not partitioned at all. The individual models are trained on the whole training set and their outputs are combined using a weighting scheme where the model weights typically depend on model accuracy (see e.g., Kuncheva, 2004; Georgakakos et al., 2004).

24.2.4 Statistical Soft Partitioning

Statistically-driven approaches with "soft" splits of the input space are represented by *mixtures of experts* (Jacobs et al., 1991), *bagging* (Breiman, 1996) and *boosting* represented by a popular boosting algorithm *AdaBoost* (Freund and Schapire, 1997). A new version of AdaBoost for regression is *AdaBoost.RT* by Shrestha and Solomatine (2006) where other mentioned methods are introduced as well.

24.2.5 Instance-Based Learning

Instance-based learning is based on the combination of training examples and thus constructing a local approximation to the modelled function that applies well in the immediate neighbourhood of the new query instance encountered (k-NN, local weighted regression and other instance-based learning methods).

In the next section popular models based on hard partitioning are discussed in more detail, along with suggested methods of their improvement.

24.3 Popular Models Using Hard Partitioning: Regression and M5 Model Trees

24.3.1 Existing Algorithms

Regression and model trees represent an important class of machine learning modular models. They use the following idea: split the parameter space into areas (subspaces) and build a separate regression model of zero or first order for each one. If models in the leaves are of zero order (numeric constants) then this model is called a regression tree (Breiman et al., 1984); if the models are of first order (linear regression models) then the model is referred to as an M5 model tree (Quinlan 1992; "M5" stands for "Model trees, version 5"). Tree-based models are constructed by a divide-and-conquer method. The set T is either associated with a leaf, or some test is chosen that splits T into subsets corresponding to the test outcomes and the

same process is applied recursively to the subsets. The splitting criterion for the M5 model tree algorithm is based on treating the standard deviation of the output values that reach a node as a measure of the error at that node, and calculating the expected reduction in this error as a result of testing each attribute at that node.

In the case of numeric inputs the rules (boolean tests) a_i used to split the data set have the form "$x_i < C$" where i and C are chosen to minimize the standard deviation in the subsets resulting from the split. *Mn* are local specialised models built for subsets filtered down to a given tree leaf. In fact, the resulting model can be seen as a committee of linear models that are specialized on subsets of the training set, each of which belongs to a particular region of the input space (Fig. 24.3).

This idea is not new: a combination of specialized models ("local" models) is used in modelling quite often. One can find a clear analogy between model trees (MTs) and a combination of linear models already used in dynamic hydrology in the 1970s – see e.g., a paper on multi-linear models by Becker and Kundzewicz (1987). However, the M5 model tree approach based on the principle of information theory makes it possible to generate the models automatically according to the overall quality criterion; it also allows for varying the number of models.

Each leaf in a model tree represents a local model and in principle is (locally) more accurate than a global model (even a non-linear one, e.g. a neural network) trained on the whole data set. The linear regression method is based on an assumption of linear dependencies between input and output. In a M5 model tree a step

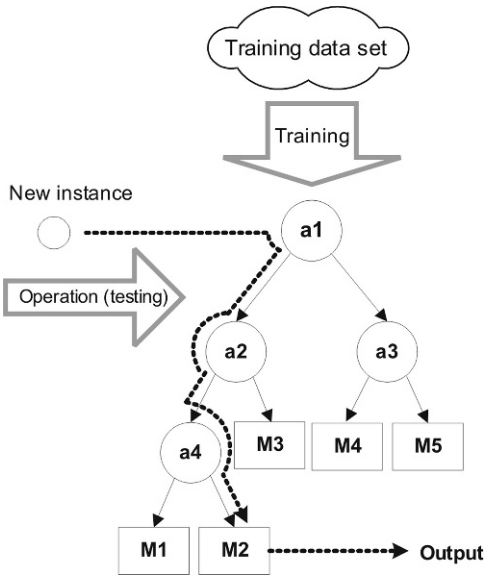


Fig. 24.3 Building a tree-like modular model. Rules a1.a4 are used to partition the data. Models M1.M5 are data-driven models (linear regression models in the case of an M5 model tree) built on subsets filtered down to this particular node

towards non-linearity is made – by building a model that is locally linear, but overall is non-linear. MTs may serve as an alternative to non-linear models like ANNs (which are *global* models) and are often almost as accurate as ANNs but have some important advantages:

- training of MTs is much faster than ANNs, and it always converges;
- the results can be easily understood by decision makers;
- by applying *pruning* (that is making trees smaller by combining sub-trees in one node) it is possible to generate a range of MTs – from an inaccurate but simple linear regression (one leaf only) to a much more accurate but complex combination of local models (many branches and leaves).

Probably the first application of M5 model trees in hydrologic forecasting was reported by Kompare et al. (1997). Solomatine and Dulal (2003) used M5 model trees in rainfall-runoff modelling of a catchment in Italy and found that its accuracy was at least as high as that of an ANN.

24.3.2 A Non-Greedy Approach to Building Model Trees (M5opt Algorithm)

Standard M5 adopts a greedy algorithm which constructs a model tree with a non-fixed structure by using a certain stopping criterion. M5 minimizes the error at each interior node, one node at a time. This process is started at the root and is repeated recursively until all or almost all of the instances are correctly classified. In constructing this initial tree M5 is greedy, and this can be improved. In principle, it is possible to build a fully non-greedy algorithm; however, the computational cost of such an approach would be too high. In the M5opt algorithm, introduced by Solomatine and Siek (2004a, 2006), a compromise of combining greedy and non-greedy approaches was adopted (Fig. 24.4).

M5opt enables the user to define the level of the tree up to which the non-greedy algorithm is applied, starting from the root. If a full exhaustive search is employed at this stage, all tree structures and all possible attributes and split values are tried; an alternative is to employ a randomized search, for example, a genetic algorithm. The levels below are then constructed using the greedy M5 algorithm. This principle still complies with the way that the terms of linear models at the leaves of the model tree are obtained from split attributes of the interior nodes below these leaves before the pruning process. M5opt has a number of other attractive additional features: *initial approximation* (M5 builds the initial model tree in a way similar to regression trees where the split is performed based on the averaged output values of the instances that reach a node; the M5opt algorithm builds linear models directly in the initial model tree); and *compacting the tree* (improvement to the pruning method of M5). Examples of using this method for building hydrologic models are given in Solomatine and Siek (2006).

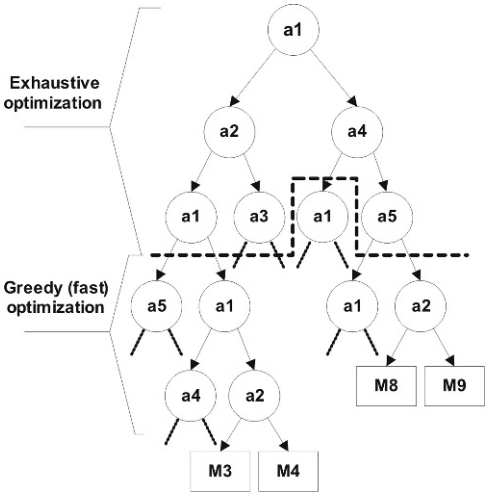


Fig. 24.4 M5opt algorithm: combination of exhaustive and greedy optimization of a model tree

24.4 Complementary Models

In contrast to the modular models where each sub-model is focussing on modelling certain sub-process, the so-called complementary models are the combinations of models that do not model the same process but instead complement or work with each other. For example, a DDM may be used to correct errors in a physically-based model or alternatively errors in another DDM. This type of approach has been employed by Shamseldin and O'Connor (2001) in which ANNs were used to update runoff forecasts; the simulated flows from a model and the current and previously observed flows were used as input, and the corresponding observed flow as the target output. Updates of daily flow forecasts for a lead-time of up to four days were made. It was reported that ANN models gave more accurate improvements than autoregressive models. Lekkas et al. (2001) showed that error forecasting provides improved real-time flow forecasting, especially when the forecasting model is poor. Abebe and Price (2004) used this approach to correct the errors of a routing model of the River Wye in the UK by an ANN. Solomatine et al. (2007) built an ANN-based rainfall-runoff model where its outputs were corrected by an instance-based model.

24.5 Domain Knowledge in Building a Modular Model

24.5.1 Degrees of Involving a Domain Expert

An important issue in modelling is the optimal use of domain knowledge, and modular modelling is not an exception. Models are devices that have to be built with

the active participation of domain experts and the latter typically try to introduce as much domain knowledge into the models as possible. Experts typically are expected to ensure that the data are right, to specify the "hidden" processes, to choose model types, to ensure modeling is an iterative process, and of course to interpret the results. Machines (machine learning) are expected to build models on the properly selected data sets. Ideally, humans and machines should constitute "modelling hybrids". (Here we assume that humans are the carriers of domain knowledge.)

This is also true for modular models when the expert input (domain knowledge) is expected not only in determining the model type, but also, possibly, in the way the data are partitioned.

A machine learning algorithm minimizes the training (cross-validation) error considering it as the ultimate indicator of the algorithms performance, so is purely data-driven. Domain experts, however, may have additional considerations in assessing the model skill, and want to have certain control over the decisions (A) and (B) in Fig. 24.1 and over the choice of models used in each unit. The challenge is to integrate the background domain knowledge into a machine learning algorithm by allowing the user to determine some important structural properties of the model based on physical insight, and leaving more tedious tasks to machine learning. Fig. 24.5 shows several levels of involvement of a human expert in building MMs. One possibility, fully machine-driven (like the M5 or M5opt algorithms) has been already considered. The other two possibilities are discussed below.

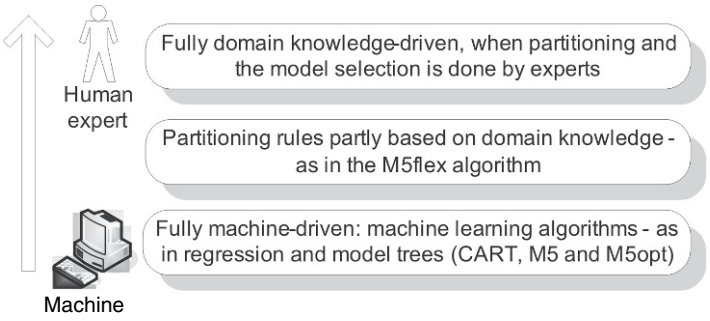


Fig. 24.5 Methods of building a modular model, with various levels of involving a human domain expert

24.5.2 Partitioning Rules Partly Based on Domain Knowledge

For various machine learning algorithms introduction of a domain expert may have different forms. For example, for building linear models with hard partitioning, a version of the M5 algorithm, *M5flex*, has been recently developed by Solomatine and Siek (2004b). In this version, a domain expert is introduced by making a decision about the splits at important nodes (Fig. 24.6). This method enables the user to determine split attributes and values in some important (i.e. top-most) nodes, and

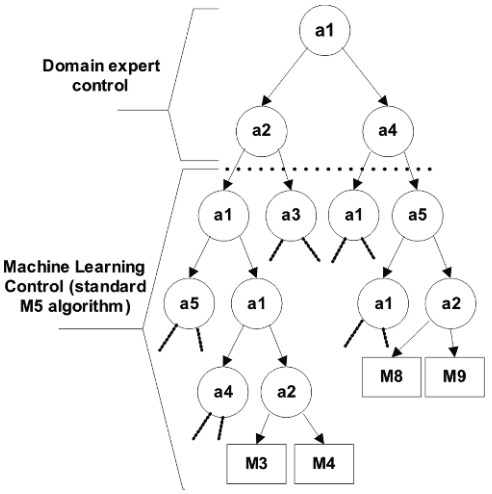


Fig. 24.6 M5flex algorithm: more control is given to a domain expert in the splitting process at important nodes

then the M5 machine learning algorithm takes care of the remainder of the model tree building. In the context of flood prediction, for example, the expert user can instruct the M5flex to separate the low flow and high flow conditions to be modelled separately. Such models can be more suitable for hydrological applications than ANNs or standard M5 model trees. The inclusion of a domain expert makes it possible to make the models more realistic and to increase their accuracy. The reader is referred to the examples in Solomatine and Siek (2004b).

24.5.3 Models Fully Based on Domain Knowledge for Partitioning and Model Selection

It is also possible to allow a domain expert to construct the rules performing the hard partitioning of the training set, and to select the model types. Note that the models (modules) may not necessarily be only data-driven, and may include expert judgement. If an overall model uses various types of models, it can be called a *hybrid model*.

Xiong et al. (2001) combined several physically-based forecasting models with the help of a fuzzy system. Solomatine et al. (2007) built a committee on the basis of several types of data-driven models including instance-based models, M5 model trees and neural networks.

In a study by Solomatine and Xue (2004) the flow predictions in the Huai river basin (China) were made on the basis of previous flows and precipitation, and a committee hybrid model was built. The problem was to predict Q_{t+1} flow one day ahead. The following notations are used: flows on the previous and current day as Q_{t-1} and Q_t respectively; precipitation on the previous day as P_{t-1} ; moving average (2-days) of the precipitation two days before as $Pmov2_{t-2}$; moving average (3-days) precipitation four days before as $Pmov3_{t-4}$.

As a first step a domain expert was asked to identify several hydrological conditions (rules) used to split the input space into three regions; they are given below:

1. $Q_{t-1} = 1000\,\mathrm{m}^3/\mathrm{s}$ (high flows)
2. $Q_{t-1} < 1000\,\mathrm{m}^3/\mathrm{s}$ AND $Q_t = 200\,\mathrm{m}^3/\mathrm{s}$ (medium flows)
3. $P_{t-1} > 50$ AND $Pmov2_{t-2} < 5$ AND $Pmov3_{t-4} < 5$ (flood condition due to the short but intensive rainfall after a period of dry weather).

For each of these regions separate local models were built (M5 model trees and ANNs). Note that the model tree is also a modular model, so here a combination of domain knowledge data partitioning is complemented by the machine learning-based partitioning. This approach combined the best features of both, and seems to be very promising.

Yet another example of building a modular model where domain knowledge is explicitly used is the one considered by Corzo and Solomatine (2006, 2007); in it, various methods of baseflow separation are applied and then separate models are built for different flow components.

24.6 Conclusions

Data-driven modelling and computational intelligence methods have proven their applicability in modelling various water-related processes considered in the context of hydroinformatics. Since such processes are typically very complex, the modular approach to their modelling allows more accurate local modelling of sub-processes to be performed. The studies conducted recently demonstrate the effectiveness of such an approach.

The area of modular modelling has great promise also due to the fact that modular models allow for better incorporation of domain knowledge into the modelling process. In newly developed algorithms (e.g., M5flex) an expert has the possibility to intervene at important stages of model building, and such combination of machine learning with domain knowledge not only improves the model quality but also their acceptance by decision makers.

A future is seen in using *hybrid* models to combine models of different types and following different modelling paradigms, including combination with physically-based models. A challenge here is to build optimal and adaptive model structures with the adequate involvement of domain experts.

References

Abebe AJ, Price RK (2004). Information theory and neural networks for managing uncertainty in flood routing. ASCE Journal of Computing in Civil Engineering 18(4): 373–380.

Abrahart RJ, See L (2002). Multi-model data fusion for river flow forecasting: an evaluation of six alternative methods based on two contrasting catchments. Hydrology and Earth System Sciences 6(4): 655–670.

Becker A, Kundzewicz ZW (1987) Nonlinear flood routing with multilinear models. Water Resources Research 23: 1043–1048.

Breiman L, Friedman JH, Olshen RA, Stone CJ (1984). Classification and regression trees. Wadsworth International: Belmont.

Breiman L (1996). Stacked regressor. Machine Learning, 24(1): 49–64.

Corzo GA, Solomatine DP (2006) Multi-objective optimization of ANN hybrid committees based on hydrological knowledge. Geophysical Research Abstracts, 8: 09794. EGU General Assembly, Vienna, April 2006.

Corzo GA, Solomatine DP (2007). Baseflow separation techniques for modular artificial neural network modelling in flow forecasting. Hydrological Sciences Journal 52(3), 491–507.

Freund Y, Schapire R (1997). A decision-theoretic generalisation of on-line learning and an application of boosting. Journal of Computer and System Science 55(1): 119–139.

Georgakakos KP, Seo D-J, Gupta H, Schaake J, Butts MB (2004) Towards the characterization of streamflow simulation uncertainty through multimodel ensembles. Journal of Hydrology 298(1): 222–241.

Haykin S (1999) Neural networks: a comprehensive foundation. McMillan, New York.

Jacobs RA, Jordan MI, Nowlan SJ, Hinton GE (1991) Adaptative mixtures of local experts. Neural Computation 3: 79–87.

Jordan MI, Jacobs RA (1995) Modular and hierarchical learning systems. The Handbook of Brain Theory and Neural Networks, Arbib, M. (Ed.). MIT Press: Cambridge.

Jain A, Srinivasulu S (2006) Integrated approach to model decomposed flow hydrograph using artificial neural network and conceptual techniques. Journal of Hydrology 317: 291–306.

Jordan MI, Jacobs RA (1995). Modular and hierarchical learning systems. The Handbook of Brain Theory and Neural Networks, Arbib, M. (Ed.). MIT Press: Cambridge.

Kasabov NK, Song Q (2002). DENFIS: Dynamic Evolving Neural-Fuzzy Inference System and Its Application for Time Series Prediction. IEEE Trans. Fuzzy Systems 2: 144–154.

Kompare B, Steinman F, Cerar U, Dzeroski S (1997) Prediction of rainfall runoff from catchment by intelligent data analysis with machine learning tools within the artificial intelligence tools. Acta Hydrotechnica (in Slovene) 16/17, 79–94.

Kuncheva LI (2004) Combining Pattern Classifiers. Wiley, NJ.

Lekkas DF, Imrie CE, Lees MJ (2001) Improved non-linear transfer function and neural network methods of flow routing for real-time forecasting, Journal of Hydroinformatics 3(3): 153–164.

Quinlan JR (1992) Learning with continuous classes. Proc. AI'92, 5th Australian Joint Conference on Artificial Intelligence, Adams, A. and Sterling, L. (eds.), World Scientific: Singapore, 343–348.

See L, Openshaw S (2000) A hybrid multi-model approach to river level forecasting. Hydrological Sciences Journal 45: 523–536.

Shamseldin AY, O'Connor KM (2001) A non-linear neural network technique for updating of river flow forecasts. Hydrology and Earth System Sciences 5(4): 577–597.

Shrestha DL, Solomatine DP (2006) Experiments with AdaBoost.RT, an Improved Boosting Scheme for Regression. Neural Computation 17: 1678–1710.

Solomatine DP, Dulal KN (2003) Model tree as an alternative to neural network in rainfall-runoff modelling. Hydrological Sciences Journal 48(3): 399–411.

Solomatine DP, Siek MB (2004a) Semi-optimal Hierarchical Regression Models and ANNs. Proc. Intern. Joint Conference on Neural Networks, Budapest, Hungary, July 2004, 1173–1177.

Solomatine DP, Siek MB (2004b) Flexible and optimal M5 model trees with applications to flow predictions. Proc. 6th Int. Conference on Hydroinformatics, World Scientific: Singapore.

Solomatine DP, Siek MB (2006) Modular learning models in forecasting natural phenomena. Neural Networks 19: 215–224.

Solomatine DP, Xue Y (2004) M5 model trees and neural networks: application to flood forecasting in the upper reach of the Huai River in China. ASCE J. Hydrologic Engineering 9(6): 491–501.

Solomatine DP, Maskey M, Shrestha DL (2007) Instance-based learning compared to other data-driven methods in hydrologic forecasting. Hydrological Processes, 21 (DOI:10.1002/hyp.6592).

Toth E, Brath A (2002) Flood forecasting using artificial neural networks in black-box and conceptual rainfall-runoff modelling. Proc. of the International Environmental Modelling and Software Society (iEMSs) Meeting. Web: http://www.iemss.org/iemss2002/proceedings/vol2.html.

Xiong LH, Shamseldin AY, O'Connor KM (2001) A non-linear combination of the forecasts of rainfall–runoff models by the first-order Takagi-Sugeno fuzzy system. Journal of Hydrology 245(1–4): 196–217.

Velickov S (2003) Mixture of models: a new framework for modelling complex nonlinear dynamical systems. Proc. XXX IAHR Congress. Thessaloniki, Vol. D, August 2003, 123–130.

Wang W, Van Gelder PHAJM, Vrijling JK, Ma J (2006) Forecasting daily streamflow using hybrid ANN models. Journal of Hydrology 324(1–4): 383–399.

Chapter 25
Precipitation Interception Modelling Using Machine Learning Methods – The Dragonja River Basin Case Study

L. Stravs, M. Brilly and M. Sraj

Abstract The machine learning methods M5 for generating regression and model tree models and J4.8 for generating classification tree models were selected as the methods for analysis of the results of experimental measurements in the Dragonja River basin. Many interesting and useful details about the process of precipitation interception by the forest in the Dragonja River basin were found. The resulting classification and regression tree models clearly show the degree of influence and interactions between different climatic factors, which importantly influence the process of precipitation interception.

Keywords Precipitation interception · forest hydrological cycle · the Dragonja River basin · machine learning · decision trees · M5 method · J4.8 method

25.1 Introduction

Hydrological science studies the circulation of water in nature, its phenomena, distribution on the earth, movement and physical-chemical characteristics (Chow, 1964). It mainly deals with circulation of water between the atmosphere, surface of the earth and its water systems (Brilly and Sraj, 2000). Forest hydrology studies the circulation of water in forested areas. It studies the course and ways of transition of water from the atmosphere through the forest ecosystem into the ground, groundwater and surface waters and its return back to the atmosphere (Smolej, 1988).

Precipitation is the main source of water in the hydrological cycle. Mostly, it is represented by rain and snow; however, in the coastline and in mountainous,

L. Stravs
Faculty of Civil and Geodetic Engineering, University of Ljubljana, Jamova 2, SI-1000 Ljubljana, Slovenia, e-mail: lstravs@fgg.uni-lj.si

M. Brilly
Faculty of Civil and Geodetic Engineering, University of Ljubljana, Jamova 2, SI-1000 Ljubljana, Slovenia, e-mail: mbrilly@fgg.uni-lj.si

M. Sraj
Faculty of Civil and Geodetic Engineering, University of Ljubljana, Jamova 2, SI-1000 Ljubljana, Slovenia

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

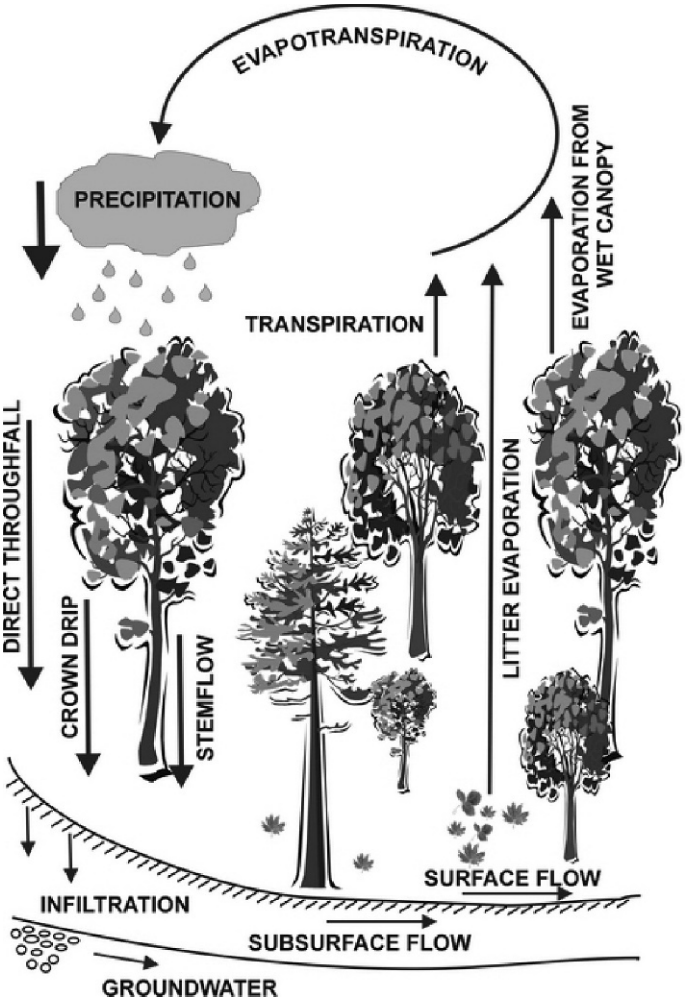


Fig. 25.1 Components of the forest hydrological cycle (Sraj, 2003a)

forested areas horizontal precipitation occurs, i.e. fog. In forested areas (Fig. 25.1) precipitation may be intercepted by the forest canopy and returned to the atmosphere via evaporation, or channeled downwards via throughfall, which is a portion of the rainfall that falls directly through gaps in the canopy or arrives on the ground as crown drip, or stemflow, which is the process that directs a portion of rainfall down tree branches and stems. Precipitation intercepted by the forest canopy (Sraj, 2003b) can be expressed as:

$$E_i = P - (T_f + S_f) \tag{25.1}$$

where:

E_i ... precipitation interception;
P ... total precipitation amount above the forest canopy;

T_f … throughfall (sum of direct throughfall and crown drip);
S_f … stemflow.

Amount of intercepted precipitation (Mikos et al., 2002) depends on vegetation and climatic factors:

- canopy capacity, which depends upon the class of species, size, shape and vegetational age, area and leaf orientation (coniferous trees intercept 20–40%, and deciduous trees 20–25% precipitation; the higher the vegetation age, the higher the intercepted precipitation (Geiger et al., 1995)).
- vegetational density (interception increases with tree density).
- intensity, duration and frequency of precipitation (smaller intensity or short duration results in a higher evaporation rate from the canopy, intensity of evaporation rate is highest at the beginning of storms, frequently occurring precipitation reduces interception).
- precipitation type (with coniferous species the water equivalent of intercepted snow exceeds the value of intercepted liquid precipitation).
- climate conditions (higher temperatures cause higher evaporation rate, the wind can have high influence on evaporation).
- periods in the course of the year (growing period, dormant period).

Based upon research, Ovington (1954) concluded that the quantity of intercepted precipitation may vary between 6 and 93%, i.e. in different conditions a very different interception rate may be achieved. The two most widely used modelling methods to estimate precipitation interception losses are process-based models of interception and evaporative loss, and empirical or semi-empirical regression models. In the field of precipitation interception modelling Rutter et al. (1971) were the first to move away from a site-specific empirical regression approach to estimate the interception loss. Rutter's model is a numerical model based on the water balance of the canopy and trunks and requires extensive climatic and canopy drainage data. The change in amount of water stored in the canopy is determined by the proportion of the rain that hits the canopy, the drainage from the canopy and evaporation of intercepted water (Schellekens et al., 1999). Gash (1979) proposed a simpler analytical model of precipitation interception based on Rutter's numerical model, in which he considered rainfall to occur as a series of discrete events and assumed for the canopy to have sufficient time to dry between events. Gash's model requires prior estimation of the canopy structure parameters.

In cooperation with the Vrije Universiteit, Amsterdam, extensive research of the hydrological processes in the Dragonja River basin was performed. The Dragonja River basin was chosen as an experimental river basin because intense natural reforestation has been identified in the last decades. This has caused a decrease in minimal and maximal flows of the Dragonja River, while at the same time no noticeable precipitation and temperature regime changes have been identified. The main intention of the research was to analyse the impact of reforestation on the water balance of the entire river basin and to determine the importance of individual climate factors influencing it.

Experimental equipment for measurements of the individual components of the forest hydrological cycle was set up. The machine learning methods M5 for generating regression and model tree models and J4.8 for generating classification tree models were selected as the methods for analysis of the results of experimental measurements in the Dragonja River basin. Successful applications of the machine learning techniques in modelling of hydrological processes like floods, debris flows and other water-related processes are well known (Stravs et al., 2004; Solomatine and Dulal, 2003). While usage of neural networks has already been widely researched and explored in the field of hydrological science (Govindaraju and Ramachandra Rao, 2000), new emerging methods from the framework of artificial intelligence like decision trees, instance-based learning, fuzzy based systems (Stuber and Gemmar, 1997), chaos theory and others have not gained much attention in the field of hydrology yet. Generally, machine learning methods are used for generating forecasting models or for generating descriptive models from which new knowledge about the modelled process can be learned.

25.2 River Basin Characteristics

The Dragonja River basin (Fig. 25.2) with a drainage area of 90.5 km^2 is situated in the southwest of Slovenia, on the Northern part of the Istria Peninsula. It flows from East to West to the North Adriatic Sea (the Piran Bay). On its mouth, there is a RAMSAR protected wetland (Secovlje salt pans), to which also the rivers Drnica and Jernejski potok flow. The Drnica used to be the tributary of the Dragonja, but after regulation of the Dragonja at its outflow to the sea, they became separated. The

Fig. 25.2 The Dragonja River basin

surface runoff area of the Secovlje salt pans is 142 km^2, extending over Slovenia (116 km^2) and Croatia, the bordering state. Close to the Dragonja outflow to the sea, there are two karstic springs (Buzini and Gabrijeli) with a karstic river basin area. It mostly extends to Croatia (73 km^2 large area). Taking into consideration all river basin sub-units of the Secovlje salt pans, the area has 215 km^2 (Fig. 25.2).

The river basin consists of long flat ridges (up to 400 m a.s.l.) above the deep and narrow river valleys, where the majority of settlements have developed. There are 5860 inhabitants; depopulation has been noticed from 1960 on, but in the late 1990s it stopped (Globevnik, 2001). The area has rural character. The land, typically owned by one family, is traditionally very small; therefore there are hardly any large hillside farms. Larger farms can be found in the river valley at its outflow to the sea. Today, a few new plantation areas on the hills have developed (vineyards, olive groves).

The average annual temperature is 14°C on the coast and 10°C on the continental side. The average yearly precipitation on the sea coast is 900 mm, whereas on the eastern side of the river basin it is 1200 mm. The hydrological characteristics of the Podkastel water station (87 km^2) are (Globevnik, 2001):

- annual mean flow (1971–1995): 1.16 m^3/s;
- autumn high water peaks: 98 m^3/s.

The Slovenian coastal area is well known for its water supply shortages, especially in the summer season when the hydrological conditions are usually quite critical.

25.3 Methods

25.3.1 Measurements

Two forest plots (400 m apart and both at around 200 m a.s.l.) in the 30–35 year-old forest above the confluence of the Dragonja River and the Rokava River were selected as areas where thorough experimental measurements of individual components of the forest hydrological cycle (Fig. 25.1) would be performed; the first plot (1420 m^2) was on the north facing slope in the Rokava River basin and the second one (615 m^2) on the south facing slope in the Dragonja River basin.

Precipitation above the canopy, throughfall and stemflow were measured on both research plots. Rainfall above the canopy was measured with a tipping bucket rain gauge and with a totalizator (manual gauge) for control (Fig. 25.3). Throughfall was measured with two steel gutters in combination with ten manual gauges, which were emptied and moved randomly (Fig. 25.3). Stemflow was measured on two of the most typical species in each plot: on the north plot on oak and hornbeam trees and on the south plot on ash and oak trees.

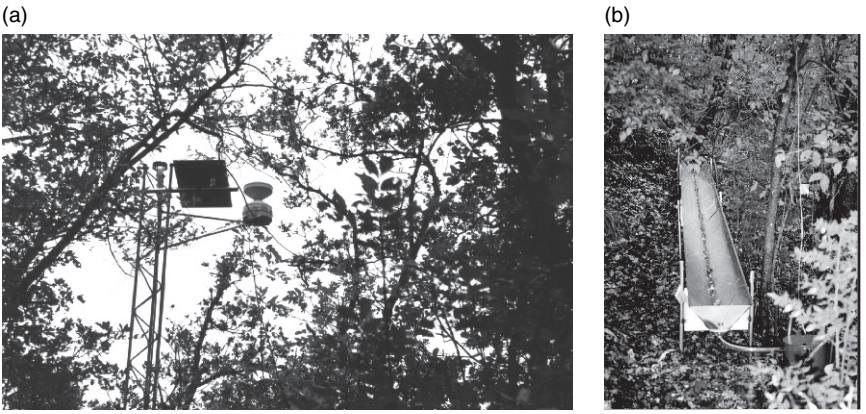


Fig. 25.3 Measurements of precipitation above the canopy (*left*) and throughfall (*right*)

All quantities were measured automatically with a 10-minute log time. Air temperature and relative humidity, wind direction and average wind speed were also measured at the nearby meteorological stations Kubed and Borst.

25.3.2 Modelling Methods

Machine learning generated models are mostly used for forecasting or prediction and for extracting new knowledge about the observed processes. In our case we used the machine learning methods M5 and J4.8 as they are implemented in the WEKA system (Witten and Frank, 2000), developed at the University of Waikato, New Zealand, to generate classification and regression tree models to analyse the impact of reforestation on the water balance of the entire river basin and learn more about the climate and other factors influencing it.

The basic idea of generating tree-like models is to develop simple, transparent models that are easy to use and interpret. The reason behind the choice of the decision trees for modeling the hydrological process of precipitation interception is obvious – we needed the result in the form and structure that can be easily interpreted and the resulting model that can uncover the empirically derived patterns of the underlying process.

By feeding the machine learning method with enough relevant input and output data of the modelled process it can automatically learn the patterns underlying the modelled process from the data only and it can divide the input data space (in machine learning theory called attributes) into subspaces where certain characteristic similarities or patterns exist.

Decision trees are generated through an iterative splitting of data into subspaces of the whole attribute space, where the goal is to maximize the distance between groups at each split (Breiman et al., 1984; Quinlan, 1986, 1992; Kompare, 1995;

Mitchell, 1997; Witten and Frank, 2000; Solomatine and Dulal, 2003). Basic components of a decision tree are the decision nodes, branches and leaves. The decision process starts with the top decision node (also called root node), which specifies a test to be carried out. The answer to this root node test causes the tree to split into branches, each representing one of the possible answers. Each branch will lead either to another subsequent decision node or to the bottom of the decision tree, called a leaf node.

Results of the modelling are decision tree models, which are a way of representing a series of rules that lead to a class value, numerical value or linear equation, and are therefore classified into:

- classification trees with class values as leaves of the model;
- regression trees with constant numerical values as leaves of the model;
- model trees with linear equations as leaves of the model.

25.3.3 Data

In the period of one year 369 events were recorded out of which 173 were recorded on the south plot and 196 on the north research plot. Events were separated by the rainless periods in which canopies could dry up. For each event the following attributes were available:

- plot orientation (North, South);
- rainfall quantity (expressed in mm);
- rainfall duration (hours);
- rainfall intensity (mm per hour);
- average air temperature (°C);
- relative humidity (%); and
- average wind speed (metres per second).

Precipitation above the canopy for single events varied from 0.2 to 100.2 mm, duration of rainfall varied from 5 minutes to almost 40 hours and rainfall intensity varied from 0.15 to 44 mm/h.

25.4 Results

Three decision tree models connecting some of the measured factors influencing the precipitation interception process in the Dragonja River basin and precipitation interception rate were developed.

In case #1 a classification tree (Fig. 25.4) was generated where attributes of each event were: plot orientation, rainfall quantity, duration and intensity, air temperature and relative humidity, and average wind speed. The output data or the modelled variable was precipitation interception percentage (relative to the precipitation amount

above the canopy for each event) classified into 7 classes; R_0 meaning 0% interception loss, R_1_20, R_21_40, R_41_60, R_61_80, R_81_99 and R_100 meaning 100% interception loss.

From the resulting model (Fig. 25.4) we can learn that for events with less than 2.4 mm of rainfall and with duration shorter than 10 minutes 100% of the precipitation (class value R_100 on Fig. 25.4) is intercepted. This means that under such conditions no groundwater recharge or surface or subsurface runoff occurs. For events with less than 2.4 mm of rainfall, duration longer than 10 minutes and average temperature of the event less than 14°C, then approximately 50% of the precipitation above the canopy is intercepted. If the average temperature of an event with less than 2.4 mm of rainfall and duration longer than 10 minutes is higher than 14°C, then once again, almost all of the precipitation is intercepted (class value R_81_99 – from 81 to 99%). For events with rainfall amount ranging from 2.4 to 7.0 mm approximately half of the rainfall is intercepted (class value R_41_60 – from 41 to 60%). At events with more than 7 mm of rainfall, average wind power and rainfall intensity also influence the precipitation interception by the forest canopy. It is interesting to note that the generated model does not distinguish the differences in the process of precipitation interception between north and south research plots, which was expected. This could also be a result of different climatic conditions recorded

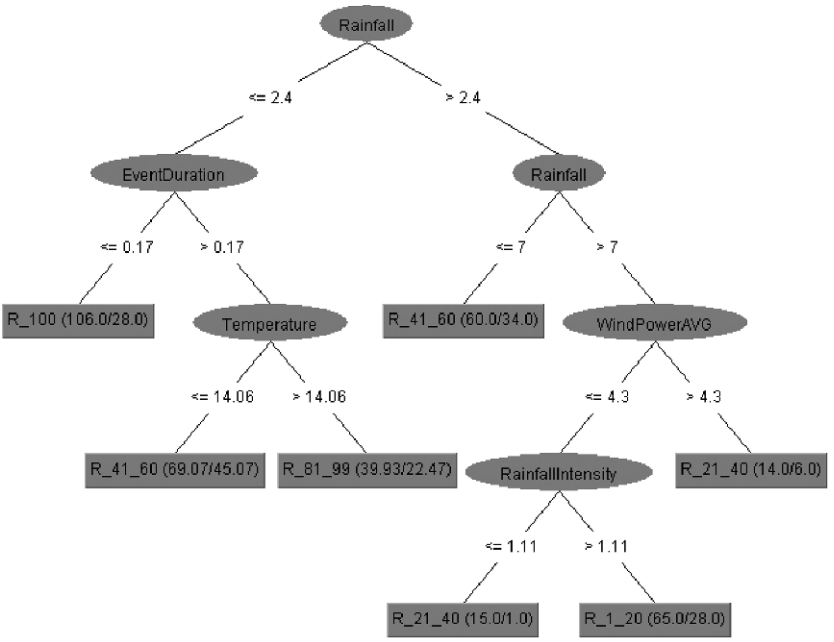


Fig. 25.4 Generated classification tree for case #1 (J4.8 method) – numbers in brackets of each leaf of the classification tree following the class value mean the number of instances that reached the leaf (*left*) and the number of incorrectly classified instances (*right*) in the model evaluation process

on each of the research plots resulting in different values of average air temperature, relative humidity and wind speed actually describing different climate conditions for each of the research plots.

The resulting classification tree (Fig. 25.4) correctly classifies 56% of the instances if it is evaluated on the training set and correctly classifies 48% of the instances in the process of 10-fold cross validation. More accurate classification trees were also generated, which correctly classify up to 65% of the instances if they are evaluated on the training set and correctly classify up to 55% of the instances in the process of 10-fold cross validation. However, they were pruned so that higher structural and explanatory transparency of the resulting models was achieved.

In cases #2 and #3 a regression tree (Fig. 25.5) was generated where attributes of each event were: plot orientation, rainfall quantity, duration and intensity, air temperature and humidity, and average wind speed. The class was the precipitation interception percentage (relative to the precipitation amount above the canopy for each event), this time in the form of a numerical value ranging from 0 to 100%. The difference between the resulting regression trees for cases #2 and #3 is in the complexity of the resulting regression tree (Figs. 25.5 and 25.6).

From the resulting models, especially from the pruned regression tree of case #3, we can learn that for events with less than 2.5 mm of rainfall and air temperature lower than 14.2°C, 81.2% of the rainfall is intercepted by the forest canopy if the event is shorter than 1.67 hours, and 47.2% of rainfall is intercepted if the event is longer than 1.67 hours. But if the temperature of the event with less than 2.5 mm of rainfall is higher than 14.2°C almost all precipitation is intercepted (95.5%). For events with more than 2.5 and less than 7.5 mm of rainfall, 42.8% of precipitation

```
Rainfall <= 2.5 :
|   Temperature <= 14.2 :
|   |   EventDuration <= 1.67 : PI_percent = 81.2
|   |   EventDuration >  1.67 : PI_percent = 47.2
|   Temperature >  14.2 :
|   |   Rainfall <= 0.5 : PI_percent = 98
|   |   Rainfall >  0.5 : PI_percent = 85.7
Rainfall >  2.5 :
|   Rainfall <= 7.5 : PI_percent = 42.8
|   Rainfall >  7.5 :
|   |   WindPowerAVG <= 3.45 :
|   |   |   RainfallIntensity <= 1.13 : PI_percent = 30.1
|   |   |   RainfallIntensity >  1.13 : PI_percent = 17
|   |   WindPowerAVG >  3.45 :
|   |   |   EventDuration <= 6.5 : PI_percent = 15.6
|   |   |   EventDuration >  6.5 : PI_percent = 40
```

Fig. 25.5 Generated regression tree (M5 method) – more complex (case #2) regression tree

```
Rainfall <= 2.5 :
|   Temperature <= 14.2 :
|   |   EventDuration <= 1.67 : PI_percent = 81.2
|   |   EventDuration >  1.67 : PI_percent = 47.2
|   Temperature >  14.2 : PI_percent = 95.5
Rainfall >  2.5 :
|   Rainfall <= 7.5 : PI_percent = 42.8
|   Rainfall >  7.5 : PI_percent = 23.2
```

Fig. 25.6 Generated regression tree (M5 method) – pruned regression tree (case #3)

is intercepted, and for events with rainfall amount higher than 7.5 mm 23.2% of rainfall is intercepted. The obtained regression tree of case #2 unveils some additional specific details for events with more than 7.5 mm of rainfall when wind speed becomes an important factor. If the average wind speed for such events is less than 3.5 m/s, approximately 10% less water is intercepted compared to events with average wind speed higher than 3.5 m/s.

25.5 Conclusions

Many interesting and useful details about the process of precipitation interception by the forest in the Dragonja River basin were found. In most cases, rainfall events with up to approximately 2.5 mm of rainfall contribute almost nothing to the recharge of groundwater at the Dragonja river basin and the Dragonja river discharge, with the exception of rainfall events longer than 1.67 hours and temperature lower than 14°C. The generated models also show that approximately 23 to 43% of the water at events with more than 2.5 mm of rainfall is intercepted by the forest as a direct consequence of natural reforestation in the last few decades.

The classification and regression tree models clearly show the degree of influence and interactions between different climatic factors, which importantly influence the process of precipitation interception. If the results obtained on both research plots are representative enough for the whole river basin where the process of reforestation occurred in the last decades, we can conclude that the impact of the land use change on the water balance of the Dragonja River basin is quite significant. In terms of water supply approximately one third of the water is lost in the river basin areas, which are covered by the forest.

The generated models captured the important properties of the processes of the forest hydrological cycle at the Dragonja River basin. Results were in the context of what was expected and known about the precipitation interception process. Furthermore, many significant details about the process in this particular river basin were uncovered in a really short modelling time. We can conclude that the usage

of machine learning methods for generating descriptive models like decision trees reduces the manpower and time spent in the process of extracting new knowledge about the processes that were measured and studied.

Usage of machine learning methods like decision trees for generation of structurally transparent and explanatory models from the data has offered great promise in helping scientists to uncover patterns hidden in their data. However, the development of models is only one of the steps in the acquisition of new knowledge; the selection, collection and preparation of data, the guidance of the model development and the interpretation of the generated models by the scientists who understand the modelled processes are equally important.

References

Breiman L, Friedman JH, Olshen RA, Stone CJ (1984) Classification and regression trees. Wadworth, Belmont

Brilly M, Sraj M (2000) Osnove hidrologije (Principles of Hydrology). University Textbook, University of Ljubljana, Faculty of Civil and Geodetic Engineering (in Slovene)

Chow VT (1964) Handbook of applied hydrology. McGraw-Hill, New York

Gash, JHC (1979) An analytical model of rainfall interception by forests. Quarterly Journal of the Royal Meteorological Society 105: 43–55.

Geiger R, Aron RH, Todhunter P (1995) The climate near the ground. Friedr. Vieweg & Sohn, Braunschweig/Wiesbaden

Globevnik L (2001) Celosten pristop k urejanju voda v povodjih (Intergrated approach to water resources management on the catchment level). Ph. D. Thesis, University of Ljubljana (in Slovene)

Govindaraju RS, Ramachandra Rao A (2000) Artificial neural networks in hydrology. Kluwer Academic Publishers, Dordrecht, Netherlands

Kompare B (1995) The use of artificial intelligence in ecological modelling. Ph. D. Thesis, Royal Danish School of Pharmacy, Copenhagen, Denmark

Mikos M, Kranjc A, Maticic B, Müller J, Rakovec J, Ros M, Brilly M (2002). Hidrolosko izrazje – Terminology in hydrology. Acta hydrotechnica 20/32: 3–324

Mitchell T (1997). Machine learning. MIT Press and The McGraw-Hill Companies, Inc

Ovington JD (1954) A comparison of rainfall in different woodlands. Forestry London 27, pp 41–53

Quinlan JR (1986) Induction of Decision Trees. Machine Learning 1: 81–106

Quinlan JR (1992) Learning with continuous classes. In: Proceedings of the Fifth Australian Joint Conference on Artificial Intelligence, pp 343–348

Rutter AJ, Kershaw KA, Robins PC, Morton AJ (1971) A predictive model of rainfall interception in forests, Derivation of the model from observations in a plantation of Corsican pine. Agricultural Meteorology 9: 367–383

Schellekens J, Scatena FN, Bruijnzeel LA, Wickel AJ (1999) Modelling rainfall interception by a lowland tropical rain forest in northeastern Puerto Rico. Journal of Hydrology 225: 168–184

Smolej I (1988) Gozdna hidrologija (Forest hydrology). In: Rejic M, Smolej I (eds) Sladkovodni ekosistemi, varstvo voda in gozdna hidrologija (Freshwater ecosystems, water conservation, and forest hydrology). University of Ljubljana, Biotechnical Faculty, Forestry Departement, pp 187–225 (in Slovene)

Solomatine DP, Dulal KN (2003) Model trees as an alternative to neural networks in rainfall-runoff modelling. Hydrological Sciences Journal 48: 399–411

Sraj M (2003a) Estimating leaf area index of the deciduous forest in the Dragonja watershed – Part I: Methods and measuring. Acta Hydrotechnica 21/35: 105–127

Sraj M (2003b) Modeliranje in merjenje prestrezenih padavin (Modeling and measuring of rainfall interception). Ph. D. Thesis, University of Ljubljana (in Slovene)

Stravs L, Kobold M, Brilly M (2004) Modeli kratkorocnih napovedi pretokov visokih voda na Savinji (Short-term flood forecasting models for the Savinja River). Proceedings – Misicev vodarski dan, Maribor (in Slovene)

Stuber M, Gemmar P (1997) An approach for data analysis and forecasting with neuro fuzzy systems – demonstrated on flood events at river Mosel. In: Proc. International Conference on Computational Intelligence, 5th Fuzzy Days, Dortmund

Witten IH, Frank E (2000) Data mining: Practical machine learning tools and techniques with java implementations. Morgan Kaufmann Publishers, San Francisco, USA

Chapter 26
Real-Time Flood Stage Forecasting Using Support Vector Regression

P.-S. Yu, S.-T. Chen and I-F. Chang

Abstract The support vector machine, a novel artificial intelligence-based approach developed from statistical learning theory, is used in this work to develop a real-time stage forecasting model. The orders of the input variables are determined by applying the hydrological concept of the time of response, and a two-step grid search method is used to find the optimal parameters, and thus overcome the difficulties of constructing the learning machine. Two structures of models used to perform multiple-hour-ahead stage forecasts are proposed. Validation results from flood events demonstrate that the proposed model can accurately forecast the flood stages one to four hours ahead. Moreover, two statistical tests are used to analyze the forecasting errors.

Keywords Flood forecasting · water stage · support vector regression

26.1 Introduction

The river stage, which can be measured directly and easily, is a more useful variable than discharge in forecasting floods, because the river stage triggers the authorities' issuance of a flood warning. Therefore, the river stage forecasting model has attracted increasing attention because of its usefulness in flood forecasting. Some recent studies in which hydrologic approaches are used to forecast river stages are as follows. See and Openshaw (1999, 2000) applied soft computing approaches to forecast river level, and integrated conventional and artificial intelligence-based models to provide a hybrid solution to the river-level problem. Krzysztofowicz (2002)

P.-S. Yu
Department of Hydraulic and Ocean Engineering, National Cheng Kung University, Taiwan,
e-mail: yups@mail.ncku.edu.tw

S.-T. Chen
Department of Hydraulic and Ocean Engineering, National Cheng Kung University, Taiwan

I-F. Chang
Department of Hydraulic and Ocean Engineering, National Cheng Kung University, Taiwan

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

used Bayesian forecasting to produce a short-term probabilistic river stage forecast. Thirumalaiah and Deo (1998), Liong et al. (2000), Bazartseren et al. (2003), Chang and Chen (2003), and Chau (2004) have also used the artificial neural network (ANN) approach, which is a popular data-driven model, to forecast river stage. Data-driven models based on artificial intelligence methods, such as neural networks, are favored and practically applicable in flood stage forecasting. This study uses the support vector machine (Vapnik 1998), a novel artificial intelligence-based method developed from statistical learning theory, to establish a real-time flood stage forecasting model.

The support vector machine (SVM), which is based on the structural risk minimization (SRM) principle, theoretically minimizes the expected error of a learning machine and so eliminates the problem of overfitting. Although SVM has been used in applications for a relatively short time, this learning machine has been proven to be a robust and competent algorithm for both classification and regression in many disciplines. Recently, the application of SVM has attracted attention in the water sector. Studies in which the SVM approach has been applied in hydrological modeling and forecasting are reviewed below.

Dibike et al. (2001) applied SVM to model the rainfall–runoff process using daily data, and demonstrated that the SVM model outperforms the neural network model. Sivapragasam et al. (2001) performed one-lead-day rainfall forecasting and runoff forecasting using SVM, in which the input data are pre-processed by singular spectrum analysis. Liong and Sivapragasam (2002) applied SVM to flood stage forecasting and concluded that the accuracy of SVM exceeds that of ANN. Choy and Chan (2003) used support vectors of the SVM to determine the structure of the radial basis function networks to model the relationship between rainfall and river discharge. Yu et al. (2004b) proposed a scheme that combined chaos theory and SVM to forecast the daily runoff. Bray and Han (2004) applied SVM to forecast runoff. Sivapragasam and Liong (2004) used a sequential elimination approach to identify the optimal training data set and then performed SVM to forecast the water level. Sivapragasam and Liong (2005) divided the flow range into three regions, and employed different SVM models to predict daily flows in high, medium and low regions.

In this study, SVM was used to establish a stage forecasting model, whose input vector accounts for both rainfall and river stage, to forecast the hourly stages of the flash flood. First, the study area was Lan-Yang Creek in northeastern Taiwan, and the rainfall and the river stage variables were chosen to account for the locations of stations and the attributes of the river basin. Most applications of SVM have depended on manual trial and error to determine the structure and the parameters of the model. Bray and Han (2004) demonstrated the difficulty of finding the optimum model structure and its parameters using an exhaustive search because of the many possible combinations. The orders of the input variables were determined based on the hydrological concept of the time of response, and a two-step grid search method was applied to find the optimal parameters to solve this problem and establish the structure of the model more systematically.

Then, two model structures, with different relationships between the input and output vectors, were presented to perform one- to four-hour-ahead stage forecasts in

real time. The results show that the proposed forecasting models with both structures yield similar accurate stage forecasts. Lastly, two statistical tests were undertaken on the error series of the forecasting models. The results of tests demonstrate that some error series are insignificant and appear as white noise, but some are not. This might suggest that the forecasting can be enhanced by some error correction scheme.

26.2 Support Vector Machine

SVM, which employs the structural risk minimization (SRM) principle, is a new approach for classification and regression. The methodology of support vector regression (SVR) is briefly described below (Vapnik 1999; Dibike et al. 2001; Liong and Sivapragasam 2002).

26.2.1 Linear Support Vector Regression

The support vector regression for the linear case finds a linear regression function that can best approximate the actual output vector y with an error tolerance ε. The decision function can be expressed as

$$f(w,b) = w \cdot x + b \tag{26.1}$$

where w and b are the parameter vectors of the function. The tolerated errors within the extent of the ε-tube, as well as the penalized losses L_ε when data are outside of the tube, are defined by Vapnik's ε-insensitive loss function as

$$L_\varepsilon(y_i) = \begin{cases} 0 & \text{for } |y_i - (w \cdot x_i + b)| \le \varepsilon \\ |y_i - (w \cdot x_i + b)| - \varepsilon & \text{for } |y_i - (w \cdot x_i + b)| > \varepsilon \end{cases} \tag{26.2}$$

Formally, this regression problem can be expressed as the following convex optimization problem.

$$\begin{aligned} \min_{w,b,\xi,\xi^*} \quad & \tfrac{1}{2} w^2 + C \sum_{i=1}^{l} (\xi_i + \xi_i^*) \\ \text{subject to} \quad & y_i - (w \cdot x_i + b) \le \varepsilon + \xi_i \\ & (w \cdot x_i + b) - y_i \le \varepsilon + \xi_i^* \\ & \xi_i, \xi_i^* \ge 0,\ i = 1, 2, \ldots, l \end{aligned} \tag{26.3}$$

where ξ_i and ξ_i^* are slack variables that specify the upper and the lower training errors subject to an error tolerance ε, and C is a positive constant that determines the degree of penalized loss when a training error occurs.

In this optimization problem, most data points are expected to be in the ε-tube. If a data set (x_i, y_i) is outside the tube, then an error ξ_i or ξ_i^* exists, which is to be minimized in the objective function. SVR avoids underfitting and overfitting the

training data by minimizing both the regularization term $w^2/2$ and the training error term $C\sum_{i=1}^{l}(\xi_i+\xi_i^*)$ in (26.3).

Introducing a dual set of Lagrange multipliers, $\underline{\alpha}_i$ and $\bar{\alpha}_i$, enables the optimization problem to be solved more easily in the dual form, by applying the standard quadratic programming algorithm:

$$\begin{aligned}\min_{\underline{\alpha}_i,\bar{\alpha}_i} \quad & \tfrac{1}{2}\sum_{i,j=1}^{l}(\underline{\alpha}_i-\bar{\alpha}_i)(\underline{\alpha}_j-\bar{\alpha}_j)<x_i\cdot x_j> \\ & +\varepsilon\sum_{i=1}^{l}(\underline{\alpha}_i+\bar{\alpha}_i)-\sum_{i=1}^{l}y_i(\underline{\alpha}_i-\bar{\alpha}_i) \\ \text{subject to} \quad & \sum_{i=1}^{l}(\underline{\alpha}_i-\bar{\alpha}_i)=0 \\ & 0\le\underline{\alpha}_i\le C,\ i=1,2,\ldots,n \\ & 0\le\bar{\alpha}_i\le C,\ i=1,2,\ldots,n\end{aligned} \tag{26.4}$$

where $<x_i\cdot x_j>$ is the inner product of x_i and x_j. After the Lagrange multipliers, $\underline{\alpha}_i$ and $\bar{\alpha}_i$, have been determined, the parameter vectors w and b can be estimated under Karush–Kühn–Tucker (KKT) conditions (Fletcher, 1987), which are not detailed herein. Therefore, the approximate function can be expressed as

$$f(x_i)=\sum_{i,j=1}^{l}(-\underline{\alpha}_i+\bar{\alpha}_i)<x_i\cdot x_j>+b \tag{26.5}$$

The values $(-\underline{\alpha}_i+\bar{\alpha}_i)$, corresponding to the data concerning the inside of the ε-insensitive tube, are zero. Hence, only the remaining nonzero coefficients $(-\underline{\alpha}_i+\bar{\alpha}_i)$ are involved in the final decision function, and the data that have nonzero Lagrange multipliers are called the support vectors. Simply, support vectors are those data that "support" the definition of the approximate function, whereas other data can be regarded as redundant. Finally, the approximate function can be rewritten as,

$$f(x_i)=\sum_{i=1}^{l}(-\underline{\alpha}_k+\bar{\alpha}_k)<x_i\cdot x_k>+b\ (k=1,2,\ldots,n) \tag{26.6}$$

where x_k stands for the support vector and n is the number of support vectors.

Another advantage of formulating the optimization problem in the dual form is shown in (26.6), in which the input vectors are multiplied as dot products. SVM can easily handle any increase in the input variables or the number of data in the input vectors, because the dot product of the two vectors can be calculated without difficulty. This feature is also useful in dealing with nonlinear SVR, as described below.

26.2.2 Nonlinear Support Vector Regression

In most real-world problems, linear function approximation is of limited practical use. The solution is to map the input data in higher dimensional feature space, in

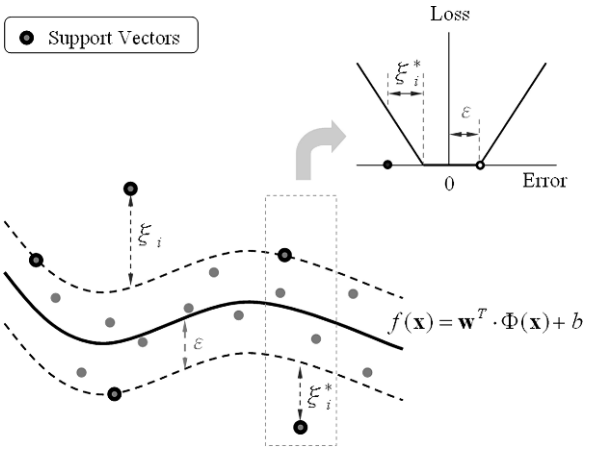


Fig. 26.1 Nonlinear SVR with ε-insensitive tube

which the training data may exhibit linearity, and then to perform linear regression in this feature space. Let x_i be mapped into a feature space by a nonlinear function $\phi(x_i)$; the decision function becomes

$$f(w,b) = w \cdot \phi(x) + b \tag{26.7}$$

Similarly, the nonlinear regression problem can be expressed as the following optimization problem. Figure 26.1 presents the concept of nonlinear SVR, corresponding to (26.8):

$$\begin{aligned} \min_{w,b,\xi,\xi^*} \quad & \frac{1}{2}w^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \\ \text{subject to} \quad & y_i - (w \cdot \phi(x_i) + b) \le \varepsilon + \xi_i \\ & (w \cdot \phi(x_i) + b) - y_i \le \varepsilon + \xi_i^* \\ & \xi_i, \xi_i^* \ge 0,\ i = 1,2,\ldots,l \end{aligned} \tag{26.8}$$

The dual form of the nonlinear SVR can then be expressed as

$$\begin{aligned} \min_{\underline{\alpha}_i, \bar{\alpha}_i} \quad & \frac{1}{2}\sum_{i,j=1}^{l}(\underline{\alpha}_i - \bar{\alpha}_i)(\underline{\alpha}_j - \bar{\alpha}_j) < \phi(x_i) \cdot \phi(x_j) > \\ & +\varepsilon\sum_{i=1}^{l}(\underline{\alpha}_i + \bar{\alpha}_i) - \sum_{i=1}^{l} y_i(\underline{\alpha}_i - \bar{\alpha}_i) \\ \text{subject to} \quad & \sum_{i=1}^{l}(\underline{\alpha}_i - \bar{\alpha}_i) = 0 \\ & 0 \le \underline{\alpha}_i \le C,\ i = 1,2,\ldots,n \\ & 0 \le \bar{\alpha}_i \le C,\ i = 1,2,\ldots,n \end{aligned} \tag{26.9}$$

Little knowledge may be available as a basis for selecting an appropriate nonlinear function $\phi(x_i)$, and further, the computation of $< \phi(x_i) \cdot \phi(x_j) >$ in the feature

space may be too complex to perform. An advantage of SVM is that the nonlinear function $\phi(x_i)$ need not be used. The computation in the input space can be performed using a "kernel" function $K(x_i, x_j) =< \phi(x_i) \cdot \phi(x_j) >$ to yield the inner products in the feature space, circumventing the problems intrinsic in evaluating the feature space. Functions that meet Mercer's condition (Vapnik 1999) can be proven to correspond to dot products in a feature space. Therefore, any functions that satisfy Mercer's theorem can be used as a kernel. The following radial basis function kernel was used in this work:

$$K(x_i, x_j) = \exp\left(-\gamma \left| x_i - x_j \right|^2\right) \tag{26.10}$$

Finally, the kernel function allows the decision function of nonlinear SVR to be expressed as follows:

$$f(x_i) = \sum_{i=1}^{l} (-\underline{\alpha}_k + \bar{\alpha}_k) K(x_i, x_k) + b \tag{26.11}$$

26.3 Study Area and Data Sets

Lan-Yang Creek, which encloses an area of 979 km^2, is located in the northeast of Taiwan. The length of the mainstream is about 73 km, and the slope of the bed is around 1/55. The mean annual precipitation is approximately 3,256 mm. Hourly water stage data (m) from two water level stations, Lan-Yang Bridge and Niu-Tou, and hourly rainfall data (mm) from rainfall stations were collected in this work. Simultaneous records obtained from water level stations for 19 flood events were extracted from these data collected from 1990 to 2004. Among these events,

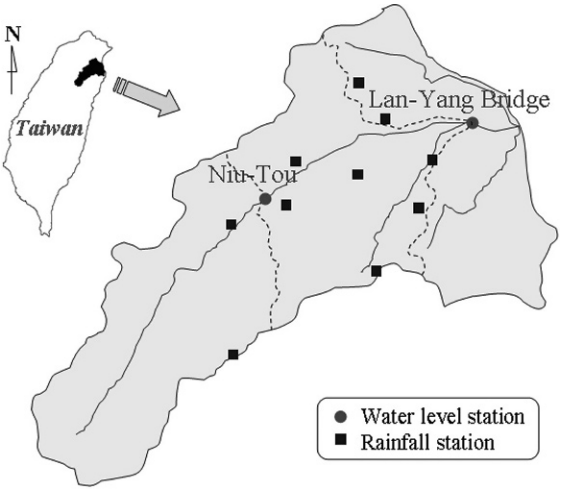


Fig. 26.2 Lan-Yang Creek basin

13 (904 data) were used for calibration, and 6 (560 data) were used for validation. Figure 26.2 shows the locations of the water level stations and ten rainfall stations, from which rainfall data were used to calculate the average rainfall, using the Thiessen polygon method, in the area between Niu-Tou and the Lan-Yang Bridge.

26.4 Model Construction

26.4.1 Determining the Input Vector

The water stage during a flood is a response to the stimuli of the routed streamflow from the upstream channel and the runoff from the rainfall–runoff process. Hence, multiple inputs were chosen as the observed stage at Niu-Tou and the observed rainfall in the area between Niu-Tou and the Lan-Yang Bridge. The observed stage at the Lan-Yang Bridge was also used as an input, because these data are strongly correlated with the data concerning future stage. Therefore, the input vector refers to three variables, including the stage at Niu-Tou, S_N; the stage at Lan-Yang Bridge, S_L; and the average rainfall of the intervening area, R.

Systematically determining both the orders and the parameters at the same time is prohibitively computationally burdensome, as identified by Bray and Han (2004). Therefore, the orders of the input variables are initially determined based on the hydrological modeling technique. Similar methods, including the time lag method and cross-correlation analysis, have been used by Solomatine and Dulal (2003), along with the average mutual information.

Two methods, both based on the idea of the time of response, were used to determine the order for the variable of stage at Niu-Tou, S_N. The first is to calculate the coefficient of correlation with different time lags n between the stage series $S_N(t-n)$ and $S_L(t)$, with respect to each flood event. The time lag from Niu-Tou to Lan-Yang Bridge is 3.31 hours, as determined by averaging the values of the calibration events. The other method is based on significant feature points, such as the peak point and the turning points, of the stage hydrograph. The time of response is the time between two corresponding feature points in $S_N(t)$ and $S_L(t)$. In some cases, in which the hydrographs are smooth, only the peak point is identified, without a turning point. Based on this method, the average time is 3.33 hours. Consequently, the order of variable S_N was set to three. That is, the input vector of the SVR model included $S_N(t-1)$, $S_N(t-2)$, and $S_N(t-3)$.

The order of rainfall R was determined from the time of concentration according to the same concept as determining the order of variable S_N. The time of concentration used herein is measured between the center of the rainfall hyetograph and the peak stage. The average time of concentration that pertains to calibration events is 5.28 hours, so the order of input R is set to five. Accordingly, $R(t-1)$, $R(t-2)$, $R(t-3)$, $R(t-4)$, and $R(t-5)$ are included in the input vector.

The order of the stage variable at Lan-Yang Bridge, S_L, could not be similarly determined. Therefore, the time series model, an autoregressive model, was used to determine the order. The stage time series pertaining to the calibration data at Lan-Yang Bridge can be identified as an AR(3) model, so the order of input S_L is three and that $S_L(t-1)$, $S_L(t-2)$, and $S_L(t-3)$ are included in the input vector.

The input vector of the SVR model contains 11 elements. That is, the forecasting model can be expressed as

$$S_L(t+1) = f_{SVR}[S_L(t+1-m_L),\ R(t+1-m_R),\ S_N(t+1-m_N)] \tag{26.12}$$

where $m_L = 1,2,3$; $m_R = 1,2,3,4,5$, and $m_N = 1,2,3$. In (26.12), the function f_{SVR} indicates the SVR model expressed explicitly as in (26.11), and t stands for the time index. At the "present" time t, available observations can be used in the input vector $[S_L(t+1-m_L), R(t+1-m_R), S_N(t+1-m_N)]$ to forecast the output variable $S_L(t+1)$ at "future" time $t+1$.

26.4.2 Normalizing Input Variables

Because the collected absolute water stage may not provide appropriate information to distinguish floods, the stage variable used in this work is the stage increment, relative to the initial stage at the time when the rainfall starts. That is, the value of initial stage, pertaining to each event, is subtracted from the stage series, and these differences are used in the SVR model.

The stage and the rainfall have different units and their values do not represent the same quantities, so all input variables were normalized to the interval from zero to one, according to the calibration data. This scheme can prevent the model from being dominated by the variables with large values, and is commonly used in data-driven models, such as ANNs. Bray and Han (2004) also showed that the SVM with normalized input data from zero to one outperforms that with unscaled input data. Therefore, the SVR model was fed normalized data, and then the model output stages were returned to their original scale. The initial water stage was added to these data to obtain stage forecasts that could be compared to the measurements.

26.4.3 Calibrating Parameters

The parameters that dominate the nonlinear SVM are the cost constant C, the radius of the insensitive tube ε, and the kernel parameters γ. The grid search

method was applied to calibrate these parameters more effectively and systematically to overcome the potential shortcomings of the trial and error method. The grid search method is a straightforward and exhaustive method. This method may be time-consuming, so Hsu et al. (2003) suggested the application of a two-step grid search method, applied on exponentially growing grids of parameters. First, a coarse grid search was used to determine the best region of these three-dimensional grids. Then, a finer grid search was conducted to find the optimal parameters. The root mean squared error (RMSE) was used to optimize the parameters. The optimal parameters $(C, \varepsilon, \gamma) = (26.91, 0.004, 0.105)$ were obtained at RMSE = 0.035 m, and the percentage of support vectors = 52.2%. The analyses and calculations of SVR herein were performed using *LIBSVM* software, developed by Chang and Lin (2001). Based on these derived parameters, (26.12) was used to perform one-hour-ahead stage forecasting.

26.4.4 Multiple-Hour-Ahead Forecasting Model

Equation (26.12) can be used as a dynamical model to provide stage forecasts two to four hours ahead to perform multiple-hour-ahead forecasting, as follows:

$$S_L(t+2) = f_{SVR}[S_L(t+2-m_L),\ R(t+2-m_R),\ S_N(t+2-m_N)] \tag{26.13}$$

$$S_L(t+3) = f_{SVR}[S_L(t+3-m_L),\ R(t+3-m_R),\ S_N(t+3-m_N)] \tag{26.14}$$

$$S_L(t+4) = f_{SVR}[S_L(t+4-m_L),\ R(t+4-m_R),\ S_N(t+4-m_N)] \tag{26.15}$$

where $m_L = 1,2,3$; $m_R = 1,2,\ldots,5$, and $m_N = 1,2,3$. Equations (26.13)–(26.15) cannot be immediately applied because R and S_N data are absent at "future" times $t+1$ to $t+3$, whereas the S_L data at "future" times are forecasts. Forecasting models can be constructed to forecast R and S_N, and thus overcome this difficulty. A simple scheme called naïve forecasting, by which the most recent observation is substituted for the forecasts, is employed. In the application of (26.12)–(26.15) to forecast stages one to four hours ahead, the optimal parameters $(C, \varepsilon, \gamma) = (26.91, 0.004, 0.105)$ are identically used. Such a model structure, from (26.12) to (26.15), is called model structure A.

Another model structure, which does not require future observations as inputs, is presented to perform multiple-hour-ahead forecasting. The available observations were input to (26.12), and were used to predict directly two- to four-hour-ahead stages. The equations are as follows:

Table 26.1 Calibration results of structure B

Model	Optimal parameters (C, ε, γ)	RMSE (m)	Pct. (%) of support vectors
f_{SVR}	(26.91, 0.004, 0.105)	0.035	52.2
$f_{2\mathrm{SVR}}$	(2.83, 0.006, 0.297)	0.058	48.7
$f_{3\mathrm{SVR}}$	(1.41, 0.006, 0.297)	0.084	57.8
$f_{4\mathrm{SVR}}$	(1.19, 0.008, 0.707)	0.097	55.3

$$S_{\mathrm{L}}(t+2) = f_{2\mathrm{SVR}}[S_{\mathrm{L}}(t+1-m_{\mathrm{L}}), R(t+1-m_{\mathrm{R}}), S_{\mathrm{N}}(t+1-m_{\mathrm{N}})] \tag{26.16}$$

$$S_{\mathrm{L}}(t+3) = f_{3\mathrm{SVR}}[S_{\mathrm{L}}(t+1-m_{\mathrm{L}}), R(t+1-m_{\mathrm{R}}), S_{\mathrm{N}}(t+1-m_{\mathrm{N}})] \tag{26.17}$$

$$S_{\mathrm{L}}(t+4) = f_{4\mathrm{SVR}}[S_{\mathrm{L}}(t+1-m_{\mathrm{L}}), R(t+1-m_{\mathrm{R}}), S_{\mathrm{N}}(t+1-m_{\mathrm{N}})] \tag{26.18}$$

where $m_{\mathrm{L}} = 1,2,3$; $m_{\mathrm{R}} = 1,2,\ldots,5$, and $m_{\mathrm{N}} = 1,2,3$. The functions $f_{2\mathrm{SVR}}$, $f_{3\mathrm{SVR}}$ and $f_{4\mathrm{SVR}}$ are, respectively, the SVR functions for forecasts two to four hours ahead. Such a model structure, (26.12) and (26.16)–(26.18) together, was called model structure B. Notably, for one-hour-ahead forecasting, model structure A is equivalent to model structure B. Table 26.1 lists the optimal parameters for model structure B, optimized using the aforementioned two-step grid search method for calibration events, along with the RMSE of calibration results and the percentage of support vectors. Notably, the percentages of support vectors of these six calibrated SVR models are around 50%. This consequence is consistent with that in the literature: Mattera and Haykin (1999) and Cherkassky and Ma (2004) mentioned that the SVR model performs optimally when the percentage of support vectors is about 50%.

26.5 Forecasting Results

26.5.1 Stage Forecasting

After the forecasting models had been established, both SVR model structures were used to forecast stages pertaining to six validation events at Lan-Yang Bridge. The proposed SVR models were used to forecast stage one to four hours ahead. Figure 26.3 presents the forecasted stage hydrographs for a validation event. Table 26.2 summarizes the forecasts. The forecasting results are good, and are only slightly poorer than the calibration results as identified by the values of RMSE, which demonstrate that model structure B performs better in two-hour-ahead forecasting, whereas model

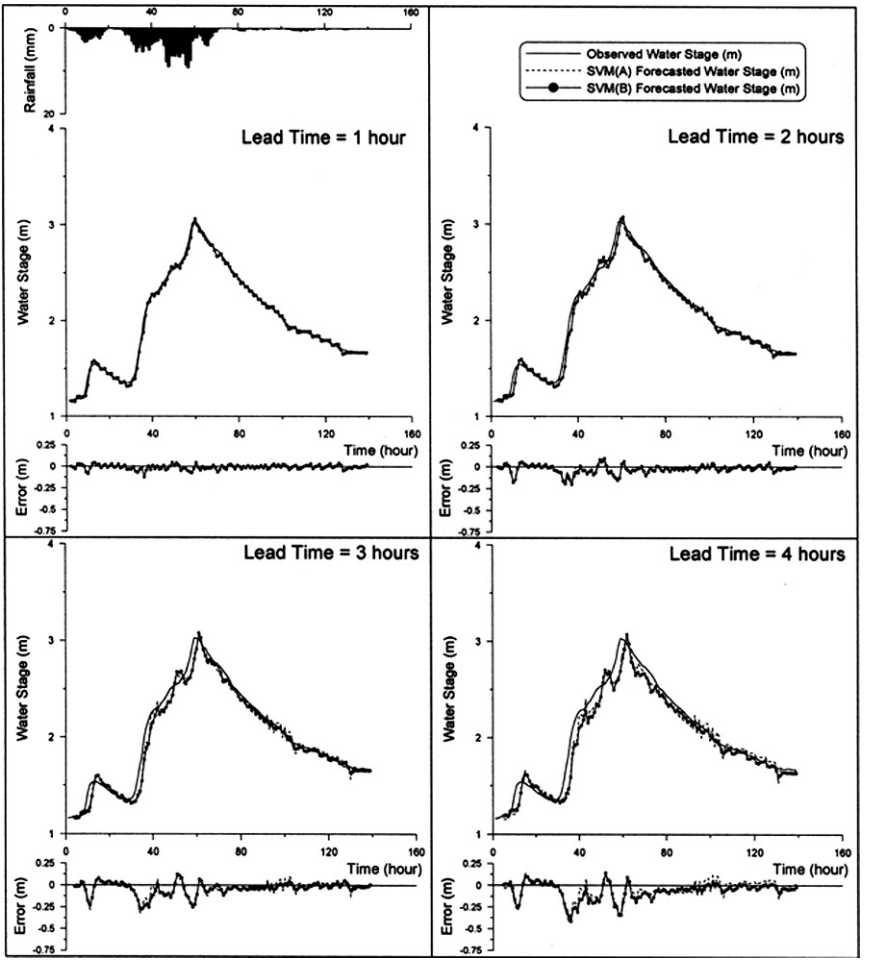


Fig. 26.3 Forecasting results

Table 26.2 Validation results

Lead time (hour)	RMSE (m)	
	Structure A	Structure B
1	0.056	0.056
2	0.102	**0.101**
3	0.146	0.146
4	**0.186**	0.218

Values in bold indicate better performance

structure A is superior in four-hour-ahead forecasting. The RMSE values of both model structures are close, suggesting that both model structures yield similarly good forecast.

26.5.2 Test of Error Series

Forecasting generally yields an error. If a forecasting model accurately describes a real system, the values of the error series will be small, random, and uncorrelated, and the error series will appear as white noise, with a mean of zero. Therefore, two statistical tests – the mean value test and the white noise test – are applied to investigate the error series (Mujumdar and Kumar 1990; Yu et al. 2004a).

26.5.2.1 Mean Value Test

The mean value of an error series of a forecasting model is desired to be around zero, if the performance of the model is to be statistically satisfactory. Student's t-test is usually used to determine these errors. Consider a variable $T(e)$,

$$T(e) = \bar{e} \Big/ \sqrt{\frac{\sigma^2}{n}} \tag{26.19}$$

where $\bar{e}$ is the mean of the error series, σ^2 the variance of the series, and n the number of data in the series. Assume that $T(e)$ follows a Student's t-distribution $t(\alpha, n-1)$. If $|T(e)|$ is less than a critical value $K = t_{\alpha/2}(n-1)$ with a significance level α, then the mean value of the error series is considered to be around zero.

26.5.2.2 White Noise Test

The Portmanteau test is performed here to test whether the error series is white noise. The variable $w(e)$ is defined as

$$w(e) = (n - n_1) \sum_{k=1}^{n_1} \left(\frac{R_k}{R_0} \right)^2 \tag{26.20}$$

where n_1 is set to 15% of the number of data, such that $n_1 = 0.15n$; R_k is the covariance with time lag k, and R_0 is the covariance with zero time lag. $w(e)$ is assumed to follow a chi-square distribution $\chi^2_\alpha(n_1)$. If $w(e)$ is less than a critical value of the chi-square distribution with a significance level α, then the error series is considered to follow a white noise process.

Table 26.3 Events that pass the tests

Lead time (hour)	Mean value test		White noise test	
	Structure A	Structure B	Structure A	Structure B
1	3	3	4	4
2	2	2	3	3
3	2	2	3	2
4	2	2	3	1

The statistical tests were conducted pertaining to six validation events. The numbers of events whose error series can pass the tests (with a significance level $\alpha = 0.05$) are listed in Table 26.3. The results show that the characteristics of errors from two model structures are similar. However, model structure B is inferior in three- and four-hour-ahead forecasting, because fewer events performed by model structure B can pass the white noise test. The results are analogous to that in Table 26.2 as identified by RMSE. We can observe that less than half the cases pass the tests. It indicates that in spite of good forecasting results of the proposed models, the errors are not shown as a white noise process for some cases. This may suggest that some amendment can be made to update the forecasting. The updating method of error correction can be considered as an appropriate approach to refine the stage forecasts.

26.6 Conclusion

SVR is used as a method to establish the flood stage forecasting model, using data collected from storm and typhoon events in Lan-Yang Creek, Taiwan. This work proposed an easy and systematic method for selecting the orders of the variables and the model parameters. The model parameters were calibrated using a two-step grid search method, and the orders of the input variables were investigated based on the hydrological concept of the time of response. These derived orders were used to propose two model structures to enable multiple-hour-ahead stage forecasts. The validation results reveal that both proposed model structures can easily predict the flood stage forecasts one to four hours ahead. Nevertheless, the statistical tests of error series indicate that some updating schemes, such as error correction, can be applied to potentially enhance the stage forecasting.

Acknowledgements The authors would like to thank the National Science Council of the Republic of China, Taiwan for financially supporting this research under Contract Nos. NSC92-2625-Z-006-003 and NSC93-2625-Z-006-001.

References

Bazartseren B, Hildebrandt G, Holz KP (2003) Short-term water level prediction using neural networks and neuro-fuzzy approach. Neurocomputing 55:439–450

Bray M, Han D (2004) Identification of support vector machines for runoff modeling. Journal of Hydroinformatics 6(4):265–280

Chang CC, Lin CJ (2001) LIBSVM: A Library for Support Vector Machines (Version 2.71, November 2004), Software available at: http://www.csie.ntu.edu.tw/∼cjlin/libsvm

Chang FJ, Chen YC (2003) Estuary water-stage forecasting by using radial basis function neural network. Journal of Hydrology 270:158–166

Chau K (2004) River stage forecasting with particle swarm optimization. Lecture Notes in Computer Science 3029:1166–1173

Cherkassky V, Ma Y (2004) Practical selection of SVM parameters and noise estimation for SVM regression. Neural Networks 17:113–126

Choy KY, Chan CW (2003) Modelling of river discharges and rainfall using radial basis function networks based on support vector regression. International Journal of Systems Science 34 (14–15):763–773

Dibike YB, Velickov S, Solomatine D, Abbott MB (2001) Model induction with support vector machines: Introduction and applications. Journal of Computing in Civil Engineering 15(3):208–216

Fletcher R (1987) Practical Methods of Optimization, 2nd edn. Wiley, New York

Hsu CW, Chang CC, Lin CJ (2003) A Practical Guide to Support Vector Classification. Available at: http://www.csie.ntu.edu.tw/∼cjlin/papers/guide/guide.pdf

Krzysztofowicz R (2002) Bayesian system for probabilistic river stage forecasting. Journal of Hydrology 268:16–40

Liong SY, Lim WH, Paudyal GN (2000) River stage forecasting in Bangladesh: Neural network approach. Journal of Computing in Civil Engineering 14(1):1–8

Liong SY, Sivapragasam C (2002) Flood stage forecasting with support vector machines. Journal of the American Water Resources Association 38(1):173–196

Mattera D, Haykin S (1999) Support vector machines for dynamic reconstruction of a chaotic system. In: Schölkopf B, Burges J, Smola A (eds) Advances in kernel methods: Support vector machine. MIT Press, Cambridge, MA.

Mujumdar PP, Kumar DN (1990) Stochastic models of streamflow: some case studies. Hydrological Sciences Journal 35(4):395–410

See L, Openshaw S (1999) Applying soft computing approaches to river level forecasting. Hydrological Sciences Journal 44(5):763–778

See L, Openshaw S (2000) A hybrid multi-model approach to river level forecasting. Hydrological Sciences Journal 45(4):523–536

Sivapragasam C, Liong SY (2004) Identifying optimal training data set – a new approach. In: Liong SY, Phoon KK, Babovic V (eds) Proceedings of the 6th International Conference on Hydroinformatics. World Scientific Publishing, Singapore

Sivapragasam C, Liong SY (2005) Flow categorization model for improving forecasting. Nordic Hydrology 36(1):37–48

Sivapragasam C, Liong SY, Pasha MFK (2001) Rainfall and runoff forecasting with SSA–SVM approach. Journal of Hydroinformatics 3(3):141–152

Solomatine DP, Dulal KN (2003) Model trees as an alternative to neural networks in rainfall-runoff modelling. Hydrological Sciences Journal 48(3):399–411

Thirumalaiah K, Deo MC (1998) River stage forecasting using artificial neural networks. Journal of Hydrologic Engineering 3(1):26–32

Vapnik VN (1998) Statistical Learning Theory. Wiley, New York

Vapnik VN (1999) An overview of statistical learning theory. IEEE Transactions on Neural Networks 10(5):988–999

Yu PS, Chen ST, Wu CC, Lin SC (2004a) Comparison of grey and phase-space rainfall forecasting models using fuzzy decision method. Hydrological Sciences Journal 49(4):655–672

Yu X, Liong SY, Babovic V (2004b) EC-SVM approach for real-time hydrologic forecasting. Journal of Hydroinformatics 6(3):209–223

Chapter 27
Learning Bayesian Networks from Deterministic Rainfall–Runoff Models and Monte Carlo Simulation

L. Garrote, M. Molina and L. Mediero

Abstract A mixed approach based on the combination of deterministic physically based models and probabilistic data-driven models for flood forecasting is presented. The approach uses a Bayesian network built upon the results of a deterministic rainfall–runoff model for real-time decision support. The data set for the calibration and validation of the Bayesian model is obtained through a Monte Carlo simulation technique, combining a stochastic rainfall generator and a deterministic rainfall–runoff model. The methodology allows making probabilistic discharge forecasts in real time using an uncertain quantitative precipitation forecast. The validation experiments made show that the data-driven model can approximate the probability distribution of future discharge that would be obtained with the physically based model applying ensemble prediction techniques, but in a much shorter time.

Keywords Flood forecasting · rainfall–runoff modelling · Bayesian networks · Monte Carlo simulation

27.1 Introduction

Real-time flood forecasting remains one of the most important challenges in operational hydrology. Although many different forecasting models have been developed and implemented in operational contexts, a consensus has not been reached

L. Garrote
Universidad Politécnica de Madrid, Dpto. Ingeniería Civil, Hidráulica y Energética, E.T.S. Ingenieros de Caminos, Ciudad Universitaria, 28040 Madrid, Spain, e-mail: garrote@caminos.upm.es

M. Molina
Universidad Politécnica de Madrid, Dpto. Ingeniería Civil, Hidráulica y Energética, E.T.S. Ingenieros de Caminos, Ciudad Universitaria, 28040 Madrid, Spain, e-mail: garrote@caminos.upm.es

L. Mediero
Universidad Politécnica de Madrid, Dpto. Ingeniería Civil, Hidráulica y Energética, E.T.S. Ingenieros de Caminos, Ciudad Universitaria, 28040 Madrid, Spain, e-mail: garrote@caminos.upm.es

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

regarding the best methodological approach. Practical approaches range from simple linear regression models to sophisticated ensemble prediction techniques using physically based distributed hydrological models. Despite all these theoretical and practical developments, limitations in forecast capability seriously affect the overall flood warning service quality. Problems like model adequacy, data assimilation and model calibration, real-time updating and operation, etc. are still very active research topics.

Three main lines of development have been pursued in the past: (1) physically based deterministic models, (2) statistical and stochastic models and (3) data-driven methods. All of them claim important successes, but they also exhibit weaknesses. Deterministic hydrological models range from simple, lumped, conceptual models to complex, distributed, physically based models. The most important strength of deterministic models is the support that they receive from practitioners. The deterministic approach remains the chosen option in the majority of operational systems. One of the reasons is that deterministic models are close to practitioners because they reproduce their intuitive reasoning about physical processes in the basin. However, deterministic models are not easy to include in operational systems. Calibration is very difficult (Beven and Binley, 1992), they are too rigid, cannot be updated easily and do not show their uncertainty explicitly. These limitations have been partially solved by the ensemble prediction technique (Day, 1985). A statistical distribution of input and model parameters is sampled to produce a large number of simulations which are summarized in a probabilistic forecast that describes the distribution of future conditions.

Stochastic methods use statistical techniques to infer model structure and estimate model parameters from historical records from past events. They range in sophistication from simple linear regression models to complex non-linear real-time filters built on dynamical system models. Stochastic models can incorporate uncertainty explicitly and many recursive updating techniques have been developed to correct model parameters, system state or both, based on data observed in real time (Brath and Rosso, 1993). The most significant drawback of stochastic models is that they rely on sophisticated mathematical techniques which are beyond the training of most practitioners on duty in flood forecasting centres.

A number of methods have been developed, which are based primarily on observational data and use some form of computational intelligence technique, such as neural networks or fuzzy representations. These data-driven methods are computationally more flexible and more efficient than complex physically based models for real-time use. Often, such models exhibit better predictive capabilities than more conventional approaches. However, their lack of physical interpretation limits the possibility for the end user to participate in model development. It is not easy to include the modeller's expert judgement or the local knowledge about the basin in data-driven models, since model structure and parameters are usually based exclusively on numerical data. Important qualitative factors, such as basin size, morphology, topography, slope, drainage network characteristics, geology, lithology, pedology, land use, infrastructure, etc. which are well known to the practitioner, are very difficult to incorporate because they cannot be translated into numerical

data. Furthermore, their application to flood forecasting is also limited because basins with long data sets for calibration or validation of these types of models are relatively scarce.

27.2 Methodology

In this chapter, a mixed approach based on the combination of deterministic physically based models and probabilistic data-driven models is presented. The approach uses a Bayesian network built upon the results of a deterministic rainfall–runoff model for real-time decision support. The knowledge of basin managers is encoded in the calibrated model. However, the structure of a deterministic model is not well suited for real-time forecasting because it does not facilitate model update or uncertainty estimation. For this reason, a probabilistic data-driven model is used to operate in real time. The data-driven model is calibrated with simulation results generated by the deterministic model through a Monte Carlo experiment.

A Bayesian network is a kind of data-driven model where the joint probability distribution of a set of related variables is inferred from observations. The network is defined on a set of qualitative random variables $U = \{X_1, X_2, \ldots, X_n\}$. A Bayesian network forms a directed acyclic graph where each node is a qualitative variable X_i defined on a finite domain and each link $R(i, j)$ represents the direct causal influence from variable X_i to variable X_j. The degree of influence between variables is expressed in terms of conditional probabilities between parent and child nodes in the network, $P\left(X_i | X_{u_1}, X_{u_2}, \ldots, X_{u_k}\right)$, where $\{u_1, u_2, .., u_k\}$ is the set of causes (parent nodes) of node i (child node).

The computational mechanism in Bayesian networks is an elaborated process derived from Bayes' theorem, a rule for updating the belief of a hypothesis h in response to evidence e:

$$P(h|e) = \frac{P(e|h)P(h)}{P(e)}$$

The solution algorithm of Bayesian networks allows the computation of the expected probability distribution of output variables conditioned to the probability distribution of the input variables. The complete inference process receives as input a set of random variables with known probability distributions for their values and propagates this evidence to update the belief of a set of goal variables, obtaining a probability distribution for the goal variables consistent with the known values, the causal relations and the conditional probabilities established in the network. The solution may be obtained, for instance, with the algorithm proposed by Pearl (1988), which was defined for networks with certain restrictions in the topology (networks called polytrees). Although the general problem of inference in unconstrained belief networks is NP-hard (Cooper, 1990; Dagum and Luby, 1993), other methods (Lauritzen and Spiegelhalter, 1988; Neapolitan, 1990) have been proposed for multiple connected Bayesian networks without the topology restrictions of polytrees. The causal representation of Bayesian networks is very appropriate

Table 27.1 Elementary networks to describe hydrologic processes

Process	Variables	Causal relations
Runoff generation	Rainfall: R Net rainfall: N Moisture content: M	$P(N^i{}_t \mid R^i{}_t, M^i{}_t)$, $P(M^i{}_t \mid R^i{}_{t-1}, M^i{}_{t-1})$
Runoff concentration	Net rainfall: N Discharge: Q	$P(Q^i{}_t \mid Q^i{}_{t-1}, N^i{}_{t-1}, \ldots, N^i{}_{t-k})$
Discharge propagation	Discharge i: Q^i Discharge j: Q^j	$P(Q^i{}_t \mid Q^i{}_{t-1}, Q^j{}_t, Q^j{}_{t-1})$

to construct models for prediction and diagnosis. For example, they have been successfully applied in complex domains such as medical diagnosis (Heckerman, 1991; Shwe and Cooper, 1991).

The Bayesian model presented here is described in Garrote and Molina (2004) and Molina et al. (2005). Hydrologic processes (runoff generation, runoff concentration, discharge propagation, etc.) are described through causal relations. The most relevant causal relations among variables are selected and described in a graph. Several examples of elementary networks describing basic hydrologic processes are presented in Table 27.1.

27.3 Deterministic Model Calibration

The proposed methodology has been implemented and tested in several basins controlled by two flood forecasting centres in Spain: Valencia and Málaga. Results from the Málaga application are presented here, corresponding to two basins: Guadalhorce and Guadalmedina, which are located upstream of the city of Málaga. The topography of the basins is steep in the headwaters, and very flat in the lowlands that consist mainly of irrigated land and urban areas. Both basins are subject to recurrent flooding problems, and have one or more reservoirs that are operated for water supply and flood control. The Basin Authority (Cuenca Mediterránea Andaluza) is responsible for reservoir operation and the Regional Government coordinates civil defence operations.

A fairly conventional deterministic rainfall–runoff model based on Hortonian runoff generation and the unit hydrograph was selected to test the approach. Model choice was based on simplicity, conditioned by the preferences of operators on duty in the flood forecasting centre of Málaga. Runoff generation was simulated with the Soil Conservation Service (SCS) curve number model (Mockus, 1972). Runoff concentration was simulated with the synthetic SCS unit hydrograph (Snider, 1973). A total of 16 episodes were available for the case study basins, although basin response was significant only in a few cases (three to five) in each subbasin. Parameter estimation was difficult, since all subbasins exhibited very irregular behaviour. For instance, Fig. 27.1 shows data of total precipitation versus total runoff in four subbasins, compared with theoretical predictions for different values of the curve

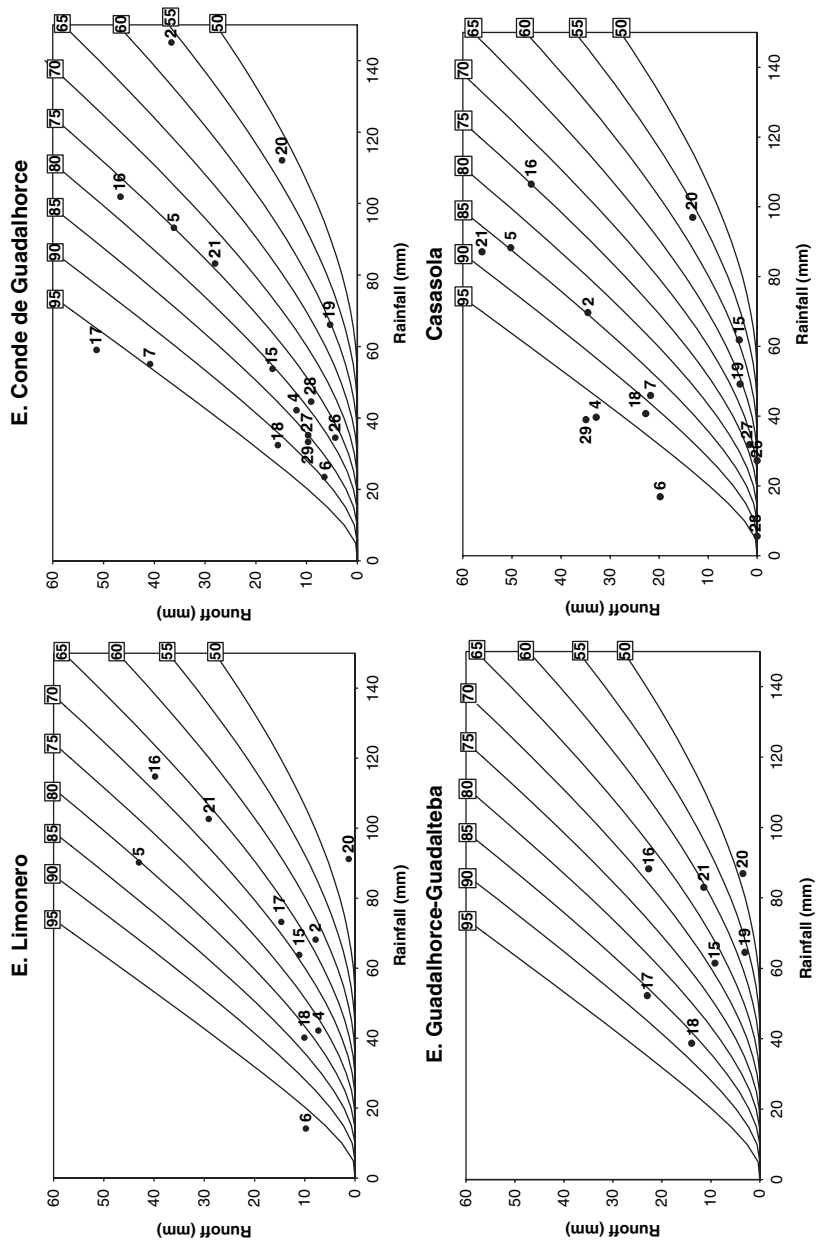


Fig. 27.1 Scatter plot of precipitation versus runoff in four subbasins compared with the curve number model

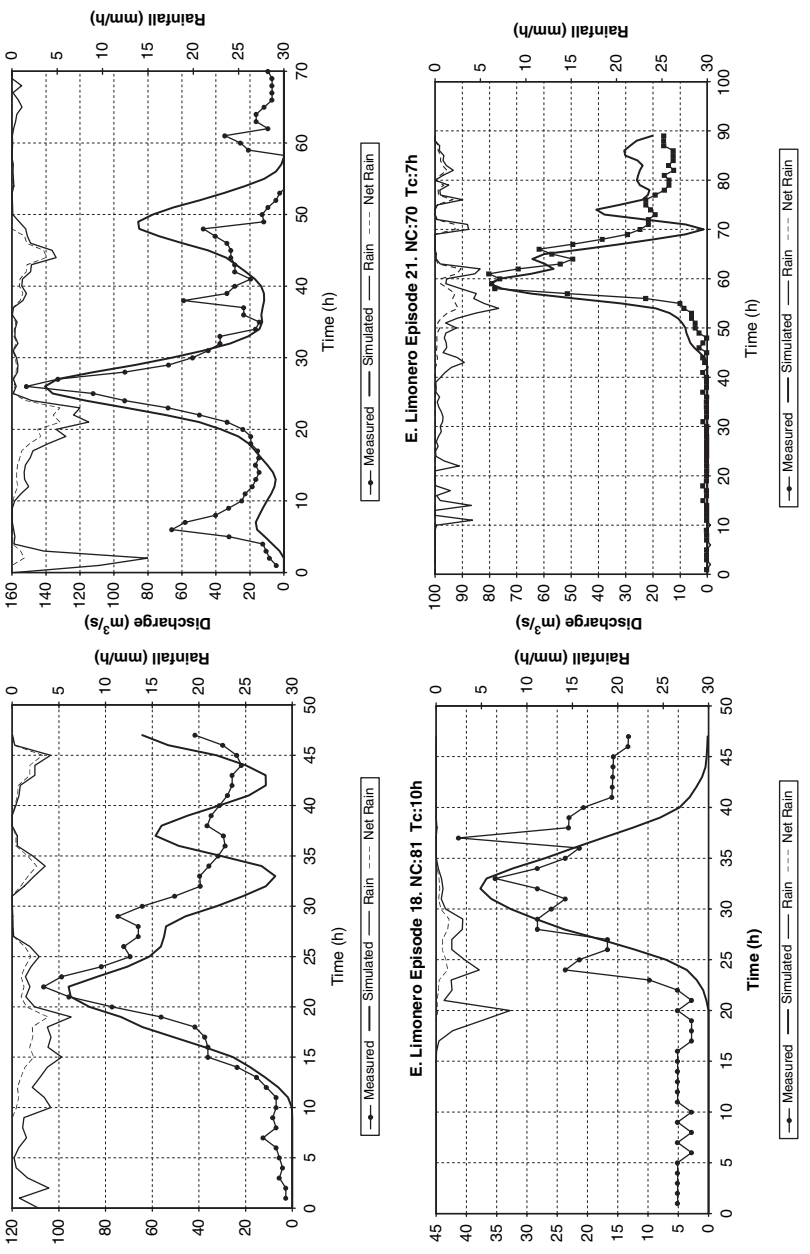


Fig. 27.2 Results of model calibration in the Limonero basin for four episodes

number model parameter. The large variability observed in the data can be attributed to model inadequacy, variability in the basins' initial conditions, uncertainty in rainfall or runoff estimation (in fact, in several episodes, the estimated runoff was larger than the estimated rainfall) or several other factors.

The effect of the variability is that it is impossible to find a single set of model parameters that would fit all the observations. For instance, the results of model calibration for four episodes in the Limonero basin are presented in Fig. 27.2, showing the best fit obtained for every episode. Values of the curve number parameter range from 70 to 81 and values of the lag time range from 7 to 10 hours. Therefore, after the calibration process, model parameters remained uncertain and were considered random variables.

Results of the calibration process are summarized in Table 27.2, where the range of values of model parameters is presented. Since only a few episodes were used for model calibration, there was no indication regarding the probability distribution of model parameters. A uniform probability distribution in the estimated range was selected for sampling in the Monte Carlo experiment presented in the next section.

Table 27.2 Summary of calibration results for the deterministic model

Basin	Runoff generation curve number	Runoff concentration lag time (h)
Guadalteba	70–85	9.5–10.5
Guadalhorce	65–90	17–19
Conde de Guadalhorce	50–80	9.5–10.5
Casasola	75–90	3.5–4.5
Cártama	65–90	11–13
Campanillas	75–90	1–1.5
Guadalhorce final	65–90	1.5–2.5
Limonero	70–80	7–10

27.4 Monte Carlo Simulation

The data set for the calibration and validation of the Bayesian model is obtained through a Monte Carlo simulation technique, combining a stochastic rainfall generator and the deterministic rainfall–runoff model. The stochastic rainfall generator simulates the generation and evolution of convective rainfall cells with exponential decay. The model is organized as a set of spatial rainfall generation entities (convective cells or frontal systems) that move over the basin topography according to the synoptic meteorological situation. Model input includes a list of descriptors of the synoptic meteorological situation and a simple mechanism to generate the storm velocity field. The rainfall generation entities are created randomly through a spatial Poisson process. Every entity originates an elliptical bell-shaped rainfall field in

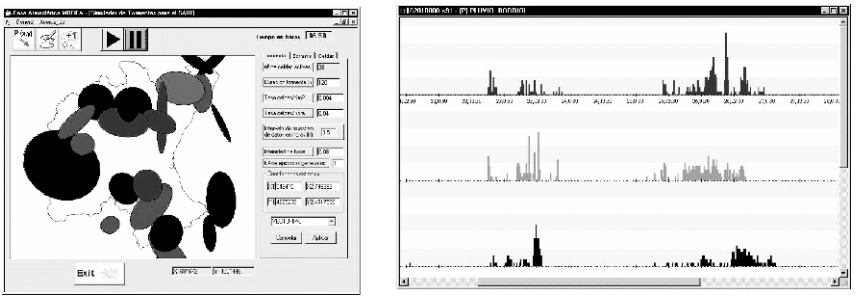


Fig. 27.3 User interface of the rainfall simulator and time series of synthetic rainfall at selected locations

its area of geographical influence. The parameters that govern rainfall cell density, creation rate, maximum rainfall intensity, size and temporal evolution are random, and are sampled from probability distributions during the Monte Carlo simulation. Total rainfall field is the sum of the contribution of all rainfall generation entities, plus added white noise to introduce more randomness in the process. The result is a rainfall field that produces time series at specific locations which are similar to values observed in rain gauges. A computer application implementing the model generates synthetic rainfall series at selected locations in the basin. A sample screen of the user interface showing cell size and position in an intermediate time step is presented in Fig. 27.3, together with three time series of rainfall generated in one model realization.

The stochastic rainfall model was run to generate 3,000 hours of synthetic rainfall series in storms of different duration in the same location as the actual rain gauges in the basin. The deterministic rainfall–runoff model was run with the synthetic rainfall series and random parameters to generate time series of discharge at the basin outlet. A total of 100 model runs were performed for each synthetic storm. Model results were used to obtain time series of rainfall, net rainfall and discharge at the nodes of the Bayesian network. The result of the Monte Carlo simulation experiment is a large database which is consistent with the prior knowledge of basin behaviour encoded in the deterministic model and with the results of the calibration process, which incorporates actual observations and expert judgement.

27.5 Bayesian Model Learning

The database of simulated events contains a variety of basin behaviours expressed in numerical values which are used for the learning process of the Bayesian network. The first step is to convert numerical values into qualitative values in the discrete domains of the Bayesian network variables. The probability distribution of model results for each combination of qualitative values of the independent variables is then estimated by analysing the simulation results. This process is illustrated in

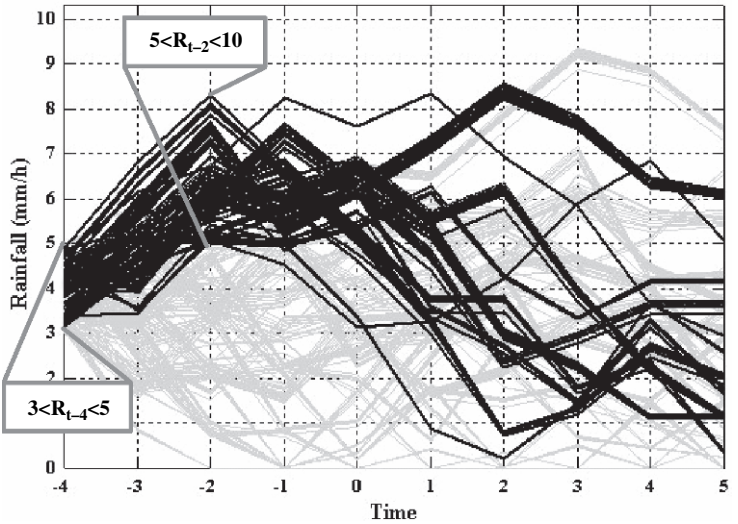


Fig. 27.4 Sample data analysis for a group of model results

Fig. 27.4, where model results for net rainfall are shown for the combination of a value of rainfall in the interval [3,5] for time step $t-4$ and [5,10] for time step $t-2$.

The former analysis is applied in a moving window that samples the entire series of simulated results. Results obtained for all instances of the same combination of values of the independent variables are collected and analysed, to obtain the probability distribution of the dependent variable. For instance, histograms of frequencies of discharge forecasted in three time steps into the future are presented in Fig. 27.5. These distributions are converted into qualitative values using the corresponding membership functions. The qualitative time series thus generated are used in the learning process of the Bayesian network. For each combination of values for independent variables the distribution of the corresponding value for the dependent variable is estimated through conditional probabilities. If a sufficient number of cases are generated, conditional probabilities are estimated applying the classical formula of frequencies.

For the causal relation: $X_{u_1}, X_{u_2}, \ldots, X_{u_n} \rightarrow X_i$, the conditional probability $P(X_i = x | X_{u_1} = x_1, \ldots, X_{u_n} = x_n)$ is estimated with the equation

$$P(X_i = x | X_{u_1} = x_1, \ldots, X_{u_n} = x_n) = \frac{N(X_i = x, X_{u_1} = x_1, \ldots, X_{u_n} = x_n)}{N(X_{u_1} = x_1, \ldots, X_{u_n} = x_n)}$$

where $N(X = x)$ indicates the number of cases in the sample where the variable X gets the value x.

A set of about 300,000 cases produced by simulation was used in the learning process of each Bayesian network. The number of cases was adjusted verifying that a sufficient number of instances for all combinations of discrete values for each set

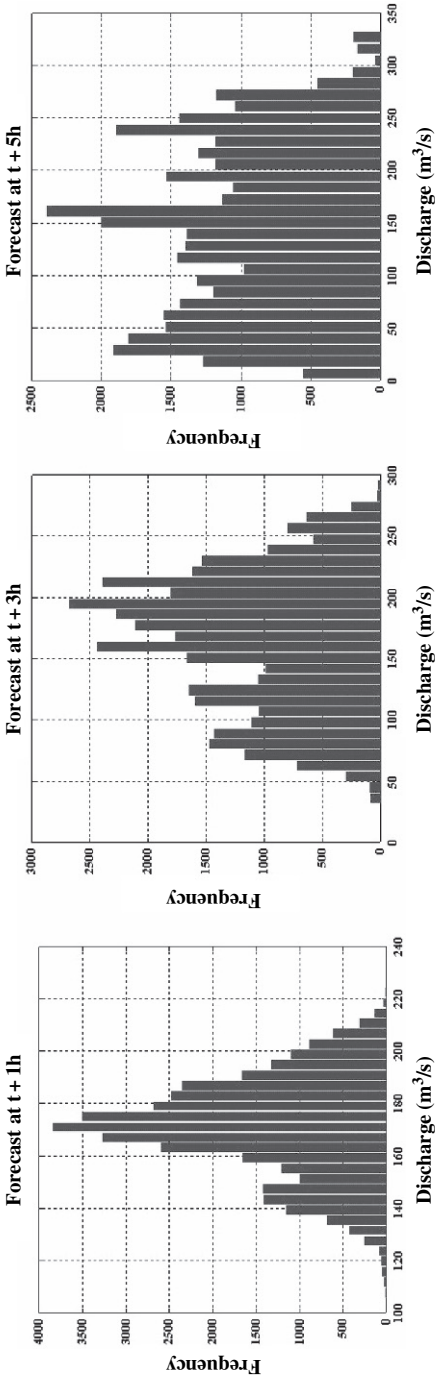


Fig. 27.5 Histogram of simulated discharge in three time horizons

of cause nodes should be present. This guarantees that the Bayesian network learned from all physically possible situations.

Once the conditional probabilities were estimated, a process of network validation was applied to verify the adequateness of network structures and the usefulness of their predictions. This involved comparing different Bayesian networks describing a single process in order to select the best representation using objective criteria. There are several measures of model performance proposed in the literature of Bayesian networks, such as conditional entropy (Herskovitz and Cooper, 1990) or mutual information (Friedman et al., 1999). With the help of these types of parameters, the quality of the Bayesian networks was estimated. The global process of network definition and calibration was repeated until a satisfactory performance was obtained, using the conclusions of the evaluation to refine part of the models.

27.6 Application

The Bayesian model presented has been included in a decision support system called SAIDA (Spanish acronym for Intelligent Agents Society for Decision-making during Floods). SAIDA is a computerized system based on artificial intelligence techniques that provides assistance in flash flood situations for basin control centres (Cuena and Molina, 1999). The goal of the system is to support an effective conversation between an operator and a computerized system for decision support purposes. The key idea of SAIDA is to place emphasis primarily on decision issues. Classical simulation and forecasting issues are kept in the background, although they can be activated at the user's request.

The main function of SAIDA is to inform basin managers and civil protection officials of the development of a flood situation and to suggest possible lines of action. The objective is to select the most relevant information in every situation and present it in a clear and concise way to the decision-making centre. The operator can quickly understand the current situation, can identify the main problems that have to be solved and can be briefed on the actions that could be taken to minimize the risks and reduce the problems.

SAIDA uses Bayesian networks to make probabilistic forecasts of the time evolution of representative variables related to the flood threat. A sample of the forecast is presented in Fig. 27.6, which shows the mean value and the probability distribution of forecasted discharge in four time steps into the future, using the qualitative domain selected for the variable. Each of the categories in the domain corresponds to a scenario relevant for decision making, and has a list of actions associated with it. The decision maker is provided with an estimation of the probabilities of reaching each of those scenarios in several time horizons.

The individual Bayesian network models are combined in an "influence network", a computational structure that relates hydrologic variables through Bayesian networks. The forecasts are presented on a geographical display as coloured icons associated to relevant hydrologic variables or problem areas in the river network, as shown in Fig. 27.7. By activating a problem area, the operator can obtain a descrip-

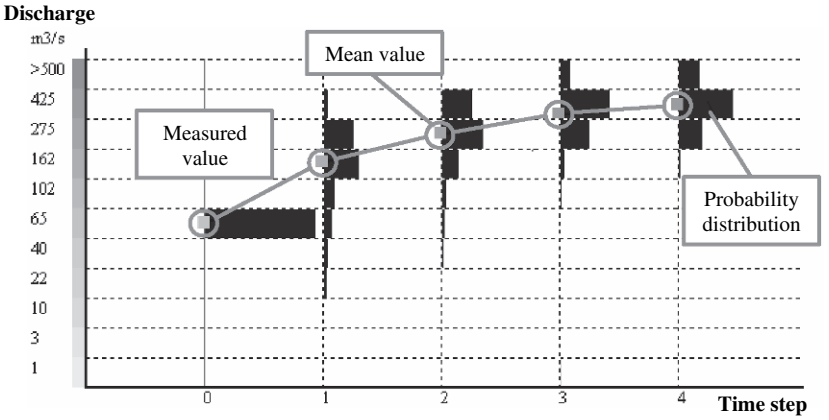


Fig. 27.6 Probabilistic predictions of the Bayesian model for four time steps

tion of the relationship between relevant hydrologic variables related to the problem and a probabilistic forecast of the time evolution of those variables in qualitative terms.

This approach is useful for making predictions in complex basins with short response times, because Bayesian networks can make inference using the probability distribution of variables directly. The ensemble prediction technique could also pro-

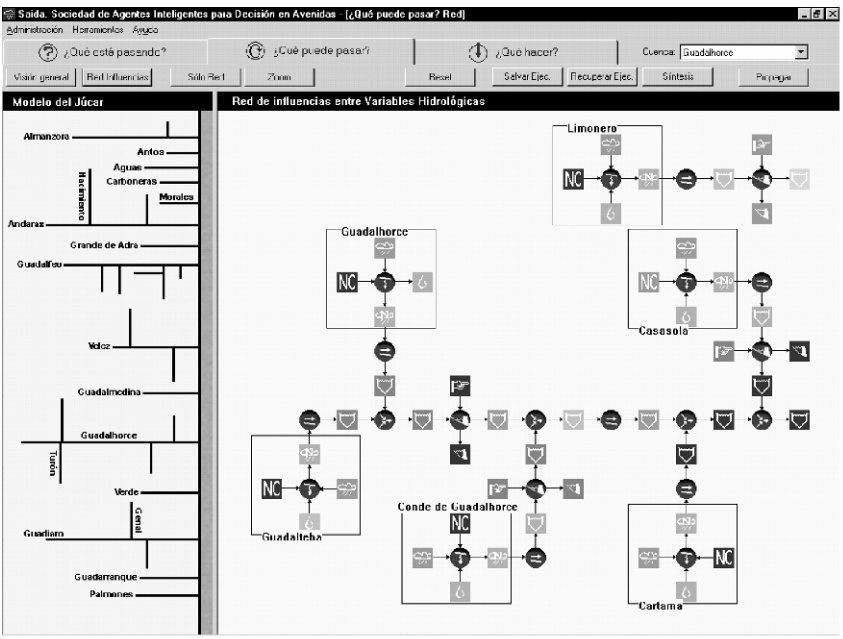


Fig. 27.7 Example of influence diagram for the Guadalhorce River basin

vide this type of decision support information, but it is computationally unfeasible, because a large number of combinations are needed in order to cover the ensemble. Bayesian networks present two main advantages. First, the computational burden is carried out in advance, not during the flood. The number of simulations is not conditioned by the time window available to make the forecast. Secondly, the computational effort is divided into individual steps. The influence network is used to establish relations among variables in basins of complex topology, using the individual Bayesian networks. In a large basin, the computational effort required grows linearly with the number of subbasins, not exponentially, as in the case of ensemble prediction.

27.7 Conclusion

A methodology to calibrate Bayesian networks using a deterministic rainfall–runoff model was presented. The methodology allows the user to make probabilistic discharge forecasts in real time using an uncertain quantitative precipitation forecast. The approach is based on the generation of cases for Bayesian network learning through a Monte Carlo experiment, using a stochastic rainfall generator and a deterministic rainfall–runoff model with random parameters. The validation experiments made show that the data-driven model can approximate the probability distribution of future discharge that would be obtained with the physically based model applying ensemble prediction techniques, but in a much shorter time.

The computational structure of the Bayesian network also allows for an efficient user interface for real-time decision support. The Bayesian model described has been included in the SAIDA decision support system, to provide the user with a quick estimate of the general situation in the basins and a probability distribution of expected damage in problem areas. All this computing structure can be integrated in a geographical overview that is easy to understand by decision makers and which can provide information on the current situation and possible evolution of areas of interest.

Acknowledgements The research was carried out under grant REN2003-09021 of Spain's Inter-Ministerial Commission of Science and Technology, through the National Plan of Scientific Research, Development and Technological Innovation (I+D+I).

References

Beven KJ, Binley A (1992) The future of distributed models: model calibration and uncertainty prediction. Hydrological Processes 6: 279–298.

Brath A, Rosso R (1993) Adaptive calibration of a conceptual model for flash flood forecasting. Water Resources Research 29(8): 2561–2572.

Cooper GF (1990) The computational complexity of probabilistic inference using Bayesian belief networks. Artificial Intelligence 42: 393–405.

Cuena J, Molina M (1999) A multi-agent system for emergency management in floods, in Multiple Approaches to Intelligent Systems, Iman I, Kodratoff Y, El Dessouki A, Ali M (eds) Lecture Notes in Computer Science Vol. 1611, Springer, pp. 460–469.

Dagum P, Luby M (1993) Approximating probabilistic inference in Bayesian belief networks is NP-hard. Artificial Intelligence 60(1): 141–153.

Day GN (1985) Extended streamflow forecasting using NWSRFS. Journal of Water Resources Planning and Management 111: 157–170.

Friedman N, Nachman, I, Dana P (1999) Learning Bayesian network structure from massive datasets: The Sparse Candidate algorithm. Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, pp. 206–215.

Garrote L, Molina M (2004) A framework for making probabilistic forecasts using deterministic rainfall-runoff models, in Hydrological Risk: Recent Advances in Peak River Flow Modelling, Prediction and Real-time Forecasting, Brath A, Montanari A, Toth E, (eds), Bios, pp. 239–246

Heckerman D (1991) Probabilistic Similarity Networks. MIT Press, Cambridge, Massachusetts.

Herskovitz EH, Cooper GF (1990) Kutató: An entropy-driven system for the construction of probabilistic expert systems from data. Proceedings of the Sixth Conference on Uncertainty in Artificial Intelligence, pp. 54–62.

Lauritzen SL, Spiegelhalter DJ (1988) Local computations with probabilities on graphical structures and their application to expert systems. Journal of the Royal Statistical Society B 50(2): 157–224.

Mockus V (1972) Estimation of direct runoff from storm rainfall, in National Engineering Handbook. Part 630: Hydrology, NRCS, pp. 10.1–10.24.

Molina M, Fuentetaja R, Garrote L (2005) Hydrologic models for emergency decision support using Bayesian networks, in Symbolic and Quantitative Approaches to Reasoning with Uncertainty, Godo L (ed) Lecture Notes in Computer Science 3571: 88–99.

Neapolitan RE (1990) Probabilistic Reasoning in Expert Systems. Theory and Algorithms, Wiley.

Pearl J (1988) Probabilistic reasoning in intelligent systems: Networks of plausible inference. Morgan Kaufmann.

Shwe M, Cooper G (1991) An empirical analysis of likelihood-weighting simulation on a large, multiply connected medical belief network. Computer and Biomedical Research 24(5): 453–475.

Snider D (1973) Hydrographs, in National Engineering Handbook. Part 630: Hydrology, NRCS, pp. 16.1–16.23.

Chapter 28
Toward Bridging the Gap Between Data-Driven and Mechanistic Models: Cluster-Based Neural Networks for Hydrologic Processes

A. Elshorbagy and K. Parasuraman

Abstract The emergence of artificial neural network (ANN) applications in hydrology, among other data-driven techniques, has created a new chapter in this field of science that has been termed "neurohydrology" (Abrahart, 1999). However, the number of operational solutions is still limited because most of the practitioners and many researchers have difficulty accepting ANNs as a standard technique. This chapter presents an attempt to bridge the gap between ANN models and mechanistic hydrologic models using cluster-based ANN models. A novel spiking modular neural network (SMNN) model is proposed, configured, and trained in a way that assigns different sub-components of the hydrologic processes to corresponding sub-components of the developed SMNN model. The modular nature of the SMNN helps to find domain-dependent relationships. The proposed model configuration reflects the ability of the ANN technique to mimic mechanistic models in terms of their sensible internal structure and the way they model various hydrologic processes. The meteorological data, including air temperature, ground temperature, wind speed, relative humidity, and net radiation, from a semi-arid region in northern Alberta, Canada, are used in this chapter to estimate actual evapotranspiration (AET). The estimated AET is contrasted against the values measured by an eddy covariance system. The results of the SMNN model, which uses an unsupervised clustering technique to group the input data into a definite number of clusters, are compared with the results of another modular ANN model that relies on a supervised clustering technique, and with a traditional global feedforward ANN model. The estimates of AET using the Penman–Monteith model are also presented for reference and comparison purposes. The SMNN model provided the best performance and was shown to be effective in discretizing the complex mapping space into simpler domains that are easier to learn.

A. Elshorbagy
Centre for Advanced Numerical Simulation (CANSIM), Department of Civil & Geological Engineering, University of Saskatchewan, Saskatoon, SK, Canada S7N 5A9

K. Parasuraman
Centre for Advanced Numerical Simulation (CANSIM), Department of Civil & Geological Engineering, University of Saskatchewan, Saskatoon, SK, Canada S7N 5A9

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

Keywords Clustering · spiking modular neural networks · actual evapotranspiration · predication · K-mean

28.1 Introduction

In the last decade, artificial neural network (ANN) applications have come to represent a large portion of the water resources research literature. The presence of a large number of ANN applications in water resources can be attributed to various reasons. The ability of ANNs to behave as universal approximators (Hornik et al., 1989; Elshorbagy et al., 2001), the higher accuracy of predictions compared to other techniques (Hsu et al., 1995; Elshorbagy et al., 2000a; Abrahart and See, 2000), the ease of use especially when the understanding of the underlying physics is not readily available (Jayawardena and Fernando, 1998; ASCE Task Committee on Artificial Neural Networks in Hydrology, 2000), and the fascination that users have in this technique are possibly the main reasons for adopting ANNs in water resources. The ANN-related research in the water resource literature ranges from pure applications for studying various hydrologic processes (Bastarache et al., 1997; French et al., 1992) to the introduction of new network types and network training algorithms (Hsu et al., 2002; Parasuraman et al., 2006), and includes important state-of-the-art reviews (Maier and Dandy, 2000), to attempts at devising methodologies for the determination of optimum model inputs to ANN models (Bowden et al., 2005).

The continuous and persistent flow of ANN applications to hydrology and water resources, along with the emergence of various training algorithms, has contributed to the creation of a new chapter in hydrology that has been termed "neurohydrology." However, regardless of the maturity of neural networks as a science and the related improvement of its associated training algorithms, the technique remains in its infancy when it comes to real-world usage and adoption by water resource practitioners or decision makers. This may reflect a lack of confidence in the technique and the gap between researchers and practicing engineers. This problem stems partly from the fact that such tools are perceived to ignore the physics and partly because they are black-box models that induce relationships only from the data. Practitioners, concerned with the opaqueness of the technique, fail to recognize any similarity between ANNs and mechanistic (process-based) models. ANNs are seen to be ignoring seasonality in the process, lacking sensible internal structure, and not filtering inputs based on any physical grounds. Until these issues are addressed, neurohydrology will remain a research endeavor that lacks real-world validation.

The aim of this chapter is to address some of the issues contributing to the opaqueness of ANNs in hydrology and to present a methodology for pattern recognition and the clustering of inputs, both prior to and within ANN modeling. The proposed supervised and unsupervised clustering could help make the behavior, the inputs, and the internal structure of the ANN model more sensible to process-based oriented users and decision makers.

28.2 Problems of ANN Applications in Water Resources

The main problems and concerns about the applications of ANNs to water resources issues, which have been and still are core areas of research, can be briefly presented under the following eight categories:

1. *Identification of the optimum set of inputs.* In spite of recent notable efforts made to address this issue (Bowden et al., 2005), the question of how and on what basis the optimum set of inputs can be selected remains open. One possible approach is to match the set of inputs used by a process-based model counterpart, and let the ANN model filter out some of the inputs.
2. *Choice of global or module-based configuration.* Zhang and Govindaraju (2000) and Parasuraman and Elshorbagy (2007) partitioned the input space and thereafter developed a separate ANN model for each component, whereas Hsu et al. (2002), Hong et al. (2005), and Parasuraman et al. (2006) devised ANN models that could cluster the input space before modeling. Nevertheless, solid criteria or guidelines to select either a universal or a modular ANN model in each case are still non-existent.
3. *Determination of the most appropriate network architecture.* There are cases of reported success using recurrent neural networks (RNNs) (e.g., Coulibaly et al., 2001) but although such models can capture the dynamics of the hydrologic processes they nevertheless fail to provide clear practical advantages over the fully connected feedforward networks (Hochreiter and Schmidhuber, 1997). To date, there are no guidelines on which type of networks to employ in different cases.
4. *Selection of the best transfer function.* Sigmoidal-type transfer functions are the most commonly used functions in ANN applications (Maier and Dandy, 2000). However, Moody and Yarvin (1992) found that non-sigmoidal transfer functions perform best when the data are noiseless and contain highly non-linear relationships. This finding is supported by the promising performance of wavelons (Parasuraman and Elshorbagy, 2005), rather than neurons, for learning the highly non-linear signals and processes.
5. *Deciding on the training algorithm.* The error function is usually minimized using iterative first order (gradient descent) or second order (Newton's method) techniques. Genetic algorithms (GA) have been employed recently to improve the performance of ANNs (Jain and Srinivasulu, 2004; Parasuraman and Elshorbagy, 2007; Dawson et al., 2006). Traditional GA-trained ANN models can suffer from slow convergence, which casts some doubts on the practical advantages.
6. *Combining or separating network outputs.* Even though most practical applications would involve more than one variable of interest to be predicted, such a problem has not been adequately addressed in the literature. Most process-based hydrologic models have the capability of simulating more than one output (e.g., runoff and evapotranspiration). Therefore, ANN models cannot be compared to process-based models or appeal to process hydrologists without being able to

predict more than one output simultaneously. While minimizing multiobjective error functions for multiple variables, trade-off decisions become more complex and lead to inevitable compromise with regard to the prediction accuracy of the outputs.

7. *Assessment of the performance of ANN models.* Jain and Srinivasulu (2004) argue that absolute relative error (ARE) is the best measure since it is drastically different from the measure that is used during training (e.g., MSE). Elshorbagy et al. (2000a) developed the pooled mean squared error (PMSE) to assess the performance of the ANN model. They proved that it is an efficient measure because it combines the properties of both MSE and ARE in one measure. It also helps avoid obtaining contradictory conclusions based on various error measures.
8. *Interrelation of the previous problems.* Apparently, the most complicated problem among the set of previously mentioned problems is the fact that they are interrelated. None of these problems can be solved independently from all others. Problem 1 is related to problems 2, 3, and 6. Problem 3 is related to problems 4 and 5, while problem 4 is related to problems 2 and 5. It should not be forgotten that each one of the first six problems is related to problem 7.

This chapter focuses on problem 2, by aiming to address the issue of using input clustering to achieve a sensible network structure and to improve the overall performance of the ANN model. The main objectives are thus twofold: to take a step toward bridging the gap between mechanistic and data-driven models; and to raise the hopes of developing more conceptually sensible ANN models.

28.3 Methodology

28.3.1 Supervised k-Mean Clustering and Modeling Using a Modular Neural Network (SkCNN)

In this chapter, a methodology is developed for modeling a multiple-input–single-output (MISO) process using a cluster-based neural network model. The methodology can be extended for multiple-input–multiple-output (MIMO) processes. The following section gives a detailed explanation of the proposed and adopted methodology.

The concept of clusters or patterns in hydrology has been studied by Panu and Unny (1980), Booy and Morgan (1985), Elshorbagy et al. (2000b), and others. Mining of such patterns would help to produce more intelligent models. The first step in modeling hydrologic data using ANNs requires that the training data set should be clustered if the data points are not uniformly distributed (Shi, 2002). The number of clusters may be determined by constructing a periodogram (Panu et al., 1978). An alternative way to determine the optimal number of clusters is by trial and error (Roiger and Geatz, 2002). In the proposed methodology, clustering is achieved using the *k*-means algorithm as follows: (i) the total number of clusters, *k*,

is arbitrarily chosen; (ii) k instances (data points) within the data set are chosen at random to be the initial cluster centers; (iii) the remaining instances are assigned to their closest cluster centers based on Euclidean distance; (iv) the center (mean) of each cluster is calculated based on the instances in that cluster; and (v) if the current mean values are identical to the mean values of the previous iteration, the process is terminated. Otherwise, using the current means as cluster centers, steps (iii)–(v) are repeated.

Once clustering is achieved, each cluster is modeled using an individual neural network. Training is carried out using the Bayesian-regularization method. Once individual training is accomplished, the cluster-based neural network models are tested. Testing is performed by first assigning each testing instance to a specific cluster based on the proximity of the testing instance to the cluster centers. Secondly, the ANN model developed for the cluster under consideration is used to evaluate the testing instance. In the case of MISO processes, the training data are clustered based on their distribution in multi-dimensional coordinate space [inputs (I_1, I_2,...), output (Q)]. A coordinate mapping of the training data can be visualized by plotting I(n) vs. Q(n); thus, a cluster center is represented by a point in the coordinate space. For testing, the proximity of each testing instance to different cluster centers is determined based on input sequence alone because the output sequence is considered to be unknown.

28.3.2 Unsupervised Clustering and Modeling Using Spiking Modular Neural Networks

The structure of the SMNNs (Parasuraman et al., 2006) is shown in Fig. 28.1. The input layer consists of j input neurons ($x_1, x_2, x_3, \ldots, x_j$), where j is equal to the number of input variables. The input layer neurons are connected to the spiking layer, which serves as the *memory* of the system, learning and storing different input patterns that can be used in classifying future input vectors based on patterns learned during the training process.

In the spiking layer, clustering of the input space is achieved by unsupervised (or self-organized) learning. Self-organized learning consists of repeatedly modifying the synaptic weights of a neural network in response to activation patterns and in accordance with prescribed rules, until a final configuration appears (Haykin, 1999). Furthermore, self-organizing networks can learn to detect regularities and correlations in the input space, and accordingly adapt their future responses to certain input patterns.

Self-organization in networks can be achieved in two ways: (1) competitive learning, which produces a competitive network and (2) the self-organizing map (SOM) (Demuth and Beale, 2001). In competitive learning, the neurons of the network compete among themselves to be active (spike), the winner of which is called a *winner-takes-all* neuron (Haykin, 1999). SOMs are a special case of a self-organizing system because they learn to recognize groups of similar input vectors in such a way that neurons, which are topologically near to each other in the neuron layer

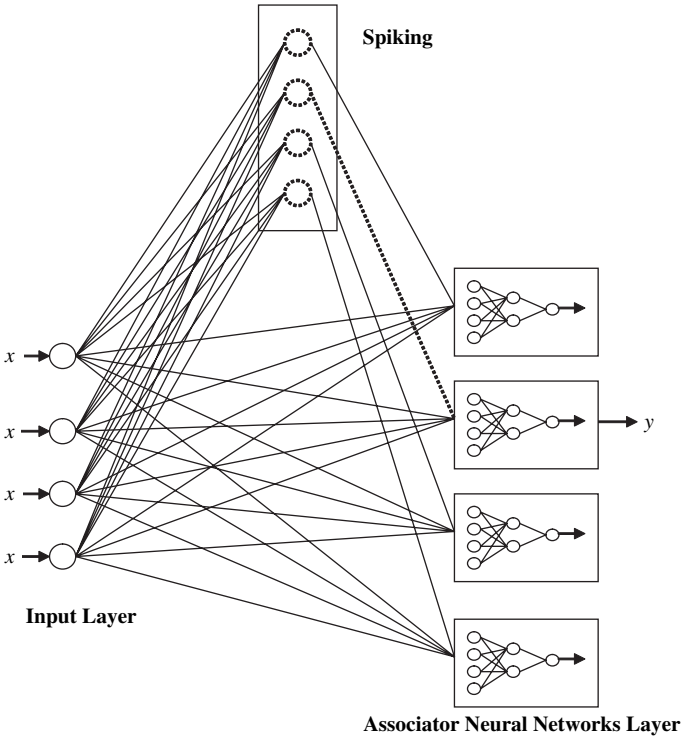


Fig. 28.1 Structure of a spiking-modular neural network (SMNN)

respond to similar input vectors. A SOM is therefore characterized by the formation of a topographic map of the input patterns in which the spatial locations of the neurons in the lattice are indicative of intrinsic statistical features contained in the input patterns (Haykin, 1999). Hence, the main difference between competitive learning and the SOM is that while the former learns only the distribution, the latter learns both the distribution and the topology (neighboring neurons) of the input space.

The weights of the self-organizing networks are initialized to the center of the input ranges. Once initialized, the neurons of the self-organizing network are trained by the Kohonen learning rule (Kohonen, 1989) to identify the clusters in the input space and allow the connection weights of the neuron to learn an input vector. Each neuron in the self-organizing network competes to respond to an input vector. Proximity of inputs to each neuron is determined based on Euclidean distance (d_c) as given in (28.1):

$$d_{c=1..m} = \left[\sum_{j=1}^{J} (x_j - w_{cj})^2 \right]^{0.5} \tag{28.1}$$

where m denotes the number of clusters and w_{cj} represents the connection weight linking the jth input variable and cth neuron of the self-organizing network. In the

case of competitive learning, the neuron whose weight vector is most similar to that of the input vector is updated to be even more similar. However, in self-organizing maps, along with the most similar neuron, the neurons in the neighborhood of the most similar neuron are also updated to be even more similar. The result of such training leads to a neural network model where the winning neuron is more likely to win the competition the next time a similar vector is presented, and less likely to win when a very different input vector is presented. For a given input vector, the neuron which represents the cluster that is closest to the input vector produces an output value of 1 (spikes), while the remaining neurons output 0. More information on self-organizing networks can be found in Demuth and Beale (2001) and Kohonen (1989).

Following classification of the input space, mapping of inputs to the corresponding outputs has to be carried out. In this study, mapping of inputs to outputs is achieved by neural networks and because these networks associate input patterns to outputs, they are termed *associator* neural networks. The associator neural networks are similar to feedforward neural networks.

SMNN performance employing both competitive learning and self-organizing maps is tested. The first solution uses a competitive network to provide the spiking layer and will be referred to as SMNN(CN). The second variant uses a SOM to provide the spiking layer and will be referred to as SMNN(SOM). Since competitive networks learn only the distribution of the input space, whereas SOMs learn both the distribution and topology of the input space, comparison of the two approaches would demonstrate the effect of topology learning on the performance of SMNNs.

28.3.3 Network Training and Generalization

A traditional global feedforward neural network (FFNN) and the associator neural networks in the cluster-based models were trained using the Bayesian-regularization backpropagation algorithm. This algorithm aims to improve the generalization property of neural networks, by developing solutions that contain smaller weights and biases, and with a smoother response that is less likely to result in overfitting (Demuth and Beale, 2001). Hence, along with minimizing the mean-squared error (MSE), the cost function in the Bayesian-regularization backpropagation algorithm (28.2) involves minimizing the mean of the sum of squares of the network weights and biases. This minimization function decreases the chances of overfitting by minimizing the values of the network connection weights and biases, which in turn leads to a smoother response function (surface). In (28.2), y_i and $y_i^{'}$ represent the measured and computed counterparts; N and n represent the number of training instances and the number of network parameters, respectively. The success of the regularization depends on the choice of an appropriate value of the regularization parameter, α. In this study, the method by MacKay (1992) is adopted, where the optimal

α is determined automatically within a Bayesian framework. A systematic search of different network configurations and user-adjustable parameters was carried out to ascertain the optimal network architecture, with the objective of minimizing the cost function. The optimal network architecture is the one which results in the least cost function:

$$\text{MSE\_REG} = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - y_i'\right) + (1-\alpha)\left(\frac{1}{n}\sum_{j=1}^{n} w_j^2\right) \tag{28.2}$$

28.3.4 Performance Evaluation

Since no single evaluation measure is suitable for evaluating the predictive abilities of ANN models (Sudheer and Jain, 2003), a multicriteria assessment is carried out in this study to evaluate the relative performance of different models. The criteria employed in this study include root-mean-squared error (RMSE), mean relative error (MRE), correlation coefficient (R), and coefficient of efficiency (E). The RMSE and the MRE provide different types of information about the predictive capabilities of the model. RMSE is more sensitive to errors at high values and outliers, whereas MRE provides a more balanced perspective of the goodness of fit at moderate values (Karunanithi et al., 1994). The correlation coefficient (R) evaluates the linear correlation between the observed and computed values, and the coefficient of efficiency (E) evaluates the skill of the model in predicting values away from the mean. The RMSE, MRE, R, and E statistics are defined by (28.3), (28.4), (28.5), and (28.6), respectively, where N represents the number of instances presented to the model; y_i and y_i' represent measured and computed counterparts; and $\bar{y}$ represents the mean of the corresponding variable:

$$\text{RMSE} = \left[\frac{1}{N}\sum_{i=1}^{N}\left(y_i - y_i'\right)^2\right]^{0.5} \tag{28.3}$$

$$\text{MRE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - y_i'}{y_i}\right| \tag{28.4}$$

$$R = \frac{\sum_{i=1}^{N}\left(y_i - \overline{y_i}\right)\left(y_i' - \overline{y_i'}\right)}{\sqrt{\sum_{i=1}^{N}\left(y_i - \overline{y_i}\right)^2 \sum_{i=1}^{N}\left(y_i' - \overline{y_i'}\right)^2}} \tag{28.5}$$

$$E = 1 - \frac{\sum_{i=1}^{N}\left(y_i - y_i'\right)^2}{\sum_{i=1}^{N}\left(y_i - \overline{y_i}\right)^2} \tag{28.6}$$

28.4 Case Study

South Bison Hill (SBH), a waste overburden pile located north of Fort McMurray, Alberta, Canada, is considered in this study. SBH was constructed with waste-rock material from oil sands mining in stages between 1980 and 1996. It covers an area of 2 km^2, rises 60 m above the surrounding landscape, and has a large flat top several hundred meters in diameter. To reclaim the overburden so that it can sustain vegetation, the underlying shale is covered by a 0.2 m layer of peat on top of a 0.8 m layer of till. The top is dominated by foxtail barley; also present are other minor species such as fireweed. Estimation of the evapotranspiration (ET) from the reconstructed watershed is of vital importance because it plays a major role in the water balance of the system.

Micrometeorological techniques were used to directly measure ET and the surface energy balance. A mast located in the approximate center of SBH was equipped to measure air temperature (AT), relative humidity (RH), ground temperature (GT), all-wave net radiation (R_n), and wind speed (WS). All instruments were connected to a data logger sampled at 10 seconds and an average or a cumulate record was logged every half-hour. The energy balance of the surface is given by

$$R_n = \mathrm{LE} + H + G + \varepsilon \tag{28.7}$$

where LE is the latent heat flux (evaporation when divided by the latent heat of vaporization), H the sensible heat flux, G the ground heat flux, and ε the residual flux density, all expressed in W/m^2. LE and H were measured directly via the open-path eddy covariance (EC) technique (Leuning and Judd, 1996). G was measured using a CM3 radiation and energy balance (REBS) ground heat flux plate placed at 0.05 m depth. Latent heat fluxes were corrected for changes in air density (Webb et al., 1980) and sensible heat fluxes were calculated using the sonic virtual temperature (Schotanus et al., 1983). Flux measurements were also removed during periods of rainfall and during periods of unexpected change in state variables. No gap filling was performed. The training sets consist of 500 instances of hourly data (between 20 May 2003 and 9 June 2003), whereas 247 instances of hourly data (between 18 June 2003 and 28 June 2003) comprise the testing set.

28.5 Results and Analysis

The performance of different models in modeling LE (i.e., AET) as a function of AT, RH, GT, R_n, and WS is given in Table 28.1. Using the trial-and-error method, the optimal number of clusters for all the cluster-based models is found to be eight, and the optimal number of hidden neurons in the associator neural networks is found to be four, i.e., ANN(5,4,1), which was the same network configuration for the global FFNN model. In general, the neural network models performed better than the Penman–Monteith (PM) method in estimating the EC-measured hourly LE flux (Table 28.1, the performances of the best models are provided in bold). For

Table 28.1 Performance statistics of different models in estimating LE as a function of AT, GT, RH, WS, and R_n

Model	Training				Testing			
	RMSE	MRE	R	E	RMSE	MRE	R	E
PM	88.1	4.9	0.76	–0.06	92.5	3.2	**0.74**	–0.18
FF-NN	37	2.4	0.9	0.81	73.4	1.5	0.69	0.26
SkCNN	**16.8**	**1.3**	**0.98**	**0.96**	77.3	3.3	0.6	0.18
SMNN (CN)	32.2	2.2	0.93	0.86	**70.2**	**1.2**	0.71	**0.32**
SMNN (SOM)	33.4	2.0	0.92	0.85	73	1.3	0.67	0.27

training and testing periods, the PM estimates resulted in an E of -0.06 and -0.18, respectively. This indicates that the model predictions are worse than predictions using a constant equal to the average observed value. When comparing different neural network models, during training, the k-means supervised-cluster based model performed better than other models. However, during testing, the performance of that model deteriorated considerably, which is indicative of overfitting. During testing, the SMNN(CN) model outperformed other models in terms of RMSE, MRE, and E statistics. From Table 28.1, it can be noticed that there is considerable variability between the training and testing RMSE statistics. This variability in RMSE statistics is due to the following reason: the median of the LE data set during training and testing are 34.1 W/m^2 and 61.2 W/m^2, respectively. Since the median gives a measure of central tendency, it implies that, compared to the training data set, the testing data set is dominated more by higher values of evapotranspiration. Roughly, the values of LE in the testing data set are two times (61.2/34.1) higher than the values of LE in the training data set. Since squared error statistics gives more weight to high values, RMSE during testing is approximately twice as high as the RMSE during training, preserving the ratio between the medians. This indicates that the neural network models are not overtrained.

From the above discussions, it can be concluded that the SMNN(CN) model performed better than the other models. The better performance of the supervised-cluster based model during training and significant performance deterioration during testing may be attributed to the following reason: during training, clustering is based on both the inputs and the output, and during testing, clustering is based on the inputs alone as the output is considered to be unknown. In order to test the robustness of the adopted clustering, a confusion matrix (Roiger and Geatz, 2002) is generated to check the classification correctness. The classification correctness can be evaluated by initially clustering the data with the input sequence alone and then comparing the cluster instances with the corresponding instances when clustering is carried out with both input and output sequences. For this case study, which had eight clusters, approximately 20% accurate clustering has been achieved. Any improvement in the accuracy of clustering is expected to improve the prediction accuracy of the supervised-cluster based model. In order to test the above hypothesis, the study is extended to test the performance of a supervised-cluster based ANN model if 100% exact clustering is achieved. In this case, presuming that the output is known

a priori, clustering during testing is carried out using both the inputs and the output. The RMSE, MRE, R, and E statistics of this model are 27.90 W/m^2, 0.90, 0.95, and 0.89, respectively, which is a significant improvement over the performance of the actual supervised-cluster based model. This reinforces the earlier discussed heuristic that an improvement in classification accuracy could significantly improve the performance of the supervised-cluster based model.

28.5.1 Optimal Combination of Input Parameters

The process of evapotranspiration is controlled by different factors at different scales – vapor pressure deficit and stomatal processes are the driving variables at the scale of a single leaf or tree, whereas radiation is the driving variable at the regional scale (Jarvis and McNaughton, 1986). Hence, in this study, an effort has been made to determine the minimum number of inputs in modeling EC-measured LE at an hourly scale (the half-hour measurements were aggregated to hourly scale). Different combinations of inputs were tested with the objective of minimizing (28.2). Results indicate that using net radiation and ground temperature alone as the inputs for neural network models can result in better prediction accuracy. Although most of the evaporation models use a water vapor pressure gradient to estimate evaporation, inclusion of RH as one of the inputs for the neural network model does not improve the performance of the model because RH is a redundant variable for all tested neural network models, which have already learnt the signal of RH that is embedded in the signal of GT, due to strong land–atmosphere interaction. This reiterates the findings of Lakshmi and Susskind (2001) and Wang et al. (2004), where it is reported that evaporation is not sensitive to atmospheric humidity since the land surface states contain the signals of near-surface atmospheric conditions as a result of strong land–atmosphere interaction. This research finding is of significant importance because it would reduce the dependence on the input variables whose measurements are intricate.

Table 28.2 shows the performance of different models in estimating LE as a function of GT and R_n. The performances of the best models are provided in bold. Similar to the previous case, the optimal number of clusters and the architecture of associator neural networks are found using the trial-and-error method. The optimal number of clusters for all the cluster-based models is found to be four and the optimal architecture of the regular FFNN and the associator neural networks

Table 28.2 Performance statistics of different models in estimating LE as a function of GT and R_n

Model	Training				Testing			
	RMSE	MRE	R	E	RMSE	MRE	R	E
FF-NN	43.7	2.9	0.86	0.74	67.2	1.5	0.72	0.38
SkCNN	**24.3**	**0.7**	**0.96**	**0.92**	64.8	1.1	**0.76**	0.42
SMNN(CN)	43.9	3.3	0.86	0.74	**64.4**	**1.1**	0.74	**0.43**
SMNN(SOM)	43.9	3.5	0.86	0.74	65.9	0.9	0.73	0.40

is found to be ANN(2,4,1). Comparing Tables 28.1 and 28.2, it is found that in general, although there is an increase in training error when R_n and GT alone are used to model LE, the testing results show improvement. This indicates that the generalization property of the neural network models increases when GT and R_n alone are used as inputs, provided that the same data subsets are used for training/testing the models under consideration. The SMNN(CN) model performed relatively better than the other models in terms of RMSE and E. However, compared to the previous case, the difference in performance among the cluster-based models is not that significant because the number of clusters in this case is less. The classification accuracy of the supervised-cluster based model was 48%. The performance of the supervised-cluster based model is also evaluated when 100% classification accuracy is achieved. The RMSE, MRE, R, and E statistics of the supervised-cluster based model with 100% classification accuracy are 31.86 W/m^2, 0.54, 0.93, and 0.86, respectively.

In general, this study finds that the neural networks outperform the widely adopted PM method in estimating EC-measured LE. When the regular FFNNs

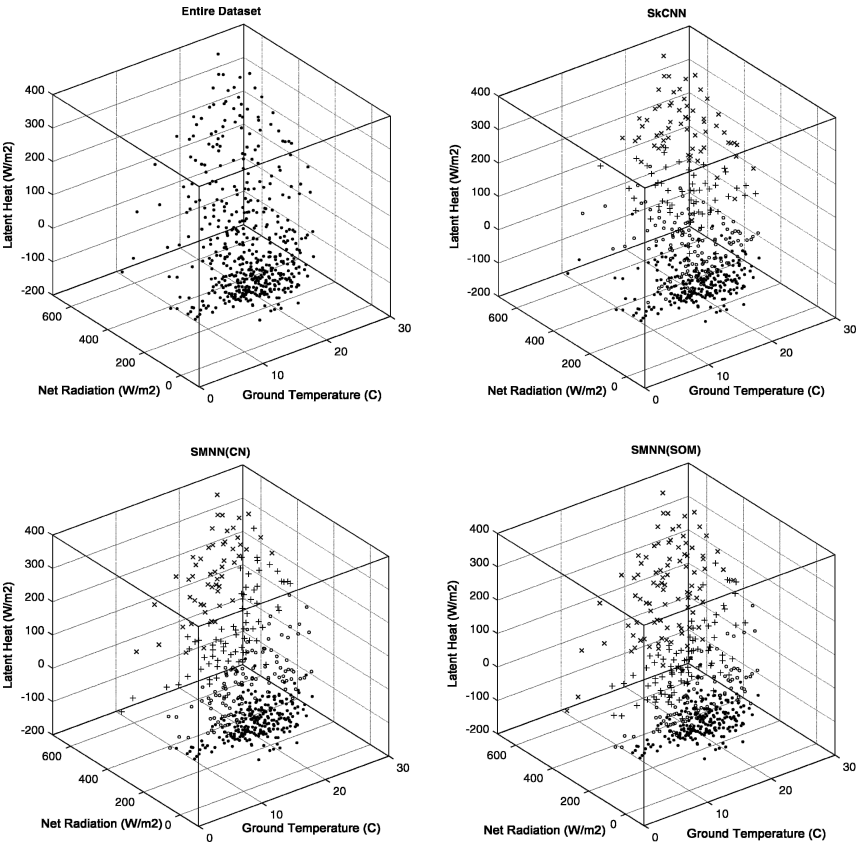


Fig. 28.2 Plots showing the performance of different cluster-based models in decomposing the complex mapping space into simpler domains

were compared with their cluster-based equivalents, the cluster-based models performed better than the FFNNs. Among the cluster-based models, the SMNN(CN) performed better than other models. The superiority of the SMNN(CN) is more when compared to the supervised-cluster model, which indicates that the clustering technique is more important than inclusion or exclusion of topology during the network training. In order to appreciate the clustering achieved by each of the cluster-based models, a plot showing how the different cluster-based models delineated the mapping space is shown in Fig. 28.2. From the figure, it can be seen that both the SMNN(CN) and SMNN(SOM) have relatively well-defined cluster boundaries, as compared to the supervised-cluster based model. Figure 28.2 also demonstrates that cluster-based models are effective in decomposing the complex mapping space into simpler sub-domains that can be learnt with relative ease by the individual associator neural network models.

28.6 Conclusions

This chapter investigated the performance of modular or cluster-based ANN models in modeling the complex eddy covariance-measured actual evapotranspiration. Various cluster-based models using supervised and unsupervised clustering techniques have been developed and compared with traditional global feedforward ANN models and the Penman–Monteith equation. It has been found that the unsupervised clustering-based ANN model, named spiking modular neural networks, outperformed other models in modeling the complex hydrologic process of evapotranspiration. The results of the analysis presented in this chapter help highlight the importance of developing efficient clustering techniques for hydrologic processes. When the input space that is used to estimate evapotranspiration is perfectly partitioned, a significant improvement in the prediction accuracy is achieved. It is recommended that more attention should be diverted toward a consideration of clustering techniques. Addressing the issue of clustering will not only help improve the prediction accuracy but contribute toward bridging the gap between mechanistic and data-driven hydrologic models.

Acknowledgements The financial support of the Natural Science and Engineering Research Council (NSERC) of Canada through its Discovery Grant Program is greatly appreciated. The second author thanks the Department of Civil & Geological Engineering at the University of Saskatchewan for its support through the Departmental Scholarship Fund.

References

Abrahart RJ (1999) Neurohydrology: implementation options and a research agenda. Area 31(2): 141–149

Abrahart RJ, See L (2000) Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecasts in two contrasting catchments. Hydrological Processes 14(11): 2157–2172

ASCE Task Committee on Artificial Neural Networks in Hydrology (2000) Artificial neural networks in hydrology, I, Preliminary concepts. Journal of Hydrologic Engineering 5(2): 115–123

Bastarache D, El-Jabi N, Turkkan N, Clair TA (1997) Predicting conductivity and acidity for small streams using neural networks. Canadian Journal of Civil Engineering 24: 1030–1039

Booy C, Morgan DR (1985) The effect of clustering of flood peaks on a flood risk analysis for the red river. Canadian Journal of Civil Engineering 12: 150–165

Bowden GJ, Dandy GC, Maier HR (2005) Input determination for neural network models in water resources applications. Part 1 – background and methodology. Journal of Hydrology 301: 75–92

Coulibaly P, Anctil F, Aravena R, Bobee B (2001) Artificial neural network modeling of water table depth fluctuations. Water Resources Research 37(4): 885–896

Dawson CW, See LM, Abrahart RJ, Heppenstall AJ (2006) Symbiotic adaptive neuro-evolution applied to rainfall-runoff modeling in northern England. Neural Networks 19: 237–247

Demuth H, Beale M (2001) Neural Network Toolbox Learning. For Use with MATLAB. The Math Works Inc: Natick, Mass.

Elshorbagy A, Panu US, Simonovic SP (2001) Analysis of cross-correlated chaotic streamflows. Hydrological Sciences Journal 46(5): 781–793

Elshorbagy A, Simonovic SP, Panu US (2000a) Performance evaluation of artificial neural networks for runoff prediction. Journal of Hydrologic Engineering, ASCE 5(4): 424–427

Elshorbagy A, Panu US, Simonovic, SP (2000b) Group-based estimation of missing hydrological data: I. approach and general methodology. Hydrological Sciences Journal 45(6): 849–866

French MN, Krajewski WF, Cuykendall RR (1992) Rainfall forecasting in space and time using a neural network. Journal of Hydrology 137: 1–31

Haykin S (1999) Neural Networks: A Comprehensive Foundation, 2nd ed. MacMillan: New York.

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Computation 9: 1735–1780

Hong Y, Hsu K-L, Sorooshian S, Gao X (2005) Self-organizing nonlinear output (SONO): A neural network suitable for cloud patch–based rainfall estimation at small scales. Water Resources Research 41(2) doi:10.1029/2004WR003142

Hornik K, Stinchcombe M, White H (1989) Multilayer feedforward networks are universal approximators. Neural Networks 2(5): 359–366

Hsu K-L, Gupta HV, Gao X, Sorooshian S, Imam B (2002) Self-organizing linear output map (SOLO): An artificial neural network suitable for hydrologic modeling and analysis. Water Resources Research 38(12): 10.1029/2001WR000795

Hsu KL, Gupta HV, Sorooshian S (1995) Artificial neural network modeling of the rainfall–runoff process. Water Resources Research 31(10): 2517–2530

Jain A, Srinivasulu S (2004) Development of effective and efficient rainfall-runoff models using integration of deterministic, real-coded genetic algorithms and artificial neural network techniques. Water Resources Research 40(4): 10.1029/2003WR002355

Jarvis PG, McNaughton KG (1986) Stomatal control of transpiration: Scaling up from leaf to region. Advances in Ecological Research 15: 1–49

Jayawardena AW, Fernando DAK (1998) Use of radial basis function type artificial neural networks for runoff simulation. Computer-Aided Civil and Infrastructure Engineering 13(2): 91–99

Karunanithi N, Grenney WJ, Whitley D, Bovee K (1994) Neural networks for river flow prediction. The Journal of Computing in Civil Engineering, ASCE 8(2): 201–220

Kohonen T (1989) Self-Organization and Associative Memory Springer-Verlag: New York

Lakshmi V, Susskind J (2001) Utilization of satellite data in land-surface hydrology: Sensitivity and assimilation. Hydrologic Processes 15(5): 877–892

Leuning R, Judd MJ (1996) The relative merits of open- and closed-path analysers for measurements of eddy fluxes. Global Change Biology 2: 241–253.

MacKay DJC (1992) Bayesian Methods for Adaptive Models. Ph.D. Thesis, California Institute of Technology

Maier HR, Dandy GC (2000) Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. Environmental Modelling and Software 15: 101–124

Moody J, Yarvin N (1992) Networks with learned unit response functions. In: Moody JE, Hanson SJ, Lippmann RP (eds), Advances in Neural Information Processing Systems 4. Morgan Kaufmann: San Mateo, CA

Panu US, Unny TE (1980) Stochastic synthesis of hydrologic data based on concepts of pattern recognition, I. General methodology of the approach. Journal of Hydrology 46: 5–34

Panu US, Unny TE, Ragade RK (1978) A feature prediction model in synthetic hydrology based on concepts of pattern recognition. Water Resources Research 14(2): 335–344

Parasuraman K, Elshorbagy A (2005) Wavelet networks: An alternative to neural networks. Proceedings of International Joint Conference on Neural Networks, IEEE Catalog Number: 05CH37662C, 2674–2679

Parasuraman K, Elshorbagy A (2007) Cluster-based hydrologic prediction using genetic algorithm-trained neural networks. Journal of Hydrologic Engineering, ASCE, 12(1): 52–62

Parasuraman K, Elshorbagy A, Carey SK (2006) Spiking-modular neural networks: A neural network modeling approach for hydrological processes. Water Resources Research 42: W05412, doi:10.1029/2005WR004317

Roiger R, Geatz M (2002) Data Mining: A Tutorial Based Primer. Addison Wesley: Reading, Mass.

Schotanus P, Niewstadt FTM, De Bruin HAR (1983) Temperature measurement with a sonic anemometer and its application to heat and moisture fluxes. Bound-Layer Meteorology 26: 81–95

Shi J (2002) Clustering technique for evaluating and validating neural network performance. Journal of Computing in Civil Engineering 16(2): 152–155

Sudheer KP, Jain A (2003) Radial basis function neural networks for modeling stage-discharge relationship. Journal of Hydrologic Engineering 8(3): 161–164

Wang J, Salvucci GD, Bras RL (2004) An extremum principle of evaporation, Water Resources Research 40: W09303, doi:10.1029/2004WR003087

Webb EK, Pearman GI, Leuning R (1980) Correction of flux measurements for density effects due to heat and water vapour transfer. Quarterly Journal of The Royal Meteorology Society 106: 85–100

Zhang B, Govindaraju RS (2000) Prediction of watershed runoff using Bayesian concepts and modular neural networks. Water Resources Research 36(3): 10.1029/1999WR900264

Chapter 29
Applications of Soft Computing to Environmental Hydroinformatics with Emphasis on Ecohydraulics Modelling

Q. Chen and A. Mynett

Abstract Owing to the high complexity and the non-linearity of ecosystems and the rapid development of technology in spatial data survey, soft computation is more and more widely used in ecohydraulics models. Soft computing is a broad field which includes cellular automata (CA), individual-based models (IBM) and box-based models with respect to paradigms, and artificial neural networks (ANN), fuzzy logic (FL), genetic algorithms (GA), chaos theory and rule-based method with respect to techniques. This chapter will concentrate on the use of CA and rule-based techniques to ecohydraulics models. Application cases include modelling of population dynamics, algal bloom forecasting and aquatic ecosystem succession.

Keywords Soft computation · cellular automata · rule-based methods · ecohydraulics models

29.1 Introduction

Limited by computation capacity and available ecological survey data, most of the conventional ecohydraulics models are aggregated and conceptually based. For example, in the Lotka–Volterra model, populations (May, 1975) are expressed by biomass instead of the number of individual species. In the Michaelis–Menten growth model, the Monod curve is taken as the basic concept (Jørgensen, 1994). These models are usually based upon Newton's second law of motion and the first and second laws of thermodynamics, and employ partial difference equations (PDFs) to describe physical processes (Abbott and Minns, 1998). The variables of the models are continuous in time and space, and the formulations strictly follow

Q. Chen
RCEES Chinese Academy of Sciences (Shuangqinglu 18, Haidian District, Beijing 100085)

A. Mynett
WL | Delft Hydraulics (Rotterdamsweg 185, 2600 MH Delft, The Netherlands)

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

the conservations of mass, momentum and energy. Such model paradigms and techniques have been playing important roles in the progress of ecological research and are still fundamental tools (Jørgensen, 1994).

However, the aggregated models mostly fail to take into account the effects of individual difference, spatial heterogeneity or local interactions. They sometimes do not even contain any spatial information, e.g. the Lotka–Volterra (LV) model; yet these properties can be crucial to the ecosystem dynamics (DeAngelis and Gross, 1992; Chen, 2004).

With respect to processes, it has been widely recognized that knowledge of the mechanisms of ecosystem dynamics is usually very limited owing to high complexity and non-linearity (Recknagel et al., 1994; Recknagel, 1997; Lee et al., 2003). In addition, the majority of the understanding is qualitative rather than quantitative that is difficult to formulate in PDFs (Chen, 2004).

Following the rapid development of computational power, in particular high-performance computation, detailed simulations have become cheaper and more available. The widespread use of advanced survey technologies makes it possible to implement large-scale and high-resolution data collection. Accordingly, more emerging techniques characterized by soft computing have been developed and applied in broad disciplines, and new interdisciplinary subjects such as ecoinformatics and hydroinformatics (Abbott, 1991) have even been established.

Soft computing is a broad field. Regarding paradigms, it consists of cellular automata, individual-based and box-based schemes. Cellular automata schematize the space into a regular lattice according to the principal spatial scale of the studied system. Each cell has some properties and takes a value from finite states. The value is updated in discrete time steps through the predefined local evolution rules that are functions of the current state of the cell itself and its neighbouring cells. Cellular automata are similar to the Eulerian approach in classical fluid mechanics and are suitable in simulating plant dynamics.

An individual-based model takes each species as the studied object to describe its properties (age, gender, size, etc.) and actions. It is similar to a Lagrangian approach in classical fluid mechanics and is advantageous in simulating species with spontaneous motions such as fish and animals.

Box-based models divide the studied space into boxes based on their different behaviour or mechanisms. The boxes are usually different in geometry. Within one box the system is considered homogeneous. Each box implements its own dynamical equations. Communication (mass and energy flows) takes place from box to box instead of individual species.

Compared to conventional model paradigms, they are usually discrete in time, space and model variables. They take each spatial unit or individual species as the target to investigate the evolution in time and the motion in space to obtain the global patterns of the system.

With respect to techniques, soft computation can be characterized by artificial neural networks, fuzzy logic, evolutionary algorithms and genetic programming. In comparison to conceptual methods, these techniques usually take empirical knowledge as a reference, and discover new rules from the collected data which are

used to supplement the insufficiency of our understanding (Weiss and Indurkhya, 1998; Witten and Frank, 2000).

Soft computing is also widely used in ecohydraulics research. Chen and Mynett (2003b) applied a CA to model prey–predator population dynamics and analysed the related system stabilities. Chen et al. (2002) used CA to simulate the competitive growth and colonization of two underwater macrophytes and explained the resulting ecosystem succession in Lake Veluwe, the Netherlands. Morales-Chaves (2004) investigated the growth and spread of zebra mussels in the Mississippi River by individual-based models. Salski and Sperlbaum (1991) and Chen and Mynett (2003a) successfully applied fuzzy logic to model eutrophication, and Recknagel et al. (1994) used ANN to forecast algal blooms. Baptist (2005) derived the vegetation-induced roughness formula from experimental data by genetic programming. Meanwhile, applications of chaos theory in ecohydraulics modelling are under rapid development as well (Jørgensen, 1994).

Despite these contributions, the application of soft computing in the ecohydraulics field is still at an infant stage compared to other fields. It is expected that soft computing characterized by discrete paradigm and data mining will play more and more important roles in ecosystem studies (Yue, 2003; Yue and Fan, 2004; Li and Ma, 2004). This chapter will concentrate on the application of the CA paradigm and rule-based technique to ecohydraulics modelling.

29.2 Cellular Automata Model Paradigm

Cellular automata were first proposed by Von Neumann (1949), and the computational theory was established by Wolfram in the 1980s (Wolfram, 1984a,b). The research has been almost entirely oriented towards the mechanism of self-organization and self-repair, until the development of Conway's Game of Life, which contains some ecological relevance.

Cellular automata are discrete mathematical systems which consist of a regular lattice of sites (cells or automata). Each site has properties and states. The states are updated in discrete time steps according to local evolution rules, ϕ, which are functions of the state of the cell itself and its neighbours. The cell size and time step are determined by the principal time and space scales of the studied ecosystem. CA often exhibit "self-organizational" behaviour. Even starting from complete disorder, the simple components can act together to produce complicated patterns of behaviour and their evolution can spontaneously generate an ordered structure. The evolution of CA is usually irreversible.

Figure 29.1 illustrates a one-dimensional and a two-dimensional CA system with the nearest neighbours, respectively. The corresponding rules of evolution are expressed as (29.1) and (29.2). The rules can be deterministic, stochastic or empirical (Chen et al., 2002). According to the definition, a CA system can be characterized by:

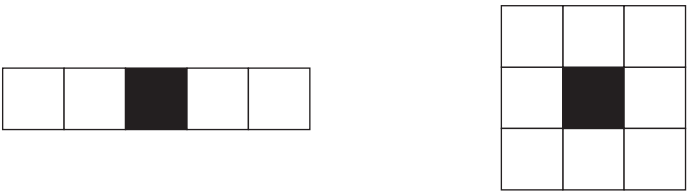


Fig. 29.1 Cellular automata (*left*: 1D, *right*: 2D)

1. Parallelism, which means all cell states are updated simultaneously
2. Homogeneity, which states that all cells follow the same evolutionary rules
3. Locality, which implies a cell can only gather information from its nearest neighbours and can only affect its direct neighbours

$$a_i^{t+1} = \phi(a_{i-1}^t, a_i^t, a_{i+1}^t) \tag{29.1}$$

$$a_{i,j}^{t+1} = \phi(a_{i,j}^t, a_{i-1,j-1}^t, a_{i-1,j}^t, a_{i-1,j+1}^t, a_{i,j-1}^t, a_{i,j+1}^t, a_{i+1,j-1}^t, a_{i+1,j}^t, a_{i+1,j+1}^t) \tag{29.2}$$

Similar to an individual-based paradigm, CA have inherent fusion with object-oriented programming technology so that it is easily implemented in parallel computation. Therefore, CA can be broadly applied to simulations of large-scale ecosystems (Engelen et al., 1993; Wootton, 2001). The following section will demonstrate the development of EcoCA, a model applying CA to simulate population dynamics of a prey–predator system.

EcoCA (Minns et al., 2000; Chen and Mynett, 2003b) is a two-dimensional stochastic CA model which has three different cell states: empty, prey and predator. The state of each cell ($a_{i,j}^t$) is exclusive; this means that at each time step, only one of the three states can exist in one cell. The evolution is based on the current cell state, the number of cells that are occupied by a predator (N_{pd}) and the number of the neighbouring cells occupied by a prey (N_{py}). The evolution rules ϕ define the probability that a cell will become prey (P_{py}) or predator (P_{pd}) or be empty (0) at the next time step. These evolution rules take into account reproduction, mortality, food availability, loneliness and overcrowding (29.3). After the probability is calculated, a random selection process is used to determine the transition of the cell state.

$$P = \phi(a_{i,j}^t, N_{pd}, N_{py}) \tag{29.3}$$

Figure 29.2 shows two snapshots from an EcoCA simulation and Fig. 29.3 presents a time series of the two populations and the phase dynamics. Because of the shortage of monitored population data, the Lotka–Volterra model (29.4) was calibrated by the outputs of EcoCA and the results (Fig. 29.4) are then used to evaluate the performance of EcoCA:

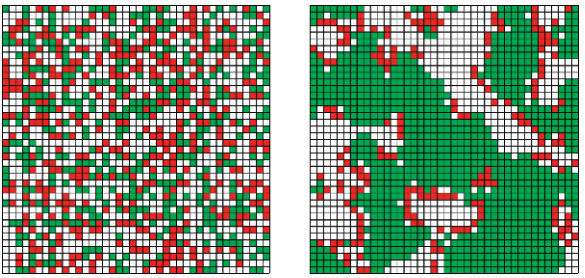


Fig. 29.2 Snapshots of the spatial patterns of prey–predator population dynamics simulated by EcoCA (*left*: initial condition; *right*: $t = 600$)

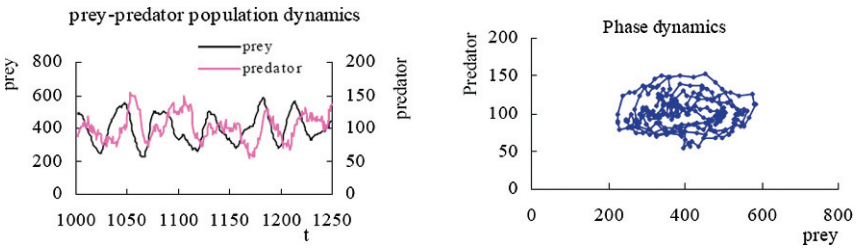


Fig. 29.3 Population dynamics and phase dynamics simulated by EcoCA

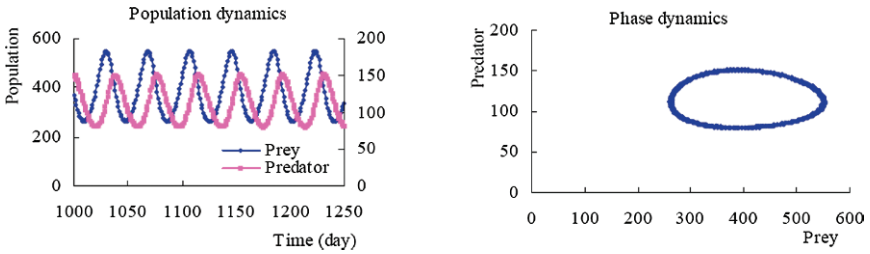


Fig. 29.4 Population dynamics and phase dynamics simulated by the LV model

$$\frac{\mathrm{d}P}{\mathrm{d}t} = aP - bP^2 - \alpha PQ \qquad (29.4)$$
$$\frac{\mathrm{d}Q}{\mathrm{d}t} = -cQ + \beta PQ$$

in which P = biomass of prey; Q = biomass of predator; a = growth rate of prey; b = carrying capacity; c = mortality rate of predator; α = functional response of prey on predator; and β = functional response of predators to prey. The cyclic behaviour is purely dependent on the initial conditions and the parameters.

It can be seen that although started from random initial conditions, the system shows strong self-organizational properties and produces eminent spatial patches after evolution (Fig. 29.2). The population dynamics are not as strictly periodic as in the Lotka–Volterra model (Fig. 29.4), but the cyclic behaviours are well captured,

as shown in Fig. 29.3. Drifting is constantly induced between orbits in contrast to the phase trajectories of the LV model. This is because of the embedded stochastic process in the EcoCA model, which is completely absent in the LV model. Therefore, it can be concluded that the EcoCA model can reproduce the overall behaviours of the LV model well.

It can be seen from Fig. 29.2 that the EcoCA model not only simulates the total population, but also provides their spatial distributions. Such information has been acknowledged to be lacking in the LV model because (29.4) contains no space variable. This means that the LV model cannot be used to analyse the spatial dynamics of ecosystems.

However, research has proved that spatial configuration is extremely important to habitat networks and ecosystem stability (Duel et al., 2002a,b; Baptist, 2005). Chen and Mynett (2003b) applied the EcoCA model to investigate the stabilities of a prey–predator system, and discovered that it is determined not only by critical population size, but also by their spatial distribution. Even with the same population size, different spatial configurations can lead to coexistence, predator extinction and extinction of both. In comparison, the LV model can only analyse the stabilities in line with critical population size.

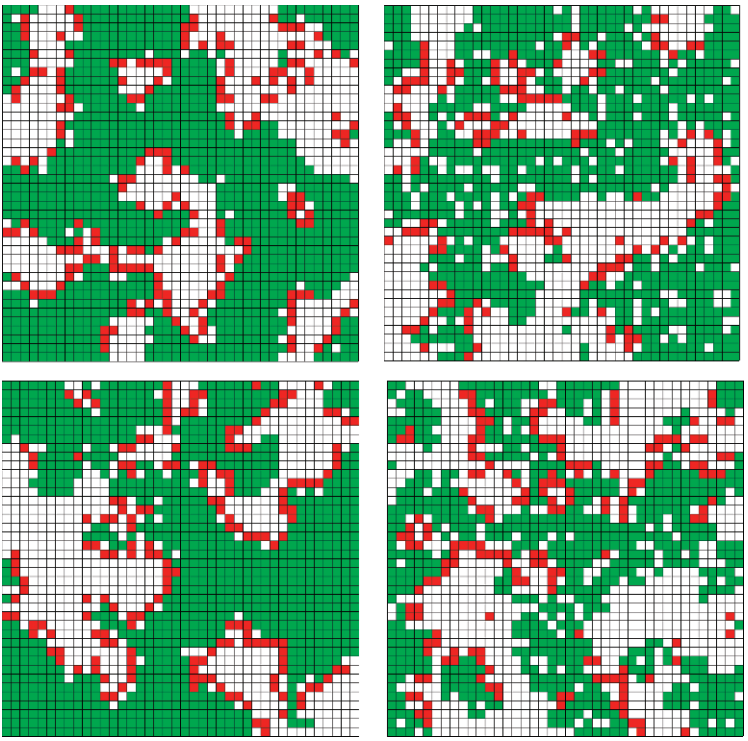


Fig. 29.5 Patterns of population dynamics under different harvest strategies (a: top-left, b: top-right, c: bottom-left, d: bottom-right; $t = 300$)

It is known that the harvest strategy is very important to population management such as sustainable fishing and aquatic culture. Therefore the EcoCA model has been used to study the different harvest schemes and four scenarios were simulated: (a) no harvest, (b) 15% harvest of prey, (c) 15% harvest of predator and (d) 15% harvest of both species. The criterion to execute a harvest is initiated when the population density is higher than 25%; otherwise harvest ceases at this time step. Figure 29.5 gives the simulation results of these four strategies. From the simulations and the analyses of population time series, it is found that a reasonable harvest of both species can increase their productivities, hence the whole system outputs, and enhance the system stability. The worst scenario is to harvest only the prey. The management strategy can be ranked as d > a > c > b, which cannot be achieved through the LV model (Azar et al., 1995).

29.3 Rule-Based Model Technique

A rule-based technique is a method that falls in between fuzzy logic and data mining. It takes experts' knowledge as the primary reference to set up the basic rules, and then supplements it with new rules discovered from the available data. The rule-based method has attracted great interest within the aquatic ecosystem modelling community. Chen and Mynett (2003a; 2004) have applied the rule-based technique to model algal blooms in the Taihu Lake, China and the North Sea. Baptist et al. (2002) used the method to simulate vegetation succession in the floodplain of the lower River Rhine. Due to the difficulty in describing the functional (physiological) responses of species to environmental factors, rule-based techniques can be a suitable method for habitat modelling (Jorde et al., 2001; Duel et al., 2002).

There are usually three main steps to build up the rules: defining the characteristic values of the variables, formulating the evolution rules and compacting the rules. The characteristic values can be defined according to experimental results or data analysis. In the $P-I$ curve (Fig. 29.6) obtained from an experiment: when $I \leq I_1$, irradiance is limited; when $I \geq I_2$, irradiance is toxic (photoinhibition). Therefore, I_1 and I_2 can be the characteristic values for variable I (Chen, 2004). However, in most cases, there is a lack of such experimental results, whilst some field-monitored data may be available. It is possible to apply partitioning analysis or cluster analysis to derive the characteristic values (MacQueen, 1967; Chen, 2004). The right-hand side of Fig. 29.6 shows a partitioning analysis of inorganic nitrogen and phosphate data collected from the Noodwijk 20 station (Chen and Mynett, 2004) in the North Sea; the mean value of each group can be taken as the characteristic value.

Step two is the formulation of the evolution rules. To do this, expert knowledge in the format of model rules must first be structured and these rules will serve as the basic rule base. New rules will be derived through feature reasoning (Chen and Mynett, 2002) and case reasoning (Chen and Mynett, 2004). For feature reasoning, clustering analysis is applied and each cluster represents a rule (Boogaard et al., 1998; Chen and Mynett, 2003a). For case reasoning, each record

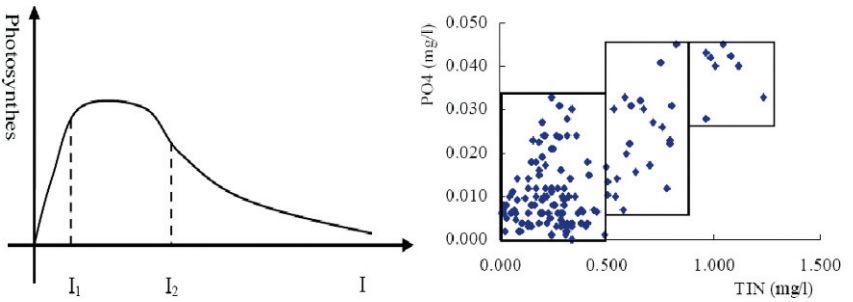


Fig. 29.6 Methods to determine characteristic values of model parameters (*left*: experiment, *right*: partitioning analysis)

is transformed into an intermediate rule, and rules that conflict with the basic rule base are then eliminated. After that, insignificant rules as determined by statistical analysis are removed. Finally, the remaining intermediate rules are weighted according to their reoccurrence frequencies and then added into the basic rule base.

The final step is to compact the rule base. The rule base increases exponentially with the number of variables and the classes of each variable. This increases the complexity of the model and the difficulty of calibration. A large rule base is also hard to interpret, which results in losing the original advantages of the rule-based technique. To compact the rule base obtained in step 2, combination of similar rules and increase of significance criterion are applied (Krone and Taeger, 2001).

The following case study is used to illustrate the application of the rule-based technique to algal bloom modelling. Affected by the inflow from the River Rhine, vernal algal blooms dominated by diatom and *Phaeocystis globosa* (*P. globosa*) occur often in the Dutch coastal waters. The blooms are not toxic, but are harmful because of the resultant loss of bivalves and recreation. Funded by the EU FP5 program Harmful Algal Blooms Expert System (HABES), a rule-based model was developed (Blauw, 2004; Chen and Mynett, 2004, 2006a,b).

According to the lab experiments and field monitoring, it was initially estimated that nutrient and light requirements for colonial blooms of *P. globosa* were > 0.0062 mg/l inorganic phosphorus and $> 100\,\mathrm{Wh/m^2\ day^{-1}}$ irradiance (Peperzak et al., 2002). Relatively high growth rates ($> 0.5\,\mathrm{day^{-1}}$) seem to occur at salinity levels of 20–35 psu and a temperature range of 7–22°C (Riegman et al., 1992). The diatom is the competing species which is mainly governed by the availability of silica.

By integrating experts' knowledge and the analysis of data from 1990–2000, a rule-based model was set up for forecasting *P. globosa* blooms (Chen and Mynett, 2006a), which is presented in Table 29.1.

Here Z is the mixing depth that can be computed by a 1D vertical turbulence model (Baptist, 2005) and Z_{cr} is the critical depth that is determined by instant irradiance (I_0). The attenuation coefficient (k_d), the maximum growth rate (u_{max}), the loss rate (l) and the half-saturation irradiance (k_s) can be calculated according to (29.5) by Newton's method. D is the turbulence diffusion coefficient and D_{cr} is the critical turbulence

Table 29.1 Rule-based foresting model for *P. globosa* bloom

Rule 1	Rule 2	Rule 3	Rule 4	Default
$Z > Z_{cr}$	$Z <= Z_{cr}$	$T >= 14.64$	$T >= 10.24$	N
$D < D_{cr}$	$D < D_{cr}$	$PO_4^{3-} <= 0.42$	$S_iO_2 > 1.17$	
$S_iO_2 > 1.17$	$T <= 10.24$	$S_iO_2 <= 1.17$	$PO_4^{3-} > 0.42$	
Y = [0.947]*	$PO_4^{3-} >= 0.42$	Y = [0.650]	Y = [0.958]	
	Y = [0.667]			

* The values in the brackets are the probabilities; Y = bloom, N = not bloom

diffusion given by (29.6). S_iO_2 is the silica concentration, T is water temperature and PO_4^{3-} is phosphate concentration. The model has an accuracy of 78%:

$$f(z_{cr}) = \frac{I_0(1 - e^{-k_d z_{cr}})}{k_d z_{cr}} - \frac{k_s l}{u_{max} - l} \tag{29.5}$$

$$D_{cr} < \frac{I_0^2 (u_{max} - l)^2 (u_m - l)}{\pi^2 k_s^2 l^2 k_d^2} \tag{29.6}$$

Since the mixing depth is mainly determined by tide, irradiance and wind, it shows strong seasonality. It is therefore pre-computed and archived in a lookup table. The few physical and chemical parameters needed by the model can be monitored online with a buoy so the derived rule-based model can be used in real-time operation.

29.4 Rule-Based Cellular Automata for Modelling Algal Blooms

The dynamics of the EcoCA model is purely dependent on geometric relations between neighbouring cells, and the rules of the EcoCA do not account for any physical or biological processes. The rule-based model for *P. globosa* is developed on a point data set, which cannot explain large-scale phenomena. There are many systems, in particular ecosystems where the dynamical behaviour is operating at a large scale but the detailed mechanisms and their statistical properties remain unclear. For these reasons, the rule-based technique and cellular automata paradigm are integrated to investigate the phenomena where local interactions play a significant role in global patterns. The following case study demonstrates the application of such rule-based cellular automata (RCA) to model algal bloom in the Dutch coast.

29.4.1 Description of the Study Area

The study focuses on the near-shore area of the Dutch coast (Fig. 29.7). The water depth is between 0 and 30 m, and the water temperature varies from 5 to 22°C,

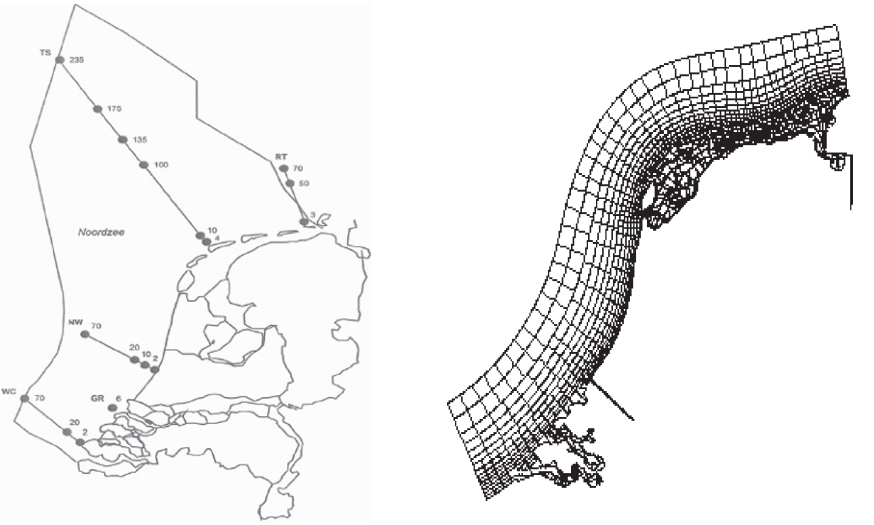


Fig. 29.7 Study area and monitoring stations in the Dutch coast (*left*) and the computation grid (*right*)

and the irradiance is between 132 and 1700 $\mathrm{Wh/m^2day^{-1}}$. The concentrations of inorganic nitrogen and phosphorus are between 0.007 and 1.246 mg/l and 0 and 0.073 mg/l, respectively. The biomass concentration (in chlorophyll *a*) is from 0.1 to 90.2 μg/l. The discharge from the River Rhine at the Maassluis station is between -2744 and 4649 $\mathrm{m^3/s}$, with a mean of 1382 $\mathrm{m^3/s}$. The water is usually well mixed except for temporary weak stratification caused by salinity. The RCA model is to forecast algal bloom (defined by chlorophyll $a \geq 30$μg/l) basing on the monitored irradiance data and the nutrient concentrations data computed by Delft3D-WAQ, which is a well-developed three-dimensional water quality model from WL | Delft Hydraulics.

29.4.2 Model Development

A curvilinear grid (Fig. 29.7) is used in the model and the calculation of nitrate and phosphate concentrations was realized through the processes in the library configuration tool (PLCT) of the Delft3D-WAQ (Chen, 2004). The boundary conditions are provided by the monitored data from the stations (Fig. 29.7) as a block function where a constant value is given between two consecutive observations. The initial conditions were configured through linear interpolation of the monitored data.

The rule-based model developed by Chen and Mynett (2004) was introduced to predict algal biomass on the basis of the calculated nutrient concentrations from the Delft3D-WAQ. The membership functions of nitrate and chlorophyll *a* (Chla)

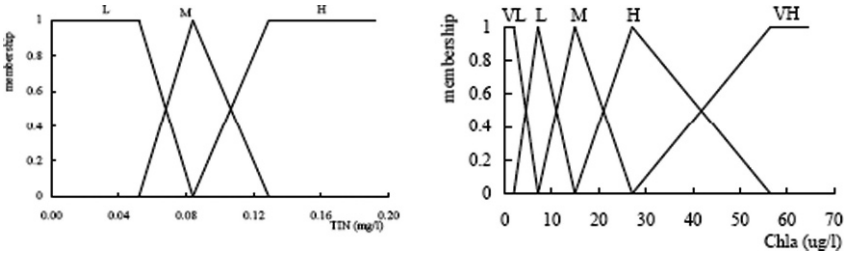


Fig. 29.8 Membership function of model variable and output (*left*: total inorganic nitrogen, *right*: chlorophyll *a*)

concentrations are shown in Fig. 29.8. The other variables include mean water column irradiance, water temperature and ortho-phosphate concentration. Chlorophyll *a* concentration at the last time step (Chla_{t-1}) is also used as the model input.

The time step (Δt) for hydrodynamic computation was 5 min, and the simulation completed a full tidal cycle, which was then repeatedly used for a year. The Δt was aggregated into 7 days in the rule-based model for algal biomass estimation.

The cellular automata module was directly implemented on the curvilinear grid, which of course did not strictly follow the original definition of CA where square grids are used. However, this approximation can be acceptable as the geometry of the cells does not have much variation in the nearest neighbours. The Moore neighbourhood configuration (Wolfram, 1984b) was applied in the CA model and the local evolution rules were formulated in general as

$$S_{i,j}^{t+1} = f\left({}^{*}S_{i,j}^{t+1}, \sum {}^{*}S_{N}^{t+1}\right) \tag{29.7}$$

where $S_{i,j}^{t+1}$ is the state of cell (i, j) at time step $t+1$, ${}^{*}S_{i,j}^{t+1}$ is the state of the cell (i, j) at time step $t+1$ which is initially estimated without local interactions, $\sum {}^{*}S_{N}^{t+1}$ is the preliminarily estimated states of the eight neighbouring cells and f are local evolution rules. In this study, the state S takes the value of the set $S_{\mathrm{Chl-a}} \in (\mathrm{L}, \mathrm{M}, \mathrm{H})$. Supposing ${}^{*}S_{i,j}^{t+1} = p$, the rules f are defined as

$$S_{i,j}^{t+1} = \begin{cases} p & \text{if over 3 neighbours hold} {}^{*}S_{\mathrm{neighbour}}^{t+1} = p \\ 0.5(p+q) & \text{if over 3 neighbours hold} {}^{*}S_{\mathrm{neighbour}}^{t+1} = q \end{cases} \quad p, q \in S_{\mathrm{Chl-a}} \tag{29.8}$$

29.4.3 Model Results

Some of the modelled results of chlorophyll *a* concentrations in year 1995 are given in Fig. 29.9 which presents the output at peak-bloom period. The spatial pattern shows that the algal blooms are centred on or near the shore area. The reason is that the residual flow of river discharge is from the South to the North following the coastal line due to the effects of the Coriolis force, so the nutrient concentrations are

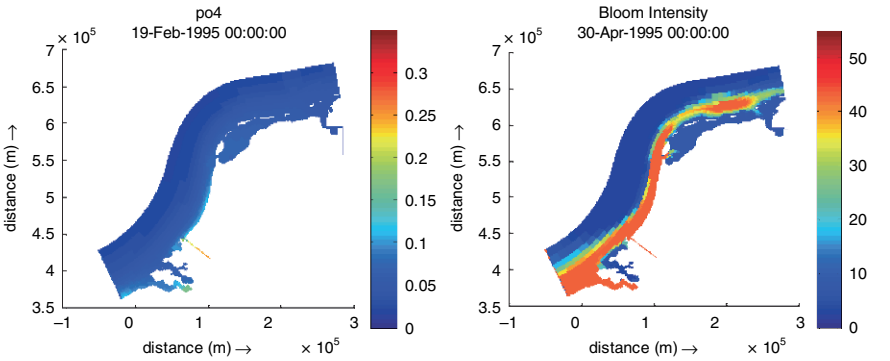


Fig. 29.9 Model results of phosphate concentrations before algal bloom by Delft3D-WAQ and algal biomass concentrations by RCA

higher along the coast. It can also be seen that the blooms are more severe near the Noordwijk transect and the Wadden Sea area because of the discharge from the land.

By examining the observations in 1995, the first peak bloom at the station Noordwijk 10 (Fig. 29.7) appeared on 3 May with a chlorophyll *a* concentration of 58.2 µg/l. The modelled bloom timing (30 April) and intensity (48 µg/l) are very close to the observations.

To further evaluate the RCA mode, the computed chlorophyll *a* concentrations along the Noordwijk transect, where the most consistent and also the best data sets were available, during the bloom period in 1995 were compared with the observations in Fig. 29.10. In addition, the time series of chlorophyll *a* concentrations at the station Noordwijk 10 in 1995 from the observations and the RCA model are presented in Fig. 29.11.

With respect to the spatial variations of chlorophyll *a* concentration along the Noordwijk transect, the results of the RCA model agree well with the observations

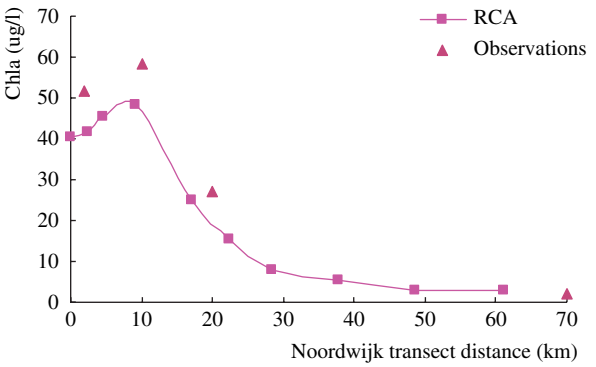


Fig. 29.10 Chlorophyll *a* concentrations along the Noodwijk transect in bloom period of 1995 from observations (3 May) and RCA model (30 April)

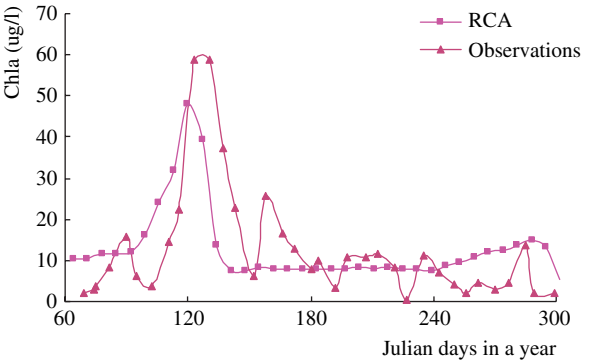


Fig. 29.11 Time series plots of chlorophyll *a* concentrations at the station Noordwijk 10 in 1995 from observations and RCA model

(Fig. 29.10). From the time series plot of algal biomass concentration at the station Noordwijk 10 (Fig. 29.11), it is seen that the RCA model is also able to capture the temporal variations.

For the time being, it is difficult to quantitatively evaluate the modelled spatial patterns. However, it could be possible in future to use satellite images for such comparison.

The development and application of rule-based cellular automata are still at the initial stages. However, the preliminary research outputs indicate that the cellular automata paradigm has enhanced the capture of patchy dynamics (Chen and Mynett, 2006b).

29.5 Discussion and Conclusions

Although soft computing is becoming more and more widely applied in ecohydraulics modelling, and already demonstrates certain advantages in some cases as shown by the examples in this chapter, it does not mean that soft computing can take over from conventional methods that are mainly characterized by physical-based description and partial differential equations. The key issue nowadays is to select proper methods from a variety of available tools according to the particular problems of interest (Chen and Ouyang, 2005a,b). In general, the selection of a paradigm or method depends on three main factors: the research objectives, understanding of the problem and the availability of data.

With respect to modelling techniques, if knowledge of the mechanisms is sufficient and data are limited, a conceptual model can be an appropriate choice. If knowledge is limited but enough data are available, data mining could be the most suitable method. When only a limited amount of both data and knowledge are available, rule-based methods taking experts' experience as a reference point is a suitable alternative (Duel et al., 2002; Lee et al., 2002).

Following the rapid development and widespread use of information technology in ecological research, large-scale and high-resolution ecosystem data are more easily available. Meanwhile, parallel computation makes detailed spatial simulation realistic. Therefore, spatially explicit models such as CA and IBM are largely facilitated by these developments and have become a promising research field in recent years. The future of ecohydraulics modelling lies in the integration of different paradigms and techniques at multiple and proper spatio-temporal scales.

Acknowledgements The research is jointly funded by the National Natural Science Foundation of China (No. 50639070), the "100 Talents Program" of Chinese Academy of Sciences, the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry, and WL | Delft Hydraulics.

References

Abbott MB (1991) Hydroinformatics – Information Technology and the Aquatic Environment. Avebury Technical, Aldershot, UK

Abbott MB, Minns AW (1998) Computational Hydraulics (2nd Edition). Ashgate Publishing Company, UK

Azar C, Holmberg J, Lindgern K (1995) Stability analysis of harvesting in a predator–prey model. Journal of Theoretical Biology, 174: 13–19

Baptist M J, Lee G E M, van der Kerle F (2002) Modelling of morphodynamics, vegetation development and fish habitat in man-made secondary channels in the river Rhine. In: King J (Ed.), *Proceedings of the 4th Ecohydraulics Conference and Environmental Flows*. Cape Town, South Africa

Baptist MJ (2005) Modelling floodplain biogeomorphology. Ph.D. thesis, Delft University of Technology

Blauw A N (2004) Final Report: Harmful Algal Bloom Expert System. EVK2-2000-22091

Boogaard H F P van den, Mynett A E, Ali Md S (1998) Self organisation feature maps for the analysis of hydrological and ecological data sets. In: Babovic, V., and Larsen, L.C., (eds.), Proceedings of Hydroinformatics 1998

Chen Q, Mynett AE, Minns AW (2002) Application of cellular automata to modelling competitive growth of two underwater species *C. aspera* and *P. pectinatus* in Lake Veluwe. Ecological Modelling, 147: 253–265

Chen Q, Mynett AE (2003a) Integration of data mining techniques with heuristic knowledge in a fuzzy logic modelling of eutrophication in Taihu Lake. Ecological Modelling, 162: 55–67

Chen Q, Mynett AE (2003b) Effects of cell size and configuration in cellular automata based prey-predator modelling. Simulation Modelling Practice and Theory, 11: 609–625

Chen Q, Mynett A E (2004) A robust fuzzy logic approach to modelling algal biomass. Journal of Hydraulic Research, 42: 303–309

Chen Q (2004) Cellular Automata and Artificial Intelligence in Ecohydraulics Modelling, Taylor & Francis Group plc, London UK

Chen Q, Ouyang Z (2005a) Watershed Ecology and Modeling System. Acta Ecologica Sinica, 25(5): 1153–1161

Chen Q, Ouyang Z (2005b) Integrated ecohydraulics model and the application. Journal of Hydraulic Engineering, 36(11): 1273–1279

Chen Q, Mynett A E (2006a) Forecasting phaeocystis globosa blooms in the Dutch Coast by an integrated numerical and decision tree model. Aquatic Ecosystem Health and Management, 9(3): 357–364

Chen Q, Mynett AE (2006b) Modelling algal blooms in the Dutch Coast waters by integrated numerical and fuzzy cellular automata approaches. Ecological Modelling, 199: 73–81

DeAngelis DL, Gross LJ (1992) Individual Based Models and Approaches in Ecology: Concepts and Models. Routledge, Chapman and Hall, New York

Duel H, Lee GEM van der, Penning WE (2002). Habitat Modelling of Rivers and Lakes in the Netherlands: An Ecosystem Approach, Canadian Water Resources Journal, 28: 231–248

Duel H, Baptist M J, Geerling G J (2002) Cyclic Floodplain Rejuvenation as a strategy for both flood protection and enhancement of the biodiversity of the river Rhine, In King J (Ed.), *Proceedings of the 4th Ecohydraulics Conference and Environmental Flows*. Cape Town, South Africa

Engelen G, White R, Uljee I (1993) Integrating Constrained Cellular Automata Models, GIS and Decision Support Tools for Urban Planning and Policy Making. In Timmermans, H. (Eds.), Design and Decision Support Systems in Architecture and Urban Planning, Kluwer Academic Publishers, Dordrecht.

Jorde K, Schneider M, Peter A (2001) Fuzzy based models for the evaluation of fish habitat quality and instream flow assessment. *Proceedings of the 2001 International Symposium on Environmental Hydraulics*, Arizona, USA

Jørgensen SE (1994) Fundamentals of Ecological Modelling (2nd Edition). Elsevier Science BV

Krone A, Taeger H (2001) Data-based fuzzy rule test for fuzzy modelling. Fuzzy Sets and Systems, 123: 343–358

Lee J H W, Huang Y, Dickman M, Jayawardena AW (2003) Neural network modelling of coastal algal blooms. Ecological Modelling, 159: 179–201

Lee G E M, Aarts H P A, Boogaard, H F P (2002) Analysis of uncertainty in expert rules and input data for habitat suitability models. In King J (Ed.), *Proceedings of the 4th Ecohydraulics Conference and Environmental Flows*. Cape Town, South Africa, 2002.

Li D, Ma Z (2004) Perspectives of Numerical Ecology and Ecological Modelling. Review and Prospects of Ecology Research. China Meteorological Press, Beijing, 89–99.

May RM (1975) Biological populations obeying difference equations: stable points, stable cycles and chaos. Journal of Theoretical Biology, 49: 511–524

MacQueen, J (1967) Some methods for classification and analysis of multivariable observation. *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, Vol. 1, pp. 281–297. Berkeley: University of California Press

Minns AW, Mynett AE, Chen Q (2000) Boogaard H F P van den. A cellular automata approach to ecological modelling. In Odgaard, A.J., (Ed.), Proceedings of Hydroinformatics Conference, Iowa, USA

Morales-Chaves Y (2004) Analysis of mussel population dynamics in the Mississippi River. Ph.D. Thesis, Civil and Environmental Engineering, University of Iowa, Iowa City

Peperzak L (2002) The Wax and Wane of *Phaeocystis globosa* Blooms. PhD dissertation, University of Groningen

Recknagel F, Petzoldt T, Jaeke O, Krusche F (1994) Hybrid expert system DELAQUA – a toolkit for water quality control of lakes and reservoirs. Ecological Modelling, 71: 17–36

Recknagel F (1997) ANNA-Artificial neural network model for predicting species abundance and succession of blue-green algae. Hydrobiologia, 349: 47–57

Riegman R, Noordeloos AAM, Cadee GC (1992) Phaeocystis blooms and eutrophication of the continental coastal zones of the North Sea. Marine Biology, 112: 479–484

Salski A, Sperlbaum C (1991) Fuzzy logic approach to modelling in ecosystem research, In: Bouchon–Meunier, B. Yager, R. and Zadeh, L. (Ed.), Uncertainty in Knowledge Bases, Springer LNCS 521

Von Neumann J (1949) Theory of self-reproducing automata, Univ. of Illinois Lectures on the Theory and Organization of Complicated Automata, Burks, A.W. (Ed.), University of Illinois Press, Urbana

Weiss SM, Indurkhya N (1998) Predictive Data Mining, A Practical Guide. Morgan Kaufmann Publishers, Inc., San Francisco, USA

Witten IH, Frank E (2000) Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations. Morgan Kaufmann Publishers, San Francisco, USA
Wolfram S (1984a) Computation theory of cellular automata. Communication in Mathematical Physics, 96: 15–57
Wolfram S (1984b) Cellular automata as model of complexity. Nature, 311: 419–424.
Wootton JT (2001) Local Interactions predict large-scale patterns in empirically derived cellular automata. Nature, 413: 841–844
Yue T (2003) Handbook of Numerical Models in Environment. Scientific Press, Beijing
Yue T, Fan Z (2004) Research and Perspectives on Ecological Model Development. Review and Prospects of Ecology Research. China Meteorological Press, Beijing, 80–88

Chapter 30
Data-Driven Models for Projecting Ocean Temperature Profile from Sea Surface Temperature

C.D. Doan, S.Y. Liong and E.S. Chan

Abstract Knowledge on tidal fluctuations and sea temperature profiles, in addition to other forcings, is absolutely essential in ocean model simulations. This chapter demonstrates the application of one of the data-driven techniques, genetic programming (GP), to sea temperature profile simulations. A GP model was trained to simulate sea temperature profiles based solely on corresponding sea surface temperatures. Training and verification results show that a trained GP could simulate observed data very well.

Keywords Genetic programming · sea surface temperature · South China Sea · boundary forcing · ocean circulation model

30.1 Introduction

With a large population concentrated along the shore of Southeast Asian waters, there is increasing concern over the environment in the region resulting from industrialization, for example. A good understanding of the complex dynamics of air motion, waves, currents and their interactions is significant for both scientific and practical reasons.

The Southeast Asian waters, Fig. 30.1, include the South China Sea, the Java Sea, the Sulu Sea and the Sulawesi Sea. It includes large shallow regions and deep trenches, and has a complex geometry and it is connected to deep marginal seas by some straits. The northeastern part of South China Sea adjoins a deep-sea basin up to 5000 m in depth. This basin is separated from the main body of the Pacific by

C.D. Doan
Tropical Marine Science Institute, National University of Singapore, Singapore 119223,

S.Y. Liong
Tropical Marine Science Institute, National University of Singapore, Singapore 119223,
e-mail: tmslsy@nus.edu.sg

E.S. Chan
Tropical Marine Science Institute, National University of Singapore, Singapore 119223,

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

Fig. 30.1 Southeast Asian waters and location of study area

the Luzon Strait and connected to the East China Sea through the Taiwan Strait. It is situated between the Asian continent, Borneo, the Philippines and Formosa (Taiwan). In the southern part, the Java Sea is joined to the Indian Ocean by the Lombok Strait, which is a shelf sea with depths less than 200 m. Other connections of waters include the Sulawesi Sea to the Pacific Ocean and Flores Sea to Banda Sea.

Based on limited observation data sets of the physical parameters of the water column, both cool and warm anomalies were detected in South China Sea. Dale (1956) and Nitani (1970) found a cool anomaly in the central Vietnamese coast and northwest of Luzon in summer; Soong et al. (1995) also detected a cool anomaly in central South China Sea in the winter period through analyses of TOPEX/POSEIDON data; the warm anomaly was also reported in the central South China Sea in late spring (Chu and Tseng, 1997) where most of the measurements are, in general, irregular in time and space.

With a series of limitations and the high cost of field measurements, numerical simulation models solving time-dependent flows have been demonstrated to be quite effective and relatively very economical. Numerical simulation models have proven to be an important tool in understanding seasonal ocean circulations, the seasonal thermal structures and in establishing a nowcast system for regional seas. There are already a series of studies that are capable of adequately resolving the Southeast Asian waters. Metzger et al. (1992) and Metzger and Hurlburt (1996) looked at the region and compared upper layer current and sea level from their global models with observation data; Murray et al. (1990) focused on observations and modelling in the vicinity of Lombok Strait; Masumoto and Yamagata (1991) investigated the current systems in the equatorial Pacific Ocean east of Philippines; Chu and Chang (1997, 1999) studied the seasonal thermodynamics in South China Sea using the Princeton Ocean Model (POM) with limited boundary conditions, a monthly mean climatological wind stress data set (Hellerman and Rosenstein, 1983) and bi-monthly variation of mass transport at the open boundaries (Wyrtki, 1961); Ningsih et al. (2000) simulated tide- and wind-driven circulation in the Java Sea. The wave–current interaction effect is also taken into account. The hydrodynamic results are employed to simulate the trajectory of water particles in the region. An in-house POM (Princeton Ocean Model) based ocean model, SEAOM (SouthEast Asian Ocean Model; Zhang and Chan, 2002), introduced several features such as surface wave stress and wave–current interaction terms. Recently SEAOM has been further extended to take better boundary input data, tidal time series and sea temperature data for initial input data and, later, for data assimilation.

The main objective of this chapter is to present the novel application of one of the data-driven techniques, genetic programming, GP (Koza, 1992; Langdon, 1998), to generate sea temperature profiles with relatively high performance accuracy.

30.2 Genetic Programming

Genetic programming is a member of the evolutionary algorithm family. Evolutionary algorithms (EA) are based upon Darwin's natural selection theory of evolution where a population is progressively improved by selectively discarding the not-so-fit population and breeding new children from a better population. EAs work by defining a goal in the form of a quality criterion and then use this goal to measure and compare solution candidates in a stepwise refinement of a set of data structures that returns an optimal or nearly optimal solution after a number of iterations. Evolutionary strategies (ES), genetic algorithms (GA) and evolutionary programs (EP) are three early variations of evolutionary algorithms whereas GP is a relatively recent variant (Liong et al., 2002)

GP is a domain-independent automatic programming technique evolving solutions to a given problem (Koza, 1992). The solution may be accurate or approximate depending upon the level of noise in the data set for which a solution is sought. It uses optimization techniques to evolve simple programs mimicking the way humans

construct programs by progressively rewriting them. Trial programs are repeatedly modified in the search for better or fitter solutions (Langdon, 1998).

The problem of finding a function in symbolic form, that fits a given set of data, is called symbolic regression. The aim of symbolic regression is to determine a functional relationship between input and output data sets. Symbolic regression is error-driven evolution and it may be linear, quadratic or higher order polynomial.

30.2.1 Basic Operations in Symbolic Regression

A population in GP consists of computer programs. To ease the process of creating new programs from two parent programs, the programs are represented by parse tree structures composed of a function set and a terminal set. The functions are mathematical or logical operators and terminals are constants and variables. The GP trees are dynamically modified by genetic operators in order to optimize its fitness value during the evolving process. Genetic operators include selection, crossover and mutation, and each operator is applied to selected individuals. Selection is the process of copying individuals into the next generation of the population probabilistically selected based on fitness value. Crossover operator interchanges randomly selected subtrees of each chosen parent pair to generate syntactically correct offsprings. In the mutation operation, a single parental program is probabilistically selected from the population based on fitness. Many types of mutations are possible and only two kinds are outlined here. In the first kind a function or a terminal can only be replaced by a function or a terminal. In the second kind an entire subtree can be replaced by another subtree generated by the same random process used to generate the initial population.

The following three examples serve to illustrate parse tree and GP operations. Figure 30.2 shows how an expression such as $\{\sqrt{(b^2-4ac)}-b\}/2a$ is represented as a parse tree. In Fig. 30.3 a crossover operation is demonstrated. It should be noted that bold parts of two parent trees in Fig. 30.3 representing

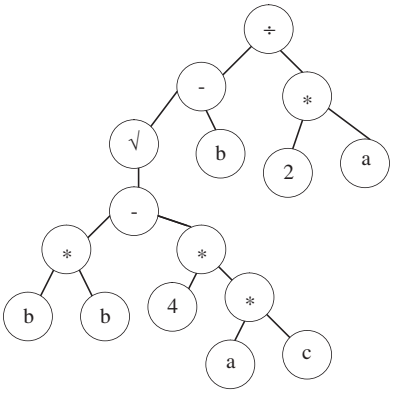
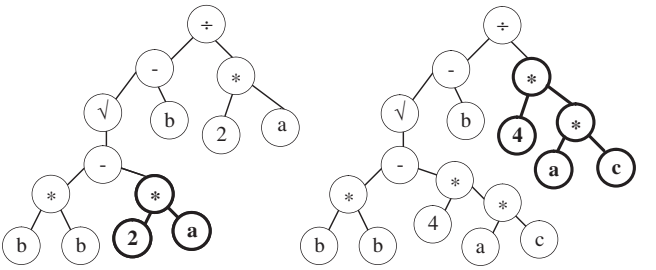
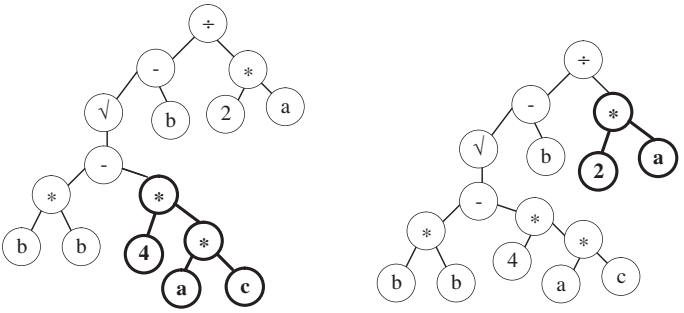


Fig. 30.2 GP tree representation of $\{-b+\sqrt{(b^2-4ac)}\}/2a$



Parent 1: $\{\sqrt{(b^2-2a)}-b\}/2a$ Parent 2: $(\sqrt{b^2}-4ac)/4ac$

(a) Parents



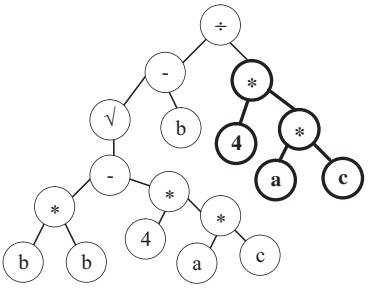
Child 1: $\{\sqrt{(b^2-4ac)}-b\}/2a$ Child 2: $(\sqrt{b^2}-4ac)/2a$

(b) Children

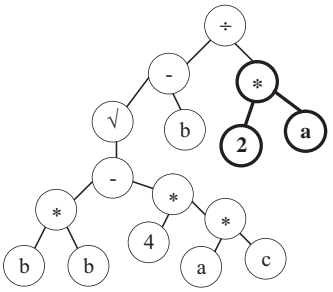
Fig. 30.3 Crossover operation in GP

expressions $\{\sqrt{(b^2-2a)}-b\}/2a$ and $(\sqrt{b^2}-4ac)/4ac$ (which are both incorrect) exchange each other to create two children representing two new expressions $\{\sqrt{(b^2-4ac)}-b\}/2a$ (the correct solution) and $(\sqrt{b^2}-4ac)/2a$. Figure 30.4 illustrates one possible mutation operation in GP. The bold-faced subtree "(4*a*c)" of a parent tree representing expression $\{\sqrt{(b^2-4ac)}-b\}/4ac$ is replaced by subtree "(2*a)" to form the correct expression $\{\sqrt{(b^2-4ac)}-b\}/2a$.

The search process in GP is guided by a fitness function. The definition of a fitness function is the most important aspect in GP and it depends upon the problem domain. The performance of GP largely depends upon how well the fitness function represents the objective or goal of the problem at hand. The main generational loop of a run of GP consists of the fitness evaluation, selection and the genetic operations discussed above. Each individual program in the population is evaluated to determine how fit it is at solving the problem. Programs are then probabilistically selected from the population based on their fitness to participate in the various genetic operations, with reselection allowed. While a fitter program has a better chance of being selected, even individuals known to be unfit are allocated some trials in a probabilistic way.



(a) Parent: $\{\sqrt{(b^2-4ac)}-b\}/4ac$



(b) Child: $\{\sqrt{(b^2-4ac)}-b\}/2a$

Fig. 30.4 Mutation operation in GP

30.3 Sea Temperature Profile Projected from Sea Surface Temperature

This section demonstrates the application of genetic programming to projecting the sea temperature profile based solely on sea surface temperature. The typical ocean temperature profile (in low to middle latitudes) is depicted in Fig. 30.5. The top part of the ocean is the surface layer (or mixed layer). In this layer, driven by wind and wave-breaking, the heat provided by the sun is well mixed. The middle part of the ocean is called the thermocline layer. In this layer, the temperature reduces rapidly with depth. The third part of the ocean is the deep layer. The deep ocean is not well mixed; it is made up of horizontal layers of equal density. Much of this deep ocean water is between 0 and 3°C.

A temperature profile is required as the boundary and initial conditions for the hydrodynamic ocean model simulation. However, apart from sea surface temperature (SST), which can be obtained from the orbiting satellite, the rest of the profile is not available unless they happened to be measured. This chapter investigates the application of GP to simulating the ocean temperature profile from SST for a given location.

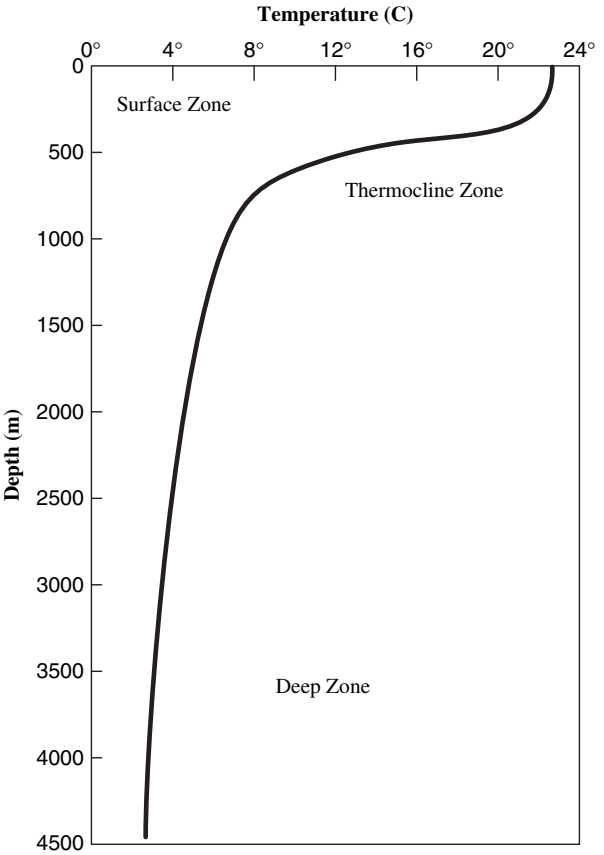


Fig. 30.5 Typical ocean temperature profile

The study region covers an area measuring 7° by 7°C on the eastern portion of the South China Sea domain as shown in Fig. 30.1. The area ranges from 114.5 E to 120.5 E and from 15.5 N to 21.5 N.

A root mean squared error, RMSE, and a Nash–Sutcliffe index, R^2, are used as goodness-of-fit measures. They are expressed as

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (x_i - \hat{x}_i)^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2} \tag{30.1}$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum (y - \hat{y})^2} \tag{30.2}$$

where $y =$ observed value, $\hat{y} =$ predicted value, $\bar{y} =$ mean of the observed values and N is the number of data points considered.

Monthly averaged ocean temperature in May taken from the Levitus98 database is used in this study (http://www.cdc.noaa.gov/cdc/data.nodc.woa98.html). Typically,

Table 30.1 Depth level in the Levitus98 database

Level	Depth	Level	Depth	Level	Depth
1	0	9	150	17	800
2	10	10	200	18	900
3	20	11	250	19	1000
4	30	12	300	20	1100
5	50	13	400	21	1200
6	75	14	500	22	1300
7	100	15	600	23	1400
8	125	16	700	24	1500

the data are available at 24 depth levels up to 1500 m. Depth levels of Levitus98 data are given in Table 30.1.

The availability of data in the study area is given in Fig. 30.6. The data at points 1 and 2 are reserved for validation purposes.

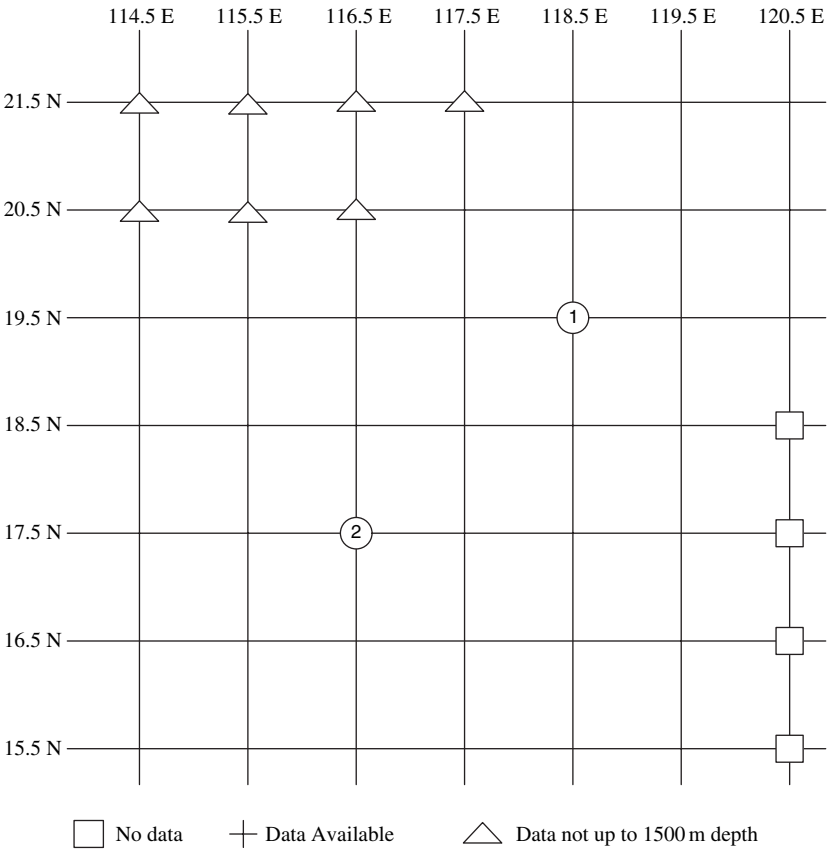


Fig. 30.6 Data availability in the study area

The task is to establish an equation with prediction capability of the sea temperature profile, T, at any location. The following relationship between T, at a particular location and depth, and sea surface temperature (SST) and its exact location (depth, longitude and latitude) is proposed:

$$T = f\,(\text{Sea Surface Temperature, Depth, Longitude, Latitude}) \tag{30.3}$$

Genetic programming is then trained to relate the dependent variable, temperature T at a given depth and location, with the four independent variables as expressed in (30.3). All variables are non-dimensionalized with respect to their individual maximum value prior to training. GP is first trained to yield the optimal relationship for the region under consideration. The GP configuration is

- Population size 50
- Number of generations 200
- Maximum number of levels 10
- Operators used in computation were addition (+), subtraction (−), multiplication (*), division (/), square root, log, power and exponential

The GP software used is GPLAB, a genetic programming toolbox for MATLAB written by Sara Silva (http://gplab.sourceforge.net/). Equation (30.4) shows the GP-trained equation selected in this study:

$$T = \frac{1}{7.396X1 + \left(\frac{X3}{X2X4}\right)^{1/4}} \tag{30.4}$$

where

$$X1 = \frac{\text{Depth}}{\text{Max(Depth)}}; X2 = \frac{\text{Longitude}}{\text{Max(Longitude)}}$$
$$X3 = \frac{\text{Latitude}}{\text{Max(Latitude)}}; X4 = \frac{\text{SST}}{\text{Max(SST)}}$$

It is noted that the above equation is only one of numerous equations generated by GP. All equations are ranked based on their performances on the training data set. Equations ranked very high in their performances may not necessary be chosen when they carry little or no physical interpretation. The following is an example of an equation with high performance and yet not chosen:

$$T = -0.1642 + e^{(2X3+X1+0.285-e^{X4})} - e^{(3X2+2X3+X4)} + 2X4 \tag{30.5}$$

It should also be noted that several transformation options are available, e.g. (1) dividing all the data by their corresponding maximum values; (2) dividing all the data by their corresponding range (maximum value – minimum value); (3) logarithmic transformation; etc. Evaluations show that transformation with the maximum value yields simpler equations and good performances as well.

The domain used in this study is from 114.5 E to 120.5 E and from 15.5 N to 21.5 N. Thus, the transformed values of $X2$ (latitude) are from 0.95 to 1, and $X3$ (longitude) from 0.72 to 1. Splitting the second term in the denominator of (30.4) into $\left(\frac{X3}{X2}\right)^{1/4} \times \left(\frac{1}{X4}\right)^{1/4}$, the effect of location ($X2, X3$) at the same water depth ($X1$) and SST ($X4$) on temperature is not dominant. The reason is the term $\left(\frac{X3}{X2}\right)^{1/4}$ ranges only from 0.92 to 1. It should be noted that in the domain considered in this study, the temperatures of the same depth at two locations (that differ by only 0.5° from each other) are not very different.

The trained GP is then validated on sea temperature profiles at various points in the same domain in the month of May; it is noted that these validation data are extracted from the Levitus98 database. Performance accuracy levels for both training and validation are very high and given in Table 30.2.

The trained GP is then subjected to a further test with measured data taken from an expedition AsiaEx (Denner et al., 1999). The AsiaEX project was a joint international project between 18 major institutions in the period from 1997 to 2001. Essentially, the project measured some of the ocean characteristics at two sites: one in the South China Sea (SCS) while the other was in the East China Sea (ECS). At the SCS location, the CTD (Conductivity Temperature Depth) equipment is used to measure salinity, temperature and depth. At the ECS location, the ADCP (Acoustic Doppler Current Profile) equipment is used to measure the speed and direction of the ocean currents.

The data used in this paper are the measured temperature at the SCS site located between Dongsha Island (formerly Pratas Island) and Kao-hsiung (Taiwan), which is in the same domain as the Levitus98 study area but for two different time stages in the month of May, as shown in Table 30.3. Table 30.3 and Fig. 30.7 show the high performance accuracy (R^2 of over 0.98) of GP-simulated sea temperature profiles against their counterparts from observations.

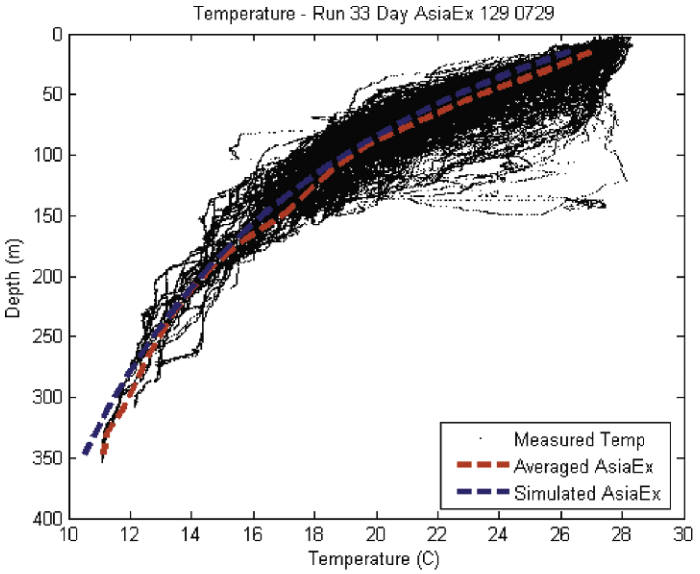
The good agreement between the simulated and the observed data is very encouraging. The study will look into the applicability of the approach to other sea

Table 30.2 Prediction accuracy of GP for Levitus98 data

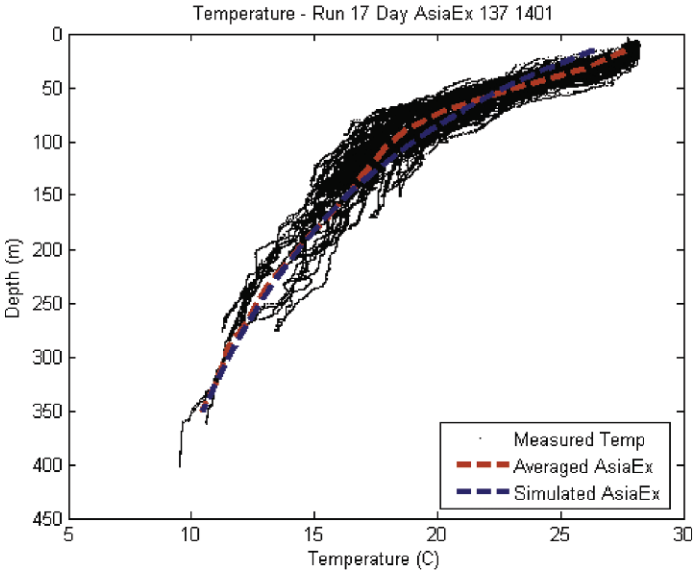
	Nash index (R^2)	RMSE
Training	0.9911	0.0286
Validation: Point 1	0.9923	0.0262
Validation: Point 2	0.9920	0.0287

Table 30.3 Performance accuracy of GP: AsiaEx data

Time period	Period	R^2	RMSE
1	3/05/01–8/05/01	0.98986	0.49075
2	14/05/01–16/05/01	0.98586	0.60059



(a) Temperature profile for data collected between 3 and 8 May 2001



(b) Temperature profile for data collected between 14 and 16 May 2001

Fig. 30.7 Comparison between GP-simulated and measured temperature profiles (a) Temperature profile for data collected between 3 and 8 May 2001 (b) Temperature profile for data collected between 14 and 16 May 2001

regions through validation with observed data. Should the comparisons be of the same order of magnitude as that shown here, one can then extract the readily available sea surface temperatures, observed by orbiting satellites (Topex/Poseidon; http://www.saa.noaa.gov/nsaa/ products/welcome), to derive the required sea temperature profile.

30.4 Conclusions

This chapter first discussed the need for good input data, such as sea temperature profiles and tidal elevations, for ocean model simulations and then demonstrated the application of genetic programming on sea temperature profile simulation.

Comparisons between data-driven simulated and observed data were performed. Results show very close agreement. The high accuracy in projecting sea temperature profiles allows ocean model users to couple the readily available satellites' observed sea surface temperature data with a GP-trained model to predict the required sea temperature profiles.

Acknowledgements The authors wish to acknowledge the contributions from their colleagues, H. K. Choo, X. H. Su and X. Y. Yu.

References

Chu PC, Chang CP (1997) South China Sea warm pool boreal spring. Advances in Atmospheric Science 14: 195–206.

Chu PC, Chang CP (1999) Dynamical mechanisms for the South China Sea seasonal circulation and thermohaline variabilities. Journal of Physical Oceanography 29(11): 2971–2989.

Chu PC, Tseng HC (1997) South China Sea warm pool detected in spring from the Navy's Master Oceanographic Observational Data Set. Journal of Geophysical Research 102(C7): 15761–15771.

Dale WL (1956) Winds and drift currents in the South China Sea. Malayan Journal of Tropical Geography 8: 1–31.

Denner WW, Chiu CS, Ramp SR (1999) Report on the Office of Naval Research, Phase III International Workshop on Shallow-Water Acoustics, Alyeska Resort, Girdwood, Alaska, July 12–15.

Hellerman S, Rosenstein M (1983) Normal monthly wind stress over the world ocean with error estimates. Journal of Physical Oceanography 13: 1093–1104.

Koza JR (1992) Genetic Programming: On the Programming of Computers by Means of Natural Selection, The MIT Press, Cambridge, MA.

Langdon WB (1998) Genetic Programming and Data Structures: Genetic programming + Data structures = Automatic Programming, Kluwer Academic Publishers.

Liong SY, Gautam TR, Khu ST, Babovic V, Keijzer M, Muttil N (2002) Genetic programming: A new paradigm in rainfall runoff modeling. Journal of American Water Resources Association 38(3): 705–718.

Masumoto Y, Yamagata T (1991) Response of the western tropical Pacific to the Asian winter monsoon: The generation of the Mindanao dome. Journal of Physical Oceanography 21: 1386–1398.

Metzger EJ, Hurlburt HE (1996) Coupled dynamics of the South China Sea, the Sulu Sea, and the Pacific Ocean. Journal of Geophysical Research 101(C5): 12331–12352.

Metzger EJ, Hurlburt HE, Kindle JC, Sirkes Z, Pringle JM (1992) Hindcasting of wind driven anomalies using a reduced gravity global ocean model. Marine Technology Society Journal 26(2): 23–32.

Murray SP, Arief D, Kindle JC, Hurlburt HE (1990) Characteristics of the circulation in an Indonesian archipelago straits from hydrology, current measurements, and modeling results, The Physical Oceanography of Sea Straits, pp. 3–23, Kluwer Academic Publishers, Norwell, MA.

Ningsih NS, Yamashita T, Aouf L (2000) Three-dimensional simulation of water circulation in the Java Sea: Influence of wind waves on surface and bottom stresses. Natural Hazards 21(2–3): 145–171

Nitani H (1970) Oceanographic conditions in the sea east of Philippines and Luzon Strait in summer of 1965 and 1966, The Kuroshio-A Symposium on Japan Current, edited by J. D. Marr, pp. 213–232, East-West, Honolulu, Hawaii.

Soong YS, Hu JH, Ho CR, Niiler PP (1995) Cold-core eddy detected in South China Sea, EOS Trans. AGU, 76: 345–347.

Wyrtki K (1961) Scientific results of marine investigations of the South China Sea and Gulf of Thailand 1959–1961, NAGA report 2.

Zhang H, Chan ES (2002) Ocean prediction modeling in regional seas, Technical Report TR-IOEPAS-1, Feb 2002.

Part VI
Model Integration

Chapter 31
Uncertainty Propagation in Ensemble Rainfall Prediction Systems used for Operational Real-Time Flood Forecasting

I.D. Cluckie and Y. Xuan

Abstract Advances in mesoscale numerical weather prediction make it possible to provide quantitative precipitation forecasts (QPF) along with many other data fields at increasingly higher spatial resolutions. It is currently possible to incorporate high-resolution NWPs directly into flood forecasting systems in order to obtain an extended lead time. It is recognised, however, that direct application of rainfall outputs from the NWP model contributes considerable uncertainty to the final river flow forecasts as the uncertainties inherent in the NWP are propagated into hydrological and hydraulic domains and can also be magnified by the scaling process. As more and more "modern" flood forecasting systems are adopting this coupled approach, it is necessary to study uncertainty propagation and interaction between the NWP and the real-time flood forecast system model cascade, which terminates technically with the decision support system (DSS).

In this study, analysis is conducted to investigate the uncertainties in rainfall predictions that form the primary perturbation in a coupled NWP-hydrological model context. The ensemble method is used to account for both uncertainties due to incorrect/inaccurate initial conditions and those derived from model structure. Conventional statistics are employed to show variations over domains as well as point-wise targets. An adapted empirical orthogonal function analysis based upon principal components (EOF/PCA) is used to measure the diversity of ensemble members, which in turn provides a way to reconstruct a composite scenario that embodies most of the significant characteristics of ensemble forecast fields. The analyses of a typical ensemble QPF case over the catchment scale reveals that, although the NWP-based QPF can generally capture the rainfall pattern, uncertainties in rainfall at the scale of model grid relative to the catchment scale were always significant. Therefore, a cautious approach should be taken before the QPF, either deterministic or ensemble based, is injected into a flood forecasting system. Detailed results are

I.D. Cluckie
Water and Environmental Management Research Centre (WEMRC), Department of Civil Engineering, University of Bristol, Bristol, BS8 1UP, UK, e-mail: I.D.Cluckie@bristol.ac.uk

Y. Xuan
Formerly of Water and Environmental Management Research Centre (WEMRC), Department of Civil Engineering, University of Bristol, Bristol, BS8 1UP, UK (currently at UNESCO-IHE)

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

discussed and comments made regarding the uncertainty propagation and the usability of the NWP-based QPF in the context of real-time flood forecasting systems.

Keywords Flood forecasting · short-range ensemble numerical weather prediction · uncertainty propagation

31.1 Introduction

With advances in numerical weather prediction (NWP) in recent years as well as increases in computing power, it is now possible to generate very high-resolution rainfall forecasts at the catchment scale and therefore more flood forecasting systems are tending to utilise quantitative precipitation prediction (QPF) from high-resolution NWPs in order to extend the forecast lead time; this is particularly so in the flash flooding area where the model performance is highly dependent on the rapid availability of knowledge of rainfall distribution in advance (De Roo et al., 2003, Ferraris et al., 2002). Many efforts have been made to utilise the QPF in the context of real-time flood forecasting in which one or more "state-of-the-art" QPFs stemming from different methods are to be integrated into the whole system. However, the effects of QPF uncertainty on the whole system can be easily appreciated either intuitively or by case studies (De Roo et al., 2003; Cluckie et al., 2004a,b). Indeed, recent research on integrating QPF directly into the real-time flood forecasting domain reveals that directly coupling the QPF with the hydrological model can result in large bias and uncertainty (Ferraris et al., 2002). In this chapter, the uncertainties of rainfall prediction from the NWP are analysed in the context of application within a real-time flood forecasting system and some conclusions are provided to guide the future full integration of NWP-derived QPFs within the system. This provides a contribution to the whole system modelling philosophy that is currently prevalent in the flood risk management sector.

31.1.1 Generation of Rainfall Predictions for the Real-Time Flood Forecasting Domain

The weather models, which are routinely run in national weather centres, have such a coarse spatial resolution that hydrological models have difficulty applying the result from them as an effective input. The so-called downscaling procedure is needed to bridge the scale gap between the large-scale weather forecast domains and catchments-sized flood forecasting domains. In this study, a dynamical-downscaling approach is applied to resolve the dynamics over 2 km grids. The forecasts/analyses from global weather models are used to settle the initial and lateral boundary conditions (IC/LBC) of a mesoscale model which is able to benefit from the resolvable terrain features and physics at higher resolutions. This sort of mesoscale model is often referred to as a local area model (LAM) and can most effectively be coupled with the hydrological modelling process.

31.1.2 Uncertainty Issues and Ensemble Prediction

Lorenz (1963, 1993) introduced the concept that the time evolution of a nonlinear, deterministic dynamical system, to which the atmosphere (essentially a boundary layer process) and the equations that describe air motion belong, is very sensitive to the initial conditions of the system. The uncertainties inherited in the larger scale model, which provides the parent domain with the IC and LBC, can also be propagated to the nested mesoscale models. Given the large number of degrees of freedom of the atmospheric phase state, it is impractical to generate directly solutions of the probabilistic equations of initial states. The ensemble forecast method runs the model separately over a probabilistically generated ensemble of initial states, each of which represents a plausible and equally likely state of the system. Each ensemble projects them into future phase space and, as such, can be represented by the statistics of the ensemble.

As far as a flood forecasting model is concerned, both the amount and the distribution of rainfall can significantly affect the final outputs. The results of flood forecasting, however, could be disastrous if the rainfall forecasts were applied without recognising field displacement problems, as the rainfall event could have been either missed or mistakenly overestimated, i.e., the uncertainties may become more significant when the focus has changed from large weather model domains to much smaller catchment scales (Ferraris et al., 2002, Cluckie et al., 2004a and b). Furthermore, numerical models tend to smooth the rainfall field (and other variables) and as such often cannot efficiently reproduce sub-grid variability, which plays a key role in the runoff generation process.

31.2 The Mesoscale QPF Ensemble System

31.2.1 Model Configurations

The mesoscale ensemble QPF system consists of the mesoscale weather model – PSU/NCAR mesoscale model (MM5) (Dudhia, 2003), the global analyses/forecast data sets from the European Centre for Medium-range Weather Forecasts (ECMWF) (Persson, 2003), and a post-processing system. Details of this system can be found in the literature (Butts et al., 2005).

MM5 is configured to have four nested domains in this case study, of which the grid size is 54 km, 18 km, 6 km, and 2 km from outermost to innermost domain respectively, as seen in Fig. 31.1. The case study catchment is the Welland and Glen, which is located on the east coast of the UK and experienced a severe flood event during Easter 1998 (09 April to 12 April 1998). To simulate this flood-producing rainfall event, data sets from the ECMWF were used, including 50 members of its ensemble prediction system (EPS), operational forecasts, and one control run of the EPS. All EPS members from ECMWF are then downscaled to the catchment scale by MM5 with two different physics schema applied; therefore the mesoscale

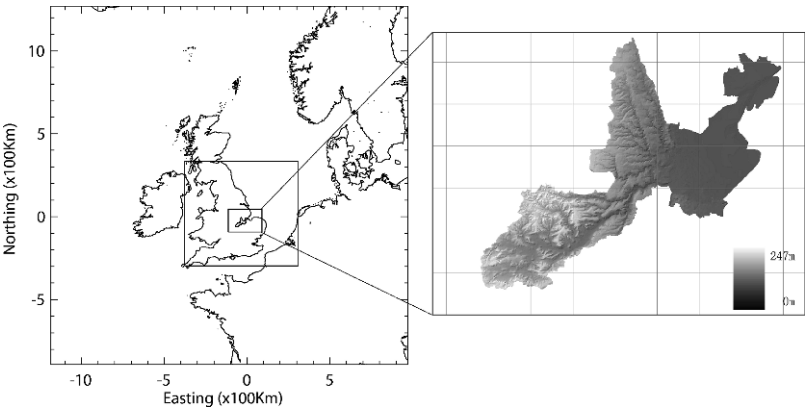


Fig. 31.1 Domain configuration (*left*, only inner three domains are shown) with the Welland and Glen Catchment Terrain (in metres and 10 km background grids)

QPF will have 104 members, each of which embodies the unique realisation of the assumed physics fields over the mesoscale domain.

MM5 was initialised at 12UTC on 09 April 1998 and provided a 24 h forecast until 12UTC on 10 April. There were 27 gauge observations available in the vicinity of the Welland and Glen catchment, from which the 24 h rainfall accumulation was derived in order to evaluate the spread of the 24 h forecasts from the mesoscale ensemble QPF system.

31.2.2 Analysis Approach

Conventional statistics were applied to analyse the model outputs in order to understand how the model performance and uncertainty was distributed over domains. The conventional statistics were divided into two categories, which were grid (points) orientated and area (aggregated) value orientated. It can be seen that the grid-based statistics generated the pattern of the particular fields, whereas the area-based provided the general evaluation of the fields that were later used as input into lumped models.

The ensemble members were also transformed into empirical orthogonal functions (EOF) that can provide insights into how individual members varied with each other (Wilks, 1995). In the analysis, values of grids are seen as variables and the entire ensemble data set is defined to be all variables sampled over the sampling space, which refers to the ensemble case members.

31.2.3 Analysis of the Distribution over the Area

The operational run, control run, and the mean value of members are shown in the top row of Fig. 31.2, together with three randomly selected members displayed in

the middle. Not surprisingly, the patterns of the operational run and the control run agree very well. The main structures were captured in the mean pattern with one storm centre located towards the eastern border that was missed. The huge variability among ensemble members is clearly reflected by the three lower images, of which the lower left predicts a very strong but small rainfall centre whilst missing the other one almost completely.

The standard deviation, the coefficient of variation, and the skewness are shown in the bottom row of Fig. 31.2. The standard deviation indicates that two heavy rainfall centres are also the locations generating much uncertainty. The coefficient of variation in this case implies that the middle of the eastern boundary is such that a single member forecast cannot be accepted owing to its large error band. The distribution of the skewness shows that the PDF of the ensemble over the domain exhibits high heterogeneity with a highly skewed PDF occurring to the northeast of the domain, which means that the majority of the ensembles predict small rainfall while several members produced large values. It can also be concluded that models are more sensitive to the coastal area than the inland area where there is another storm centre.

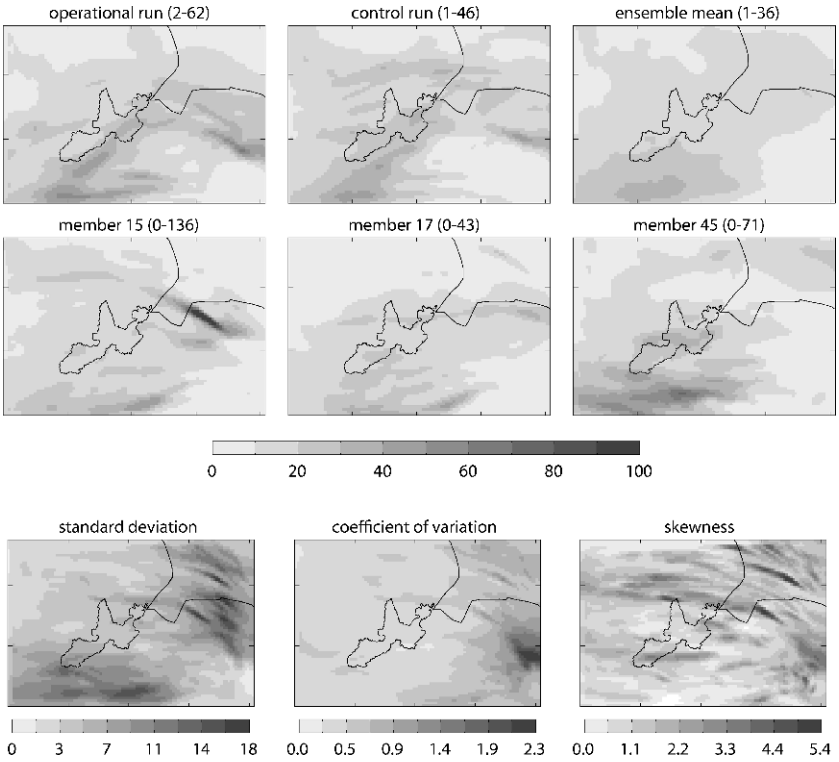


Fig. 31.2 Rainfall forecast distribution over domain4 in mm (*top two rows* with the value range shown in brackets) and indices to describe variation (*bottom three figure*). The unit of standard deviation is mm whereas the remaining two are dimensionless. Horizontal and vertical axes of each sub-figure are marked every 50 km

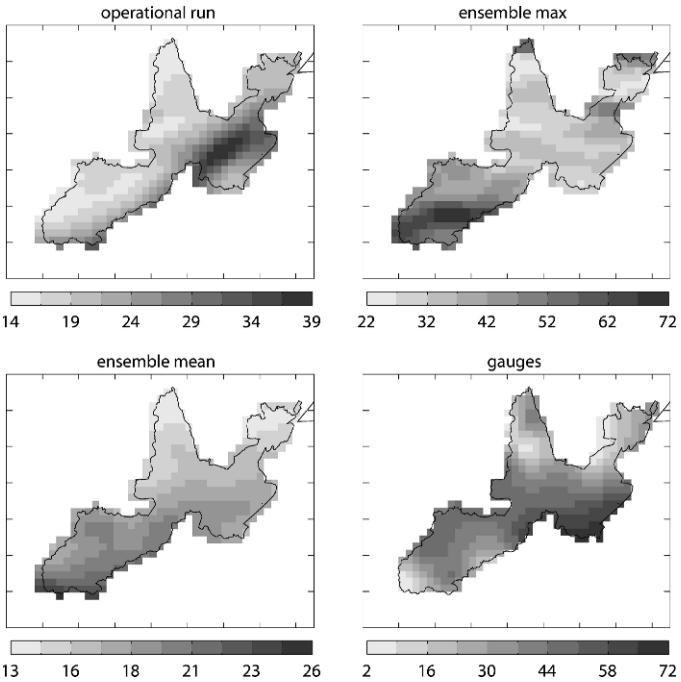


Fig. 31.3 Rainfall distribution over the Welland and Glen catchment. (Unit: mm. The one titled "ensemble max" shows the pessimistic forecast. Horizontal and vertical axes of each sub-figure are marked every 10 km)

Rainfall fields are zoomed and cropped in Fig. 31.3 as the scale changes from domain to the catchment. Note that the operational forecast and the members mean are just the magnified versions of those shown previously and thus the pattern can be identified more clearly. What is more interesting, however, is the pessimistic forecast (the upper envelope of members) and the one interpolated from gauge data. In practice, apart from the mean, the extreme forecast value is usually used to provide a risk-averse forecast. Compared with the "real" distribution, the ensemble max has included the maxima of the gauges but failed to realise the distribution pattern; the operational run has a comparable pattern, but the total rainfall amount over the catchment only equals half of that observed.

31.2.4 Variations of Aggregated Value

It is customary in flood forecasting to use rainfall aggregated over the catchment due to the fact that traditionally the vast majority of operational hydrological models tend to use areal rainfall inputs, which is usually derived from limited ground-based

measurements. Because of this, it may be more helpful to check the uncertainties in the QPF in terms of area-averaged or catchment-averaged values.

Again, the quantities are compared for both domain and catchment, and the dimensions have been scaled to unity. It is preferred to analyse them under a time frame so that a more meaningful dynamic series can be obtained. By doing this, a familiar "spaghetti" diagram, as seen in Fig. 31.4, was utilised.

It is difficult to find pronounced trends from the two spaghetti diagrams given in the top row in Fig. 31.4, but generally, spreads of both increase dramatically after 6 h and then stabilise until 24 h. The spread of the catchment case is larger and more chaotic, while for the entire domain case each member appears to be staying on track with minimum disruption. This is contrary to the intuition obtained from what was shown in the distribution images (see Fig. 31.2), which indicate that the catchment is in an area where the variation was small. Indeed, the spatial scaling issue comes into play at this time, and the variation tends to increase when the area size decreases. The comparison of the standard deviation of both cases confirms this expectation. As seen in the lower left of Fig. 31.4, the variation of the catchment case exceeds its counterpart from the whole domain.

A time-dependent tendency can be found in the C_v value shown in Fig. 31.4, from which it can be seen that the normalised variation of the catchment is being amplified while time increases. However, such a trend is not pronounced in the whole domain case within a short forecast period. One reasonable explanation is that for a large area, many more members and a longer elapsed time may be needed to reveal this behaviour, while the scaling process may work as a non-linear magnifier to expose such a trend at a much earlier time with fewer ensemble members.

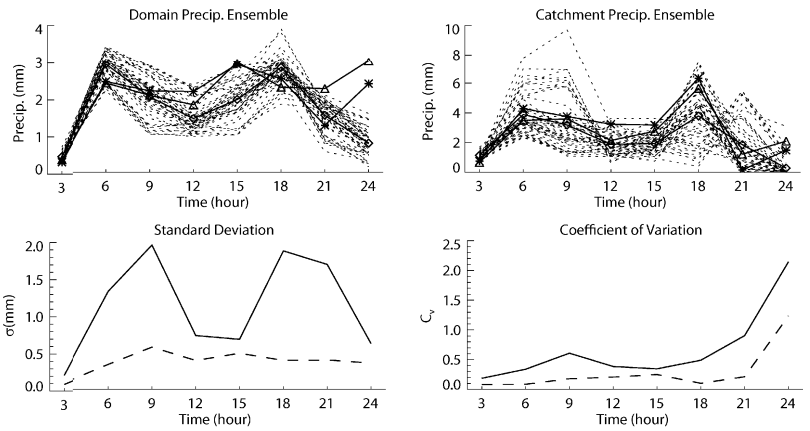


Fig. 31.4 Areal rainfall forecast of entire domain and catchment (*top*: where *solid lines* marked by *triangles*, *stars* and *diamonds* represent operational run, control run, and members mean, respectively, and *dotted lines* represent ensemble members) together with the standard deviation and coefficient of variation (*lower*: where *dashed lines* depict values from the domain run and *solid lines* represent that from the catchment run)

31.2.5 Empirical Orthogonal Function (EOF) Analyses

The empirical orthogonal function (EOF), also known as principal component analysis, is used to look at the variation across ensemble members. Although the ensemble mean is able to provide a general averaged scenario, EOFs (or principal components) not only present the spatial structure but also provide associated information about how much variance can be accounted for.

Apart from the uncertainties propagated from the initial and lateral boundary conditions that are settled by the large-scale weather models, the mesoscale model can also contribute uncertainties of model structures; for instance, those of the moisture physics schema. In this study, two ensembles are firstly built up by applying different model physics, which are Resiner 1 and Resiner 2 (see Dudhia et al., 2003), and a grand ensemble including members from both is then generated. EOF analyses were conducted over all three ensembles, which revealed some local features related to the choice of model physics.

The first four leading EOFs are shown in Fig. 31.5, in which the brackets in each subtitle indicate the fraction of the total variances that a particular mode can account for. Clearly the leading EOFs have revealed statistically important spatial features; for example, two storm centres, seen before, appeared in EOF1. It would be more obvious to see this if a rainfall field was constructed based on the mode of the EOF, like Figs. 31.6a to c. The mode accounted for the total variances (27%, 13%, 6%, and 5%), however, are not significantly large, which means that there is a substantial

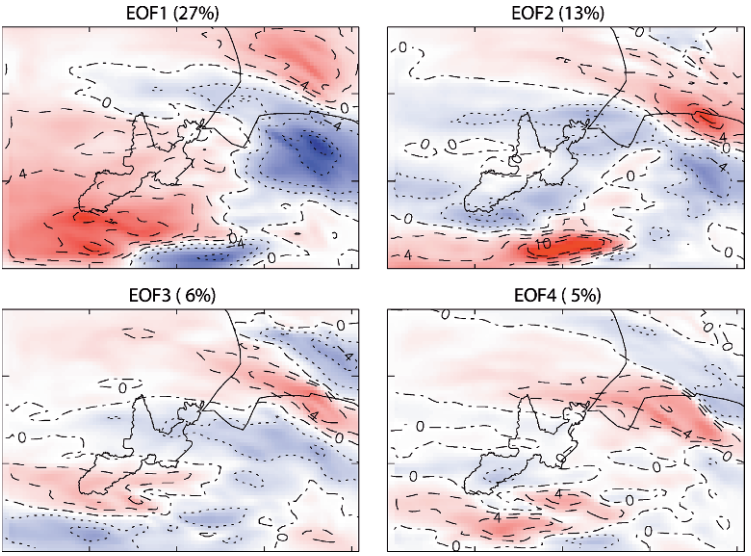


Fig. 31.5 First four leading scaled EOF contours in mm and the fraction of total variance they account for (the zero contours are in *dash-dot lines* and positive/negative values are shown with *dash/dotted contours*. *Coloured shadow* is used to emphasise the relative amplitude. Horizontal and vertical axes of each sub-figure are marked every 50 km)

spread of trajectories in the model phase space, or in other words, the four leading modes can hardly produce a general scenario due to the uncertainty of predictions across the members.

The variation of model outputs due to model structure perturbation can be found by comparing panel a and b of Fig. 31.6, which are rainfall patterns reconstructed from the first EOFs of the ensembles with two different model physics schema. More importantly, one of the different localised features, which is the small storm centre located across the south-east boundary of the catchment, has also been discovered by gauge measurements but missed out by the ensemble mean (see previous discussion). The pattern of panel (c) is from the grand ensemble including both types of physics schema; therefore it has a bigger coverage that encompasses local variations from both panels (a) and (b).

Since to some extent the fraction of the variance accounted for by the EOF mode can be seen as a measure of the divergence of ensemble members in terms of orthogonal transformation, Fig. 31.6d reflects the fact that ensembles with physics option 1 spread more widely than those from option 2; furthermore, the grand ensemble has the highest cross-member variations as it combines both options. However, the

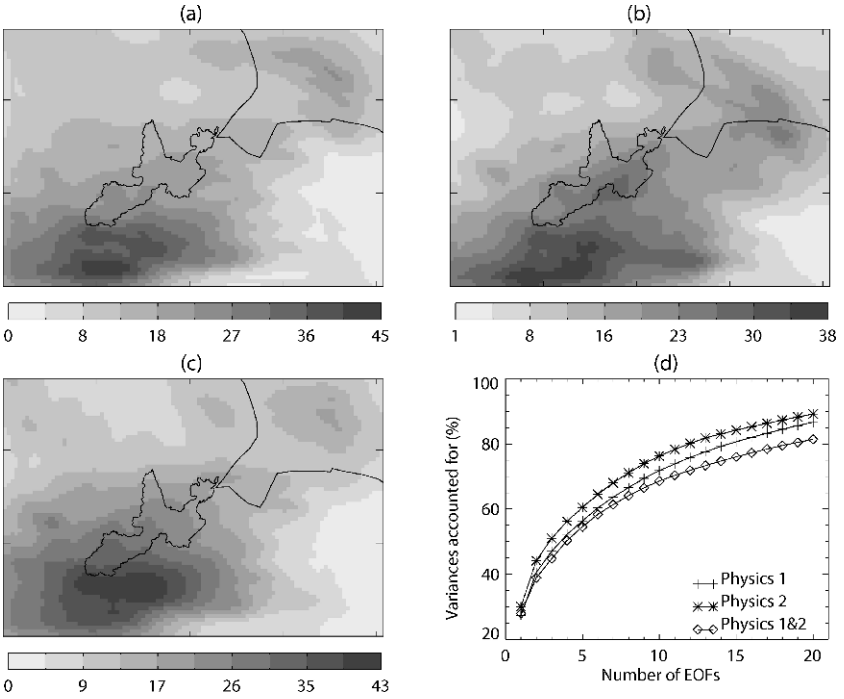


Fig. 31.6 Reconstructed rainfall patterns in mm from the first leading EOF with physics option 1, option 2, and the mix option (**a**, **b,** and **c**). Horizontal and vertical axes of each sub-figure are marked every 10 km) and the variance accounted by the EOF with different model physics options (**d**)

shape of the curve probably implies the upstream IC/LBC uncertainties overwhelm those from model structure perturbations.

31.3 Conclusions

A mesoscale ensemble quantitative precipitation forecast system was established to evaluate the propagation of uncertainty in the application of the QPF in a real-time flood forecasting system. The grid-orientated statistics were applied to reveal the variations across grid points and the spatial distribution. Aggregated values, such as catchment mean precipitation, were also analysed focusing on the temporal evolution. Finally, the empirical orthogonal function analyses (EOF) was used to summarise the uncertainties of model ensembles in general. From these analyses, the following conclusions can be made.

A large amount of uncertainty exists in the rainfall distributions over both domains and catchment scales; therefore, members of an ensemble may differ significantly in terms of rainfall forecast distribution. Ensemble means are not necessarily the best estimate regarding pattern. The distribution of this variation itself is non-uniform; large variations/uncertainties tend to appear in the vicinity of heavy rainfall areas, which can also have a strongly skewed PDF while stable areas tend to have a more symmetric PDF. Aggregated rainfall values such as area-averaged rainfall exhibits a spatial–temporal dependency. Variations tend to increase with decreases in the domain size and grow with time.

At the catchment scale, rainfall patterns from deterministic runs and the ensemble members mean were generally similar to that from interpolation of rain gauge observations. Rainfall amounts, however, were severely underestimated for most of the ensemble members, i.e. the spread was not able to encompass appropriately what happened historically. Inefficient model physics or inadequacy of ensemble members can both result in this problem.

The leading EOFs of an ensemble can represent the most significant spatial features, which implies a practical value in generating a general scenario for operational applications. Effects of model physics can be detected as well by the EOF and the uncertainties due to mesoscale model physics are often overwhelmed by those propagated from the upstream larger scale models. The reconstructed pattern from the EOF may be more suitable to capture the real distribution than the ensemble mean.

The mesoscale QPF ensemble system has been shown to be a powerful tool by giving not only a simple deterministic rainfall forecast but, more importantly, additional valuable information about uncertainty structure and its prognostic properties.

The case study provides initial guidance on these issues but to come to a more general conclusion it will be necessary to address different issues associated with different weather systems and model resolutions. Further work will focus on the post-processing of the ensemble QPF that is able to produce refined distributions and associated probabilities suitable for being directly integrated into an ensemble-based flood forecasting system.

Acknowledgements The authors would like to thank ECMWF for access to their data sets and computing facilities. The rain gauge data was provided by the Environment Agency, and the MM5 modelling system software was kindly made available and supported by the Mesoscale Prediction Group in the Mesoscale and Micro-scale Meteorology Division, NCAR, USA. Thanks are also due to the EU FP5 FLOODRELIEF Project (EVKI-CT-2002-00117) and the EPSRC Flood Risk Management Research Consortium (FRMRC – GR/S76304/01) for financial support.

References

Butts M, Larson JK, Cluckie ID, Brandt J, Mengelkamp HT, Dubicki A, Wozniak Z, Lewandowski A, Price D, Chen Y (2005) Uncertainties Framework, Deliverable 4.1, FLOODRELIEF Project, URL: http://projects.dhi.dk/floodrelief/

Cluckie I, Han D, Xuan Y (2004a) Integrated QPF with combined data from weather radar and numerical weather model, Deliverable 4.1 FLOODRELIEF Project, URL: http://projects.dhi.dk/floodrelief/

Cluckie I, Han D, Xuan Y (2004b) Preliminary Analysis on NWP-Based QPF over UK domain. Deliverable 4.2, FLOODRELIEF Project, URL: http://projects.dhi.dk/floodrelief/

De Roo A, Gouweleeuw B, Thielen J, Bartholmes J et al. (2003) Development of a European flood forecasting system. International Journal of River Basin Management 1(1): 49–59

Dudhia J. et al. (2003) PSU/NCAR Mesoscale Modelling System Tutorial Class Notes and User's Guide: MM5 Modelling System Version 3. Mesoscale and Micro-scale Meteorology Division, National Centre for Atmospheric Research, unpublished

Ferraris LR, Rudari R, Siccardi F (2002) The uncertainty in the prediction of flash floods in the Northern Mediterranean environment. Journal of Hydrometeorology 3: 714–727

Lorenz EN (1963) Deterministic non-periodic flow. Journal of the Atmospheric Science 20: 130–141

Lorenz EN (1993) The Essence of Chaos, University of Washington Press: Seattle

Persson A (2003) User Guide to ECMWF forecast products, Meteorological Bulletin M3.2, ECMWF

Wilks DS (1995) Statistical Methods in the Atmospheric Sciences, Academic Press: London

Chapter 32
OpenMI – Real Progress Towards Integrated Modelling

D. Fortune, P. Gijsbers, J. Gregersen and R. Moore

Abstract The case for integrated catchment management has been well made and has become a core part of initiatives such as the EC Water Framework Directive. Most water and environmental professionals expect that integrated modelling will take a necessary role in decision support for catchment management, in both planning and operations. Indeed, there is a common assumption that such modelling exists and is regularly carried out. Yet a closer examination reveals that where model integration does exist, it is in a hard-wired or entirely proprietary way. The shining example is in the specialist and limited field of real-time flood forecasting, where there are some notable implementations of integrated modelling, albeit with many restrictions on the connections (or linkage) between models.

The widespread implementation of integrated modelling depends on the availability of a sufficiently flexible and powerful linkage mechanism for data exchange between models, and indeed between models and user interfaces, databases and other data-hungry processes. Until now, no such linkage has existed, and easy availability of integrated modelling has remained a dream. Under the EU Framework 5 HarmonIT Project, a remarkable collaboration between rival commercial software specialists, with the help of some excellent academic, research and operational partners, has developed the OpenMI standard for data interfaces. It is anticipated that this standard will have far-reaching consequences for modelling and for hydroinformatics in general.

An implementation of the standard has also been developed, along with sufficient utilities to allow the standard to be tested to the point of scientific proof or proof of concept. The main tests have involved the incorporation of the OpenMI data interface in a range of model source codes, from widely used commercial codes to

D. Fortune
Wallingford Software Ltd, Howbery Park, Wallingford, Oxfordshire, OX10 8BA, UK,
e-mail: david.fortune@wallingfordsoftware.com

P. Gijsbers
WL | Delft Hydraulics, NL, e-mail: peter.gijsbers@wldelft.nl

J. Gregersen
DHI Water and Environment, Denmark, e-mail: jbg@dhigroup.com

R. Moore
CEH Wallingford, UK, e-mail: rvm@ceh.ac.uk

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

specialist research models. Various combinations of populated models were then built and the data transfers tested.

This chapter presents an overview of the technical details of the OpenMI standard. It also briefly describes how existing modelling software can be made OpenMI-compliant. The full specification can be downloaded from the OpenMI-website www.openmi.org or can be downloaded from the documentation section of the OpenMI environment software installation package (including source code) available from the CVS repository on sourceforge.net/projects/openmi.

The authors acknowledge that the availability of a suitable integration mechanism is just the beginning. OpenMI now exists as a freely available, open-source standard, but its long-term future needs to be secured for it to have lasting impact on catchment management. Much research and development needs to be done to understand how best to implement integration between different sorts of models with a variety of linkages. Should we expect a major shift from single-issue modelling to integrated modelling? Does OpenMI offer new opportunities to modellers and to water managers? The authors expect that the answer to both questions is "yes", but we will have to wait and see if OpenMI really does change modelling in practice.

Keywords Open source software · OpenMI · integrated modelling · HarmonIT

32.1 Introduction

Integrated water management requires an understanding of catchment processes and the ability to predict how they will respond under different management policies. Most traditional modelling systems have not been able to meet these requirements as models have tended to represent individual processes and have been run independently. Hence, their output may not reflect the interactions between different aspects of the environment. Clearly, it is not practical to construct a single model that could simulate all catchment processes and to do so would be wasteful of the large number of existing models. A better solution is to couple models and hence enable them to exchange data as they run, thus allowing interactions to be represented in the simulation.

In response to the need created by the Water Framework Directive from its introduction of integrated water management, the HarmonIT project has developed the Open Modelling Interface and Environment (the OpenMI) to allow models to exchange data (HarmonIT, 2005a). This has been developed with the following objectives in mind: (i) the standard should be applicable to new and existing models, requiring the minimum of change to the program code; (ii) the standard should impose as few restrictions as possible on the freedom of the model developer; (iii) the standard should be applicable to most, if not all, time-based simulation techniques; and (iv) implementation of the standard should not unreasonably degrade performance.

32.2 OpenMI Concepts

The majority of model applications that are the result of a design process (as opposed to those that just evolved over time) have a common structure comprising a user interface, input files, a calculation engine and output files. Typically, the *user interface* enables the user to create or point to *input* files and allows the visualization of the *output* or *result* files. The calculation *engine* contains the model algorithms and becomes a *model* of a specific process, e.g. flow in the Rhine, once it has populated itself by reading the input files. A model can compute output. If an engine can be instantiated separately, it is an *engine component*. If, further, it supports a well-defined interface, it becomes a component. Finally, if the engine supports the OpenMI linkable component interface, then the engine is said to be *OpenMI-compliant*. An engine component populated with data is a *model component*.

The OpenMI focuses on the run-time exchange of data between populated components. These components can be data providers (e.g. models, databases, text files, spreadsheets, pre/post-processors, data editors, in situ monitoring stations, etc.) and/or data acceptors (e.g. models or online visualization tools). Thus, the OpenMI potentially allows the development of a complete integrated modelling system consisting of GUI, engines and databases. Determining when this level of system integration will be achieved will depend on the adoption of OpenMI as a *component linkage standard* by the environmental model and software development community.

At the start of HarmonIT, the components were foreseen as running within a framework, but gradually this concept was replaced by standardizing the run-time interface of linkable components, thus allowing direct communication between components. The linkable component interface, described in the namespace org.OpenMI.standard, can be implemented in a variety of ways, of which the OpenMI environment, developed by the HarmonIT project, is just one.

Essential to OpenMI is the distinction between quantities, element sets, time and values. Elements can be non-geo-referenced or geo-referenced in 0/1/2/3D. The entities are supported by metadata interfaces describing what the data represent, to which location it refers, for which time (time stamp or time span) they are valid and how they are produced.

The developers of OpenMI created a full software implementation, called the OpenMI environment, in C# (.NET). A Java implementation is being developed within the Seamless and Sensor projects (EU Framework 6). The focus of OpenMI development was primarily oriented to data exchange issues. Hence, a wide range of utilities is provided to enhance the implementation of the OpenMI interfaces from typically legacy code. For example, a wrapper package provides facilities for bookkeeping of links and data handling, such as buffering and spatial and temporal mapping. Simple tools are available to define links, to run the system and to display results.

OpenMI is based on the request–reply architecture concept. The basic workflow is the following. At configuration time, metadata of a component is inspected to identify and define the links between the components. At run-time a component is

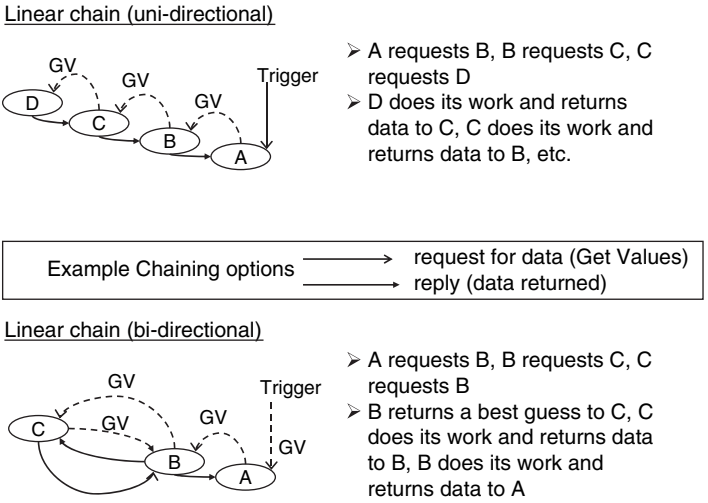


Fig. 32.1 The basis of OpenMI: the request–reply concept

requested by another component for values at a specific time and location (element set) using the GetValues-function call (see Fig. 32.1). This providing component is obliged to reply to this request and provide the data at the time and location as requested. In order to reply, it may need to do internal computations to progress in time. This computation may require external data that can be requested from other components by a GetValues-call. Once the internal time progressed at or past the requested time, data transformations can be applied to return the values at the exact time, element set and units as requested.

The interface orientation of OpenMI does not prescribe the use of the OpenMI environment as provided by the HarmonIT project. Hence, OpenMI can also be used to glue model engines with existing modelling frameworks through the OpenMI interface. OpenMI is expected to satisfy the modelling requirements of a wide group of users in various engineering domains, such as model coders, model developers, data managers and end users. The expected impacts are the simplification of the model linking process, the ability to represent feedback loops and process interactions, the establishment of a communication standard for modelling and a reduction in development time for decision support systems.

32.3 The OpenMI Standard

32.3.1 The OpenMI: A Request and Reply Architecture

OpenMI is based on the 'request and reply' mechanism. It consists of communicating components (source and target components) which exchange data in a pre-defined way and in a pre-defined format. The OpenMI defines both the component interfaces

and how the data are to be exchanged. The components are called linkable components to indicate that the OpenMI involves components that can be linked together.

From the data exchange perspective, the OpenMI is a purely single-threaded architecture where an instance of a linkable component handles only one data request at a time before acting upon another request. A component at the end of the component chain triggers the data process. Once triggered, components exchange data autonomously without any type of supervising authority. If necessary, components start their own computing process to produce the requested data. No overall controlling functionality is needed to guide time synchronization. Sometimes a local controller is needed to control convergence of data being exchanged.

32.3.2 Addressing General Linkage Issues

The OpenMI standard provides the following facilities for model linkage:

- *Data definition*: The base data model of the OpenMI describes the numerical values to be exchanged in terms of quantities (what), element sets (where), times (when) and data operations (how) (for details, see Fig. 32.5).
- *Metadata defining potentially exchangeable data*: Quantities, element sets and data operations are combined in exchange for item definitions to indicate what data can potentially be provided and accepted by a linkable component (for details, see Fig. 32.6).
- *Definition of actually exchanged data*: A link describes the data to be exchanged, in terms of a quantity on an element set using certain data operations (for details, see Fig. 32.6).
- *Data transfer*: Linkable components can exchange data by a pull mechanism, meaning that a (target) component that requires input asks a source component for a (set of) value(s) for a given quantity on a set of elements (i.e. locations) for a given time. If required, the source component calculates these values and returns them. This pull mechanism has been encapsulated in a single method, the GetValues() method. Dependent on the status of the source component, a call of the GetValues() method may trigger computation and, possibly, lead to further requests for data from other components. An important feature is the obligation that components always deal with requests in order of receipt.
- *Generic component access*: All functionality becomes available to other components through one base interface, the linkable component interface (for details, see Fig. 32.6). This interface needs to be implemented by a component for it to become OpenMI-complaint. Two optional interfaces have been defined to extend its functionality with respect to discrete time information and state management (for details, see Fig. 32.6). To locate and access the binary software unit implementing the interface, the OMI file has been defined. The OMI file is an XML file of a pre-defined XSD format, which contains information about the class to instantiate, the assembly hosting the class and the arguments needed for initialization.

- *Event mechanism*: A lightweight event mechanism has been introduced (see Fig. 32.6) to pass messages for call stack tracing, progress monitoring and to flag status changes which might trigger other components (e.g. visualization tools) to request for data via a GetValues()-call.

By convention a linkable component has to throw an exception if an internally irrecoverable error occurs. This exception is based on the Exception class as provided by the development environment.

32.3.3 Utilization and Deployment Phases

An OpenMI linkable component provides a variety of services which can be utilized in various phases of deployment. Figure 32.2 provides an overview of the phases that

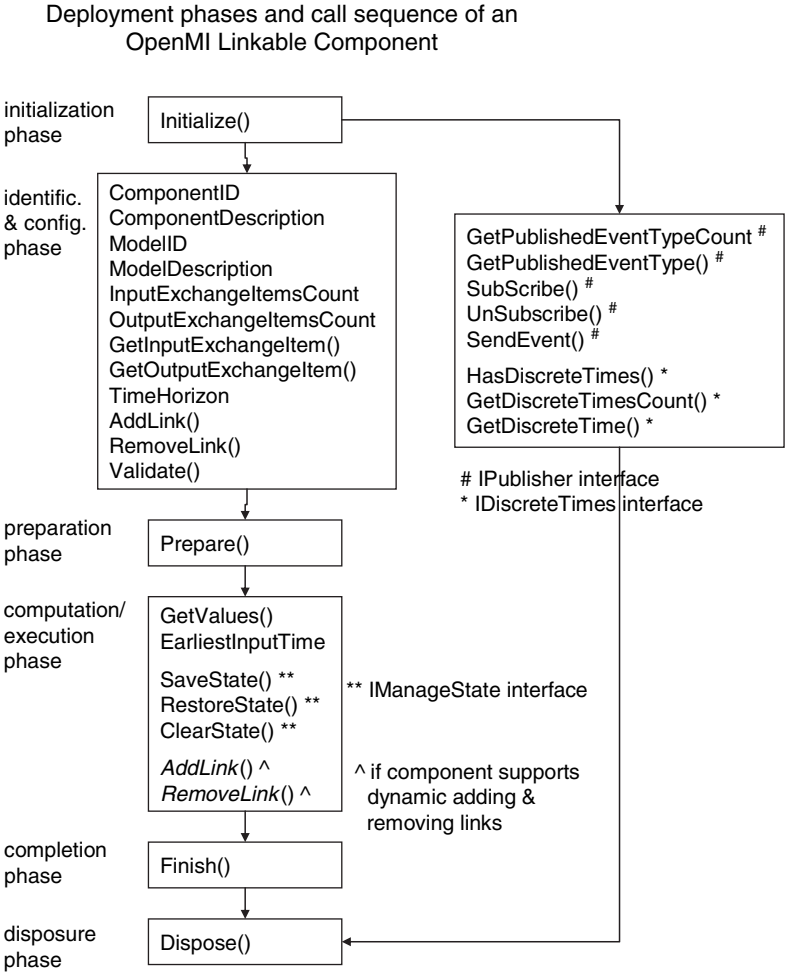


Fig. 32.2 Call phasing of a linkable component

can be identified, and the methods which might be invoked at each phase. While the sequence of phases is prescribed, the sequence of calls within each phase is not prescribed.

The phases are

1. *Initialization*: This phase ends when a linkable component has sufficient knowledge to populate itself with data and expose its exchange items.
2. *Inspection and Configuration*: At the end of the phase, the links have been defined and created, and the component has validated its status.
3. *Preparation*: This phase enables you to prepare all conversion matrices before the computation/data retrieval process starts.
4. *Computation/execution*: During this phase, the main code of the component is executed. Typically, it represents the simulation of a process such as river flow. This simulation may itself generate calls to other components.
5. *Finish*: This phase comes directly after the computation/data retrieval process is completed. Code developers can utilize this phase to close their files and network connections, clean up memory, etc.
6. *Disposal*: This phase is entered at the moment an application is closed. All remaining objects are cleaned and all memory (of unmanaged code) is deallocated. Code developers are not forced to accommodate re-initialization of a linkable component after Dispose() has been called.

Linkable components may support the dynamic addition and removal of links at computation time. However, as described in the grouping of deployment phases, this requirement is not enforced. Those who do not support this call at computation time should throw an exception.

32.3.4 Other Features

In principle, the GetValues()-call stack of all linkable components is located in one thread. Linkable components may internally use distributed computing techniques to improve computational efficiency.

Code developers may create container components holding other components, as long as the container implements the linkable component interface.

By separating various phases of deployment, code developers can choose when to instantiate and populate the engines. Exchange item information can be obtained from the engine, from files being parsed during the inspection phase or by dynamic querying of linked components.

32.4 How Does It Work – An Example

To illustrate how it works, a very simple but complete example, presented in Gregersen et al. (2005), will be used. The example addresses a conceptual lumped

rainfall-runoff (RR) model which provides inflow to a river model. The populated link class is shown in Fig. 32.3.

The sequence diagram in Fig. 32.5 shows the calling sequence for a configuration with a river model linked to a rainfall-runoff model. The diagram demonstrates how things would look if a "hard-coded" configuration was used. For normal usage of OpenMI, a configuration editor would assist you in creating the configuration.

The sequence diagram has the following steps:

1. The river model object and the RR model object are instantiated. Then the Initialize method is invoked for both objects. Models will typically read their private input files when the Initialize method is invoked. Information about name and location of the input files can be passed as arguments in the Initialize method.
2. The river model is queried for InputExchangeItems and the RR model is queried for OutputExchangeItems. The InputExchangeItems and OutputExchangeItems objects contain information about which combinations of Quantities and ElementSets (locations) can be accepted by the components as input or output, respectively.
3. A Link object is created and populated, based on the obtained lists of InputExchangeItems and OutputExchangeItems. In this example, we are using a hard-coded configuration. However, if a configuration editor was used, the OutputExchangeItem and the InputExchangeItems would be selected by the user using, e.g., a selection box.

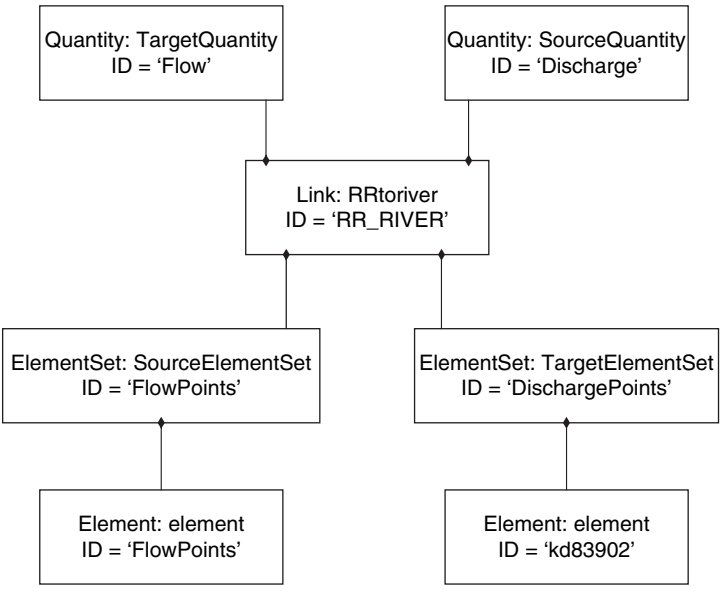


Fig. 32.3 The populated link class

4. The trigger component is created. This component is a very simple LinkableComponent whose only purpose is to trigger the calculation chain.
5. Link objects are added to the LinkableComponents. This will enable the LinkableComponents to invoke the GetValues method in the LinkableComponent to which they are linked.
6. The Prepare method is invoked in all LinkableComponents. This will make each LinkableComponent do whatever preparations are needed before calculations can start.
7. Invoking the RunSimulation method in the trigger object starts the calculation chain.
8. The trigger object invokes the GetValues method in the river model and the river model will calculate until it has reached the EndTime specified in the argument list.
9. Before the river model can make a time step, it must update its inflow boundary condition. In order to do this, the GetValues method in the RR model is invoked.
10. The RR model will repeatedly perform time steps until it has reached or exceeded the time for which it was requested for values. If the river model and the RR model are not synchronous with respect to time stepping, the RR Model must interpolate the calculated runoff in time before the values can be returned.
11. The river model has now obtained its inflow boundary value and can perform a time step. The river model will repeatedly invoke the GetValues() method in the RR model and perform time steps until it has reached or exceeded the EndTime, after which it will return control and values to the trigger object.
12. The trigger object returns control to the main program.
13. The main program will invoke the Finish and the Dispose method in all LinkableComponents. LinkableComponents will typically close output files when the Finish method is invoked. The Dispose method will typically be used by the LinkableComponents to de-allocate memory.

32.5 Meeting This Specification

32.5.1 Consequences of the OpenMI for a Model

The OpenMI enables model engines to compute and exchange data at their own time step, without any external control mechanism. Deadlocks are prevented by the obligation of a component – always to return a value whatever the situation. When each model is asked for data, it decides how to provide it – it may already have the data in a buffer because it has previously run the appropriate simulation steps, or it might have to progress its own calculation, or it might have to estimate via interpolation or extrapolation. If the component is not able to provide all the

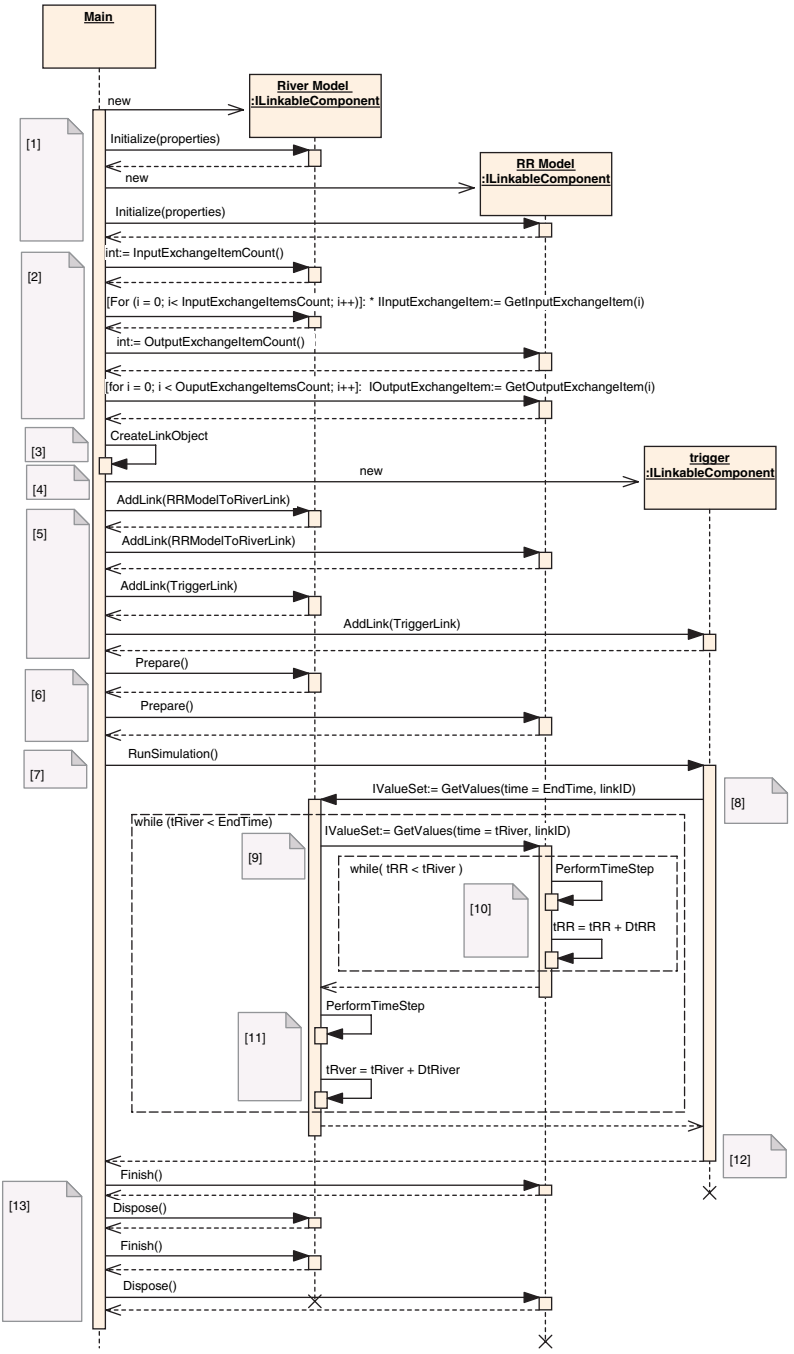


Fig. 32.4 Sequence diagram (example rainfall-runoff – river)

requested data, an exception will be raised. The exchange of data at run-time is automated and driven by the pre-defined links, with no human intervention.

To become an OpenMI linkable component, a model has to

- be able to expose information (what, where) to the outside world on the modelled variables that it can provide, or which it is able to accept;
- submit to run-time control by an outside entity;
- be structured in that initialization is separated from computation – boundary conditions must be collected in the computation phase and not during initialization;
- be able to provide the values of the modelled variables for the requested times and locations;
- be able to respond to a request, even when the component itself is time independent; and if the response requires data from another component, the component should be able to pass on the time as well in its own request;
- flag missing values and
- in the exceptional case that an entire value set is unavailable, throw an exception. Be aware that such exception will stop the entire computation process and thus should be prevented.

32.5.2 Model Wrapping

The above-mentioned requirements do not match the nature of most legacy codes, but most of it can be captured by wrapping. Figure 32.5 illustrates the OpenMI wrapping pattern that is adopted in the implementation of the OpenMI environment.

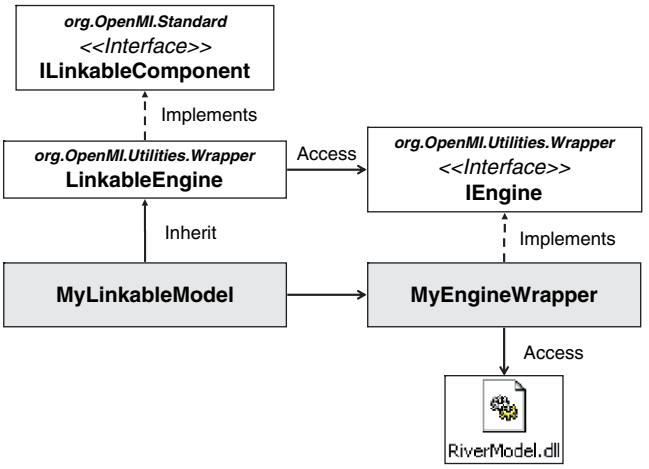


Fig. 32.5 The engine wrapping pattern

<<IEngine>>	
void	Initialize(hastable properties);
bool	PerformtimeStep();
void	Finish;
ITime	GetCurrentTime();
ITime	GetInputTime(string QuantityID, string ElementSetID)
ITimeStamp	GetEarliestNeededTime();
void	SetValues(string QuantityID, string ElementSetID, IValueSet values);
IValueSet	GetValues (string QuantityID, string ElementSetID)
double	GetMissinghValueDefinition();
string	GetComponentID();
string	GetModelID();
string	GetModelDescription();
double	GetTimeHorizon();
int	GetInputExchangeItemCount();
int	GetOutputExchangeItemCount();
IInputExchangeItem	GetInputExchangeItem(int exchangeItemIndex);
IOutputExchangeItem	GetOutputExchangeItem(int exchangeItemIndex);

Fig. 32.6 The internal IEngine interface

The wrapper uses an internal interface to access the engine. This interface, IEngine (see Fig. 32.4), may be split in a section addressing the metadata issues and in a run-time section for the numerical computation and associated data exchange. For the computational part, this wrapping concept still requires that engine cores are re-engineered according to the pattern as illustrated in Figs. 32.5, 32.6, 32.7.

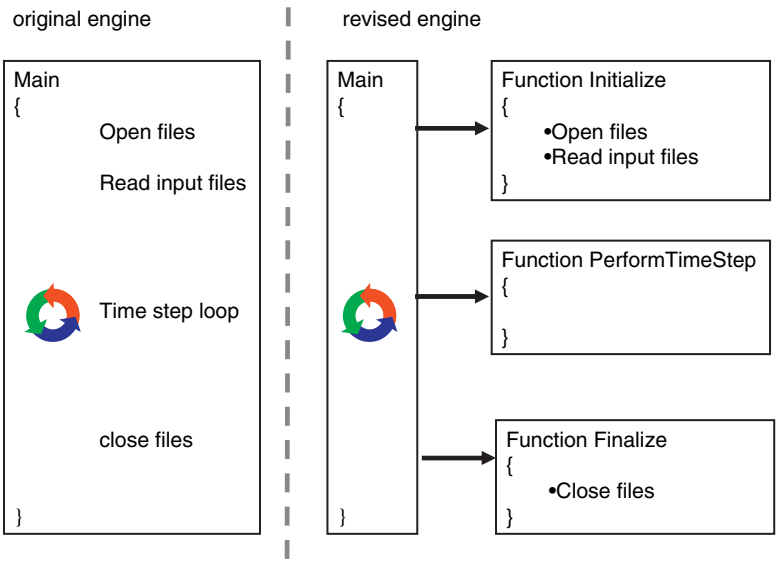


Fig. 32.7 Engine core re-engineering pattern

32.6 Application Experiences

32.6.1 Software Code Migration

OpenMI has been released in the open source under LesserGPL license conditions on Sourceforge (sourceforge.net/projects/openmi). This release includes a C#.NET implementation of the interface standard as well as an implementation of a wide range of utilities to make migration of computational engines easier.

By the time of writing, some six to eight commercial water simulation codes have been migrated, covering the range of catchment, river, rural, urban, lake and coastal applications. In addition, a variety of others are in the process of migration. Furthermore, over ten research and non-commercial codes have been migrated in the water domain, while various migration initiatives are undertaken in the agricultural and nuclear domain. OpenMI has also shown its capabilities as an interface between legacy code and modelling frameworks (e.g. the Invisible Modelling Environment, TIME, an Australian framework, and DelftFEWS, a model independent framework for operational support systems).

Most software developers have adopted the migration pattern as shown in Fig. 32.3, using the utilities as provided on Sourceforge. In order to support the internal IEngine interface (Fig. 32.4) most of them needed to re-engineer their code as shown in Fig. 32.5. The effort involved in the migration has shown to depend on the understanding of object-oriented principles and OpenMI, the structure and size of the code, the methods of implementation and the ambition level (i.e. level of model data made accessible). While the additional amount of code lines is often very limited, the migration work may take considerable effort if a drastic code re-structuring is required. In those cases, the need of restructuring often has shown to be valuable also from the perspective of code structure, improved maintainability and improved flexibility.

Taking all these considerations into account, the basic migration effort has shown to vary from one week to a few months. Most software developers are happy to spend this effort on an improved code structure. In some cases, code access has been a problem. Migrating input or output data file has shown to be an appropriate and easy way to connect inaccessible legacy code to other model codes via a one-way OpenMI linkage. In those cases, reduced performance can be expected.

32.6.2 Performance Issues

While developing the OpenMI, performance has always been an issue carefully considered. If properly defined and implemented, only one GetValues-call is required to transfer data for all linked computational elements. Furthermore, the Prepare method has been introduced to allow data transformation matrixes to be computed just before the simulation starts. Data between linkable components is typically exchanged in shared memory, as all classes implementing the ILinkableComponent

are forced to reside in one single thread. Software developers are free to choose their own implementation methods to obtain high performance within a component. For example, internal parallelization might be applied if the computational effort within a GetValues-call is substantial. Although the utilities as provided in open source have not yet been optimized, tests have shown that limited time is required for buffering, mapping and transferring the data, i.e. in the order of milliseconds or less when a few thousand elements need to be mapped (Gregersen, 2005).

32.6.3 Applications with OpenMI

As a rather new technological opportunity, applications so far have mainly been focused on proving the advantages of a standardized interface to enable model linking. Dudley et al. (2005) have demonstrated for an artificial catchment-lake setting the applicability of OpenMI to link a variety of commercial codes from various vendors to address integrated modelling opportunities for the EU-Water Framework Directive. This demonstration illustrated that once the data can be exchanged, the issues to be dealt with are modelling issues, such as instabilities at the start of the simulation due to incorrect initial conditions for the entire set of models.

While their demonstration dealt with an artificial catchment-lake setting, various applications (of other companies) have shown the advantages of process interaction in real case study areas. Most notably in creating new insight in urban water systems are applications in Greece (Gavardinas et al. 2005) and in Belgium (Brussels, undocumented), both achieved by linking commercially available software products.

In addition to applications in the field of process interaction between bio-physical models, applicability has been proved for linkages to agent-based models, for application with calibration and optimization techniques (HarmonIT, 2005b), for visual DSS development (Dirksen et al. 2005) and for ensemble Kalman filtering (Seyoum, 2005). Within the Framework 6 projects, Seamless, Sensor and Aquastress, OpenMI provides the communication basis for the development of water, land use and agricultural-related decision support systems.

While presenting OpenMI to an audience not involved in development, many outsiders have made reservations on the applicability of OpenMI due to performance issues. The applications so far have made clear that OpenMI increases the capabilities of legacy models to simulate process interaction with other models. Developers of OpenMI consider this achievement, and the opportunities it offers much more valuable than a limited risk of low computational performance.

32.7 OpenMI Now and in the Future

The OpenMI is currently being applied by a range of software developers in the catchment domain (sewers, open channels, hydrology, groundwater, waste water treatment, water quality, socio-economics) and is starting to be adopted by the

agricultural, estuarine and marine domain. These applications have highlighted how little is known about how best to implement integration between different models using a variety of linkages. There is much more work for the hydroinformatics community to do to really understand integrated modelling.

Initially, much of the research and development was carried out by software developers, but specialist model builders are now taking a larger role. New opportunities for modellers have been identified in building and running sets of integrated models, without needing constant input from software developers. It is hoped that this will lead to growth in the activity of integrated modelling, to satisfy the requirements to better represent cause and effect in catchments and other complicated processes.

The concepts of the OpenMI have been shown to be very powerful, but the experiences also have indicated the desire for further refinement of the interfaces to support a persistent state (to enable hot starts), to improve performance, to improve the element set definition for topology and the exact positioning of the data value, to enable clustering of quantities, to address component-to-component connections in relation to the link definition and to simplify the implementation of the interfaces.

It is foreseen that an update of the OpenMI is desirable to obtain a mature standard. A thread is started on the sourceforge.net/projects/openmi forum to stimulate the development discussion on the OpenMI.

Nonetheless, confidence in the OpenMI concept is high enough, even with the initial release, for a number of commercial system developers to incorporate the interface in releases of their modelling products. And OpenMI is starting to be specified as an essential element in the architecture for software systems being built under contract. To support this growing family of users, the core partners of the HarmonIT consortium have committed themselves to set up an open organization (the OpenMI Association) that will maintain the OpenMI and stimulate its development, support and uptake by the modelling software community.

Acknowledgements The authors thank the EU for co-funding the HarmonIT – IT Frameworks project (Contract no: EVK1-CT-2001-00090) that led to the development of the OpenMI.

References

Dirksen PW, Blind MW, Bomhof T, Nagandla S (2005) Proof of Concept of OpenMI for Visual DSS development. In: Zerger A, Argent RM (eds) MODSIM 2005 International Congress on Modelling and Simulation. Modelling and Simulation Society of Australia and New Zealand, Dec. 2005, pp. 184–189. http://www.mssanz.org.au/modsim05/papers/dirksen.pdf

Dudley J, Daniels W, Gijsbers PJA, Fortune D, Westen S, Gregersen JB (2005) Applying the Open Modelling Interface (OpenMI), In: Zerger A, Argent RM (eds) MODSIM 2005 International Congress on Modelling and Simulation. Modelling and Simulation Society of Australia and New Zealand, Dec. 2005, pp. 634–640. http://www.mssanz.org.au/ modsim05/papers/dudley.pdf

Gavardinas C, Fotopoulos F, Gijsbers PJA, Moore, RV (2005) OpenMI: A standard interface for linking environmental models. EnviroInfo conference 2005, Brno, Czech Republic

Gregersen JB (2005) Some notes about OpenMI performance (internal note, see Sourcefourge under source/mySourceCode/DHI)

Gregersen JB, Gijsbers PJA, Westen SJP, Blind M (2005) OpenMI: the essential concepts and their implications for legacy software. Advances in Geo-sciences 4: 37–44

HarmonIT (2005a) The org.OpenMI.Standard interface specification. Part C of the OpenMI Document Series. IT Frameworks (HarmonIT) ECFP5 Contract EVK1-CT2001-00090

HarmonIT (2005b) The org.OpenMI.Utilities technical documentation. Part F of the OpenMI Document Series. IT Frameworks (HarmonIT) EC-FP5 Contract EVK1-CT2001-00090

Seyoum SD (2005) A generic tool for OpenMI-Compliant Ensemble Kalman Filtering. UNESCO-IHE MSc.thesis WSE-HI.05-04

Chapter 33
Hydroinformatics – The Challenge for Curriculum and Research, and the "Social Calibration" of Models

J.P. O'Kane

Abstract The tools of hydroinformatics, understood as advanced commercially maintained, modelling systems, together with their environmental data acquisition systems, are expensive. Nevertheless, young engineers need exposure to them in a critical academic context, if they are to make the best use of them in their future practice. Selecting and matching models and new data to a given problem is the first step. The second is avoiding numerical calibration as much as possible so that the outcome is insight and not merely numbers. Every mismatch between a prediction and a measurement raises the question, why? This may be due to errors in the model, errors in the data, or errors in both the model and the data. There are no other possibilities. Answering such questions improves the model. "Social calibration" of the model and its data involves those stakeholders with the best knowledge of the aquatic system in question. They are shown animated graphical output from the model for historical events and asked if they are true. When the answer is yes, this step builds credibility and acceptance of the model. Only then may we use the model to examine engineering alternatives that affect stakeholders. These thoughts are reformulated as articles of a paradigm for computational hydraulics and hydroinformatics in the university with the goals of better research, curriculum, and textbooks.

The paradigm requires case material for its realisation. Illustrative material is taken from a very extensive study of flooding in the polder landscape of the lower Feale catchment in the southwest of Ireland. The 15 polders of the lower Feale were re-engineered in the 1950s to provide farmland free from flooding that generated high-value agricultural output. However, in the last 50 years, the land has gradually sunk, the marsh plants have returned, and flooding has increased. A large number of engineering interventions were analysed using DHI modelling systems and DLR, Campbell and OTT data acquisition systems, and has lead to several recommendations and conclusions.

Keywords Hydroinformatics · flooding · learning · innovation · social calibration

J.P. O'Kane
Department of Civil and Environmental Engineering, University College Cork, Cork, Ireland,
e-mail: jpokane@ucc.ie

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68,

33.1 Introduction

Twenty years ago it was already clear that digital hydraulic and hydrological laboratories would replace their physical counterparts for many tasks in hydraulic and environmental engineering. Since then, the cost of running a digital laboratory has fallen by several orders of magnitude placing enormous economic pressure on large hydraulic facilities. The software that enables a digital laboratory shows a very large variation in quality between US government-funded freeware and the top commercial systems in Europe from DHI, EDF, WL, and WS. The best commercial systems are now the tool of choice for large and complex civil engineering projects, not only for hydraulic design but also where severe constraints have been placed on the impact of the project on the aquatic environment.

The availability and deployment of hydroinformatics tools was an essential condition for the political acceptability of a number of mega-projects in Scandinavia, such as the Great Belt road- and rail-link project in the 1980s and its successor the Øresund fixed link project (Abbott, 1991). There is no other way to achieve the goal of zero impact in the aquatic environment.

33.2 The Role of the Universities

At the end of the book on computational hydraulics, Abbott and Minns (1998) raise the question of the role of computational hydraulics, and more generally, hydroinformatics, in the university, concluding that "Universities should concentrate on transmitting, producing, and refining intrinsic knowledge, and not commercial codes. The main benefit to the Universities [of commercially available hydroinformatics tools] … is that they can reduce greatly, the amount of routine and repetitive work used to obtain working models for study [and research] purposes by using professional, industry-standard modelling and other tools, tool-sets, languages, and working environments. By these means they are able to shift the focus of their work more intensively to matters of *intrinsic value* this being the value that this knowledge is experienced to contribute to the quality of life of the individual". It stands in contrast to "the *social value* of this same meaning-content of knowledge, … this being the value that society as a whole perceives this knowledge to be worth in material terms, such as is commonly represented in terms of an 'exchange value' or scarcity value. … For example, a particular form of the Boussinesq equations, best suited to representing phenomena such as surf beats and oscillations of the wave-breaker line, has a quite limited value in the eyes of society at large, but the value of the knowledge which they provide concerning the design and construction of a harbour may have a very considerable value indeed." Consequently, "the aim of University teaching and research, even when it makes use of the electronic encapsulation of knowledge, is to arrive at a deeper understanding as a means of increasing the intrinsic value of knowledge" (Abbott and Minns, 1998; pages 504, 513 and 515).

33.3 A Paradigm for the Universities

The best modelling systems represent an effort on the order of at least a hundred person-years of work. In addition, the experience of hundreds of applications to real problems in a commercial environment supported by advice from an experienced agent network, and regular user-group conferences, has continuously improved the best modelling systems. A labour force of graduate students cannot compete with this. Consequently, the first article of the paradigm for the university is to engage in the virtuous circle of modelling in the appropriate way.

33.3.1 The Virtuous Circle of Modelling

The virtuous circle begins by identifying the best modelling systems for different types of problem, acquiring them and learning to use them by applying them to solve real problems. From this it is clear that data are the sine qua non of modelling in practice, the second article of the paradigm. The circle will be closed at the end of this section.

33.3.2 Matching Data and Models

The second article of the paradigm recognises that model's proclaim, ex ante, what data must be collected. Ex post, the same model interprets the data. The sampling frequencies at selected points in space are the key decision variables in the design of the data collection campaign. Many environmental monitoring agencies continue to build useless databases because the sampling interval does not resolve the high-frequency components of large amplitude, aliasing the frequency spectrum of the signal of interest. A sampling interval of 15–30 min, or less, is essential in tidal waters. Off-the-shelf, analogue-to-digital instruments can deliver this frequency for many physical variables. But it is still a major research challenge in the case of important chemical determinands, such as dissolved nutrients, for which there are no ion-specific electrodes and for which wet analytical methods are required.

In the hierarchy of data, geometric data are most important, then data on the forcing of the model within its domain, and finally data on the boundary conditions. Boundary conditions cannot be disassociated from the model itself because they may be interpreted as a simplified model of the water body outside the boundary. Remotely sensed snap-shots of physical variables over the model domain provide the initial, intermediate, and terminal conditions of the model.

33.3.3 Parsimonious, Physical, and Social Calibration

The third article of the paradigm is parsimonious calibration of model parameters. In the case of groundwater levels and water potential in the unsaturated zone of the

soil, for example, the inverse problem of calibration cannot be avoided because underground spatial observations in three dimensions are very difficult to make. Resistivity surveys, ground-penetrating radar, and point boreholes provide an invaluable but still incomplete picture. Inverse problems are unstable in the sense that small changes to the data can lead to large changes in the values of the identified parameters of a model. Parsimonious calibration demands that we minimise this and avoid it whenever possible. In the case of surface water levels, *a goal of no adjustment of model parameters* is often appropriate; the mismatch between predictions and measurements raises the fundamental questions of the fourth article. See below.

The third article of the paradigm also recommends "social calibration" of the model and its data. In this step, those stakeholders, farmers, pilots of boats and vessels, etc., with the best knowledge of the aquatic system in question, are shown animated graphical output from the model for historical events and asked if they are true. When the answer is "yes", it builds credibility and acceptance of the model. In several cases undertaken by the author and his students, both parties learn from this experience, each pointing out features in the output that were unknown to the other party. In such cases, the act of presentation in a social context facilitates the emergence of new insights. See below. This process may be repeated a number of times leading to several revisions to or adjustments of the model; when these are small, or minor in extent, it is appropriate to speak of *parsimonious social calibration* – the ideal.

33.3.4 Insight

The fourth article of the paradigm is the goal of insight. The purpose of modelling is not simply numbers, however they are displayed, but insight! Consequently, every mismatch between measurement and prediction is subjected to the three questions that exhaust all possibilities: Is the model wrong? Are the data wrong? Are both the model and the data wrong? The first question builds on the analysis of simplified models, especially the properties of the numerical scheme. The examination begins with the linearised form and questions its stability, consistency, and convergence, its attenuation of amplitude and shift in phase, and is followed by its extension to the case when non-linearities and boundary conditions are present. The scalar wave equation, the simplest model of advection, is the point of departure in many of these analyses. Simplified re-scaled models can address the question whether physical and chemical processes not in the model should be included.

Measurement systems require the same degree of concern, but concentrated on questions of accuracy, precision, and aliasing. It is still very rare for each individual measurement to be calibrated against a standardised reference quantity in a telemetered instrument providing its own quality assurance. This is especially true for chemical determinands. The second question "are the data wrong?" or more precisely "can the necessary data of the required quality be acquired at all?" is currently a bigger challenge for research compared to the first question about models built on conservation laws.

33.3.5 Closing the Virtuous Circle of Modelling Through Reflection

The fourth article of the paradigm closes the virtuous circle through reflection, confirming the best modelling and measurement systems for different types of problem, and the lessons learnt from their use in both academic and social environments to solve a small number of selected real problems. This process of reflection is different to that in the commercial environment in the case of (a) the ICT system or instrument developer – the toolmaker and (b) the consulting engineering practice or government agency – the tool user, to use Cunge's distinction. In the first instance, academic reflection is not constrained by project or release deadlines. Academic reflection also has different goals: (a) to challenge the tool maker to make better tools that go beyond current best practice and (b) to challenge the tool user to make better use of existing tools. The first set of goals may be expected to produce research topics for graduate students that go beyond the state of the art in both instruments and ICT systems! The second set of goals will produce better textbooks to service a modern curriculum for students of computational hydraulics and hydroinformatics. Without access to the best tools and the freedom to test them in a research environment on real problems, academic engineers will be unable to achieve these goals.

33.4 Case Material

The paradigm requires case material for its realisation. Illustrative material is taken from a very extensive study of flooding in the polder landscape of the lower Feale catchment in the southwest of Ireland (Martin, 2002; Migliori, 2004). See also Cunge et al. (1980).

33.4.1 The Problem

A winter flood event in the polderised landscape of the lower Feale is illustrated in the aerial photograph taken from the southwest, with a schematic diagram of one of 73 sluiced culverts used for drainage (Fig. 33.1).

The problem is to re-engineer the system to mitigate surface flooding. In the first phase of the project, an area of 230 km^2 was modelled covering all 15 polders. Dredging was proved not to solve the problem, but pumping would. In a second phase, a pumping station was installed in one polder with the adjacent polder acting as a control. The problem is to re-engineer the system to mitigate surface flooding (Fig. 33.2).

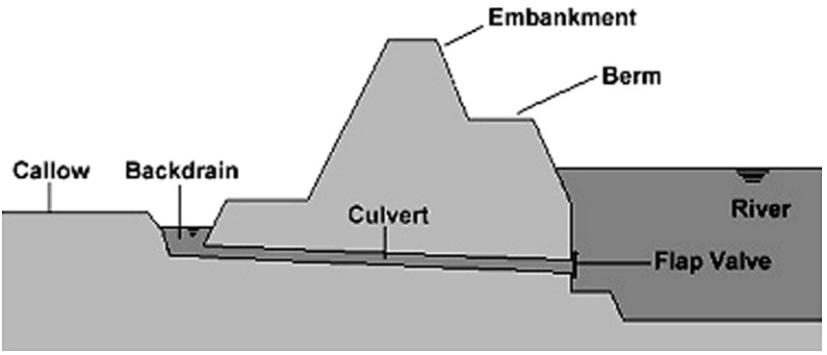


Fig. 33.1 The tidal river and its two backdrains are shown on the left with a section on the right

33.4.2 Instruments and Data

The geometry of the ground surface was obtained by acoustic depth surveys in the tidal channels, dGPS traverses, and two over-flights with the HRSC-A high-resolution, digital, multi-spectral (R,G,B,nR) stereo-camera developed by the German Aerospace Agency (DLR) for the mission to Mars. This produced a geo-referenced DEM on a horizontal grid of 1 m with 15 cm resolution in the vertical direction and a relative accuracy of 20 cm in all three directions. A piece of the DEM is shown in Fig. 33.3. Figure 33.4 shows one of the underground resistivity sections which were calibrated with borehole measurements.

OTT and Campbell instruments provided measurements of surface and ground-water levels, and evapo-transpiration, for the numerical model. Examples are shown in Figs. 33.5 and 33.6.

33.4.3 The Models

The second-phase model (Migliori, 2004) has two sub-models. The first sub-model is an extension of the first-phase model (Martin, 2002). They were built using DHI's Mike11 and MikeSHE modelling systems. The components are outlined in Table 33.1, and their domains are shown in Fig. 33.7.

In Fig. 33.7 on the right-hand side, the triangles represent culverts on streams that pass under a road separating the polders from the upland catchment, and sluiced culverts through the embankments that protect the low-lying land from inundation by the tide or the River Feale. The square is the location of the pumping station and SVAT (soil–vegetation–atmosphere–transpiration) instrumentation. The two circles are the location of pseudo-culverts, one for each polder that accounts for leakage from the tidal channels into the backdrains. All these hydraulic devices are present in the model.

Fig. 33.2 The pumped [C2M] and control [C23] polders in the top left corner of the set of 15

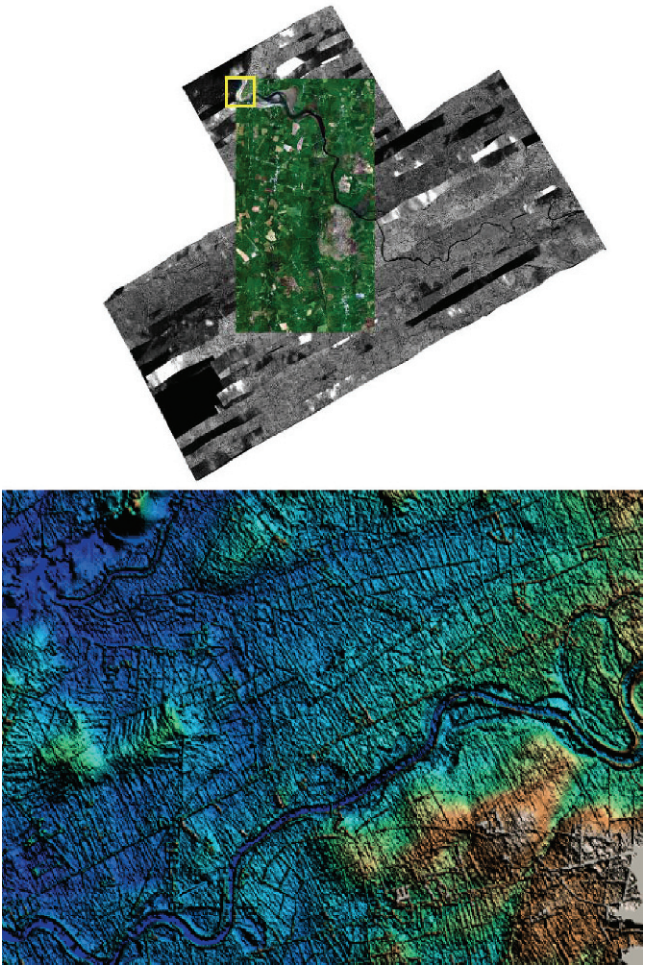


Fig. 33.3 The two DLR over-flights on the left, and part of the high-resolution DEM on the right showing oxbows on the Feale as it enters the polder area

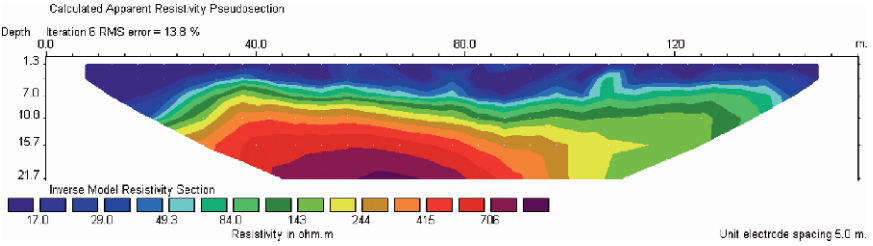


Fig. 33.4 Resistivity section in polder C2M

Table 33.1 The two sub-models and their components (second phase of the study)

Model	Model component	Simulates:	Dim.	Governing equation	Dynamic coupling
Sleveen – Net Model	River network (RN)	Fully dynamic river and drains hydraulics (flow and water level)	1-D	De Saint-Venant's equation (dynamic wave approximation)	SZ, OL
Sleveen Polder–Upland Model	Overland flow (OL)	Overland flow and water depth, depression storage	2-D	De Saint-Venant's equation (diffusive wave approximation)	SZ, UZ, and RN
	Unsaturated zone (UZ)	Flow and water content of the unsaturated zone, infiltration and groundwater recharge	1-D	Richards' equation	SZ, OL
	Saturated zone (SZ)	Saturated zone (groundwater) flow and water levels	3-D	Boussinesq's equation	UZ, OL, and RN
	Evapo-transpiration (ET)	Soil and free water surface evaporation, plant transpiration		Kristensent and Jensen equation	UZ, OL
	Pre-/post-processing	Pre- and Post-processing			

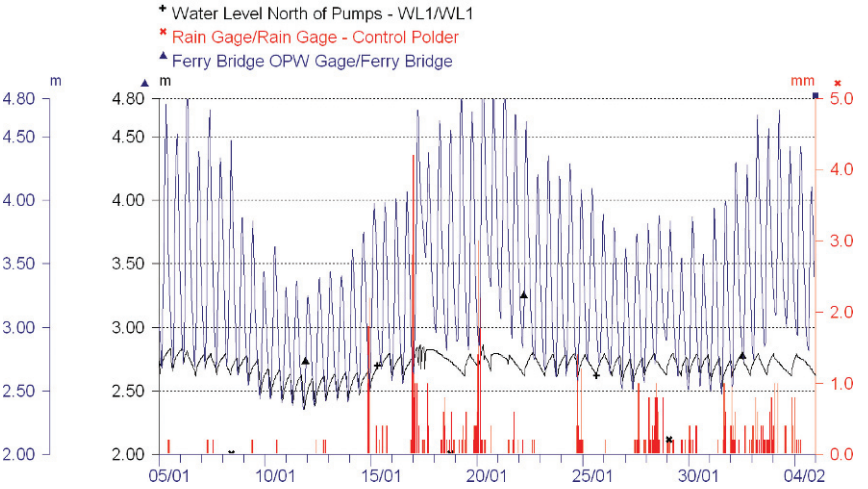


Fig. 33.5 High-frequency measurements of surface water levels

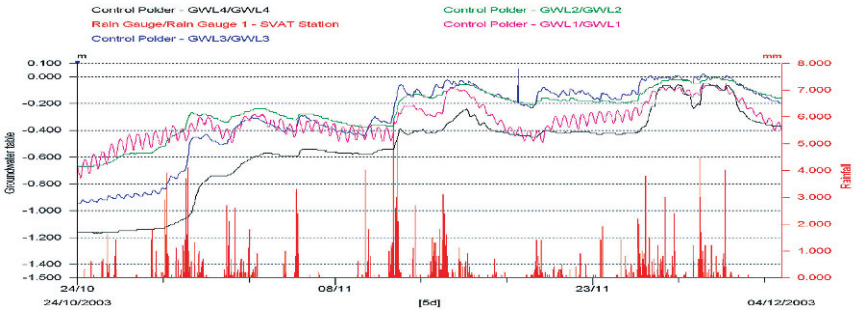


Fig. 33.6 High-frequency measurements of groundwater levels following a dry period. A tidal signal is present in groundwater gauge GWL1; the ground acting as a lowpass filter when levels are low

33.4.4 Insight – Learning from the Models and the Data

The interaction between the pumps, rainfall-runoff, evapo-transpiration, tides and flow in the River Feale is a complex set of feedback processes which can be examined in detail with the model. The model also predicts the variation in groundwater level, surface ponding and runoff within the two polders, information of value to farmers, the principal stakeholders. Figure 33.8 shows the agreement achieved after considerable calibration. In contrast, very little calibration was required in the first-phase model where significant deviation between predictions of tidal water levels and measurements lead to the discovery of an usual set of localised errors in the data associated with the change between summer time and winter time.

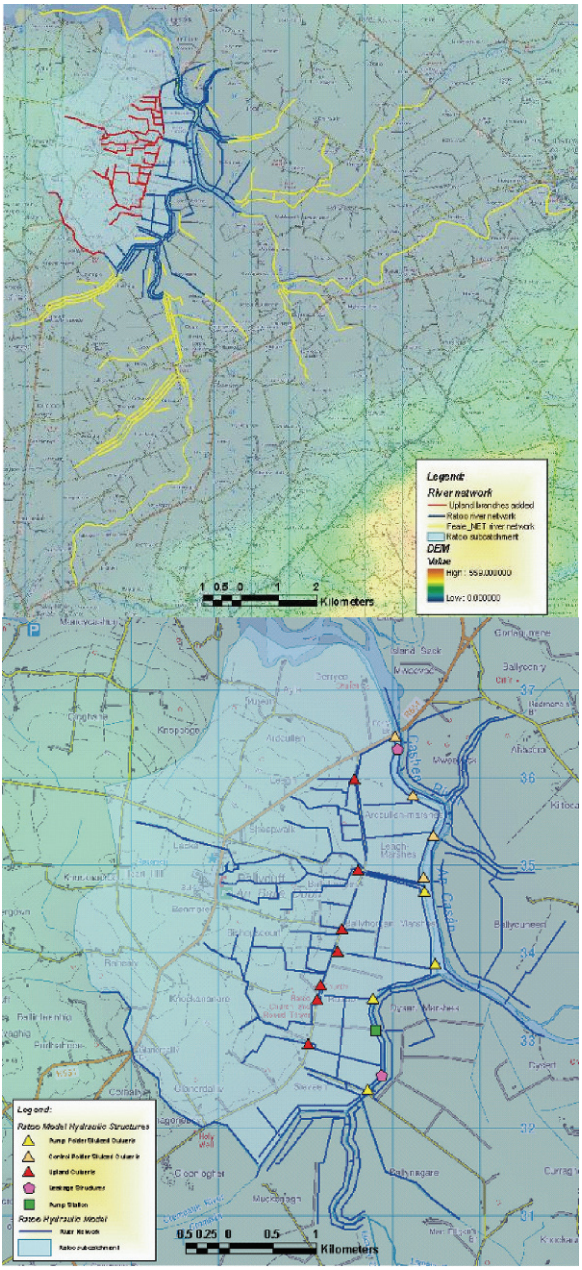


Fig. 33.7 The yellow and blue network on the left is the model from the first phase of the project (Martin, 2002). The blue and red network is from the second phase model (Migliori, 2004)

Fig. 33.8 Comparison between predicted (*black*) and observed (*blue*) groundwater level at station GWL3. The range on the scale is 1.2 m. When the level exceeds 0.0 m, surface ponding is present

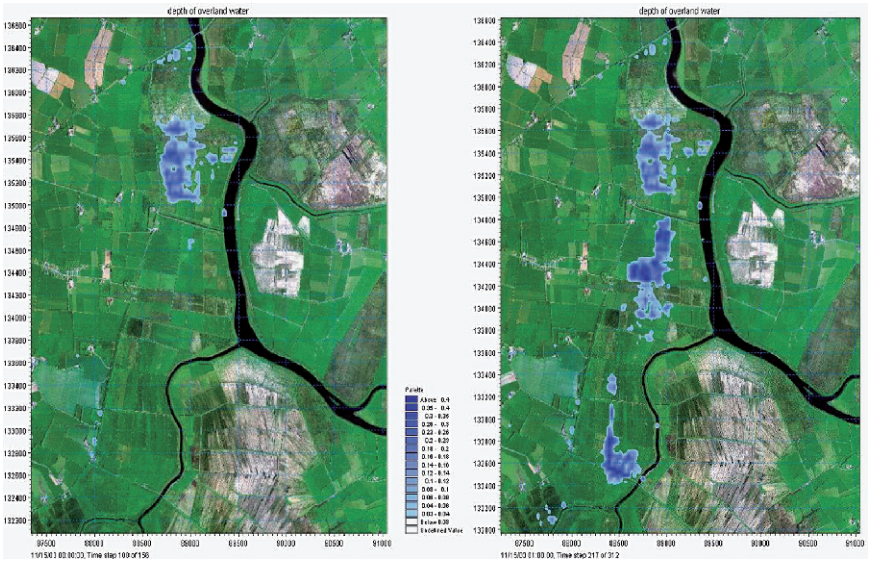


Fig. 33.9 Two snapshots from an animation at the same point in time: one with the pumping station, the other without the station. It shows the effectiveness of pumping at the moment of maximum flooding when pumps are not present during a test storm

Animated flood maps were also produced for test storms. Two snapshots are shown in Fig. 33.9. In the first phase of the project, similar animations, but calculated in a much simpler manner without interaction with groundwater, were shown to farmers. They were asked to find the fields on their farms that flood in the bird's-eye animation and to judge the realism of the animation for a recent flood event. This lead to the discovery of minor errors and improved the model.

33.5 Conclusions

The chapter illustrates the paradigm for hydroinformatics in the university by showing

- The importance of acquiring the best hydroinformatics tools in an academic environment, for use on real problems without time constraints
- The necessity of combining high-quality data with state-of-the-art models
- How insight comes from systematic comparison of model predictions and measurements while minimising parameter calibration
- How to use social calibration to profit from the experience of stakeholders, building credibility for the models
- In this particular case that a third-phase model based on ODEs and using output from the earlier models based on PDEs may deliver new insights – this work has recently started with the goal of using wind mills to dewater the polders
- The importance of case material for new curriculum and textbooks in computational hydraulics and hydroinformatics. See for example Cunge et al. (1980), Abbott and Basco (1989) and the papers in the international Journal of Hydroinformatics.

Acknowledgements The illustrations in this chapter are the work of two PhD students whose theses are acknowledged in the list of references. Financial support was provided by the Office of Public Works, Dublin, by the Programme for Research in Third-level Institutions (PRTLI3) through the UCC Environmental Research Institute and by private donation through the Cork University Foundation.

References

Abbott MB (1991) Hydroinformatics – Information technology and the aquatic environment. Avebury Technical: Aldershot

Abbott MB, Minns AW (1998) Computational Hydraulics. Second Edition. Ashgate: Aldershot

Abbott MB, Basco DR (1989) Computational Fluid Dynamics – An Introduction for Engineers. Longman: London

Cunge JA, Holley FM Jr, Verwey A (1980) Practical Aspects of Computational River Hydraulics. Pitman: London. Reprinted by Institute of Hydraulic Research: University of Iowa

Martin J (2002) De-watering the Lower Feale Catchment – "A Virtual Water World", PhD Thesis, The National University of Ireland: Cork

Migliori L (2004) The Hydrology and Hydraulics of a Pumped Polder – a Case Study in Hydroinformatics. PhD Thesis. The National University of Ireland: Cork

Chapter 34
A New Systems Approach to Flood Management in the Yangtze River, China

H. Betts, S. Markar and S. Clark

Abstract The Yangtze River Flood Control and Management Project (YRFCMP) was established following an approach by the Government of the Peoples Republic of China to the Australian Government in 1998 seeking assistance to improve its flood management systems for the Yangtze River. The project was approved and commenced under the auspices of the Australian Agency for International Development (AusAID) through the Australian Managing Contractor (AMC – SAGRIC International Pty Ltd) and the Changjiang (Yangtze) Water Resources Commission (CWRC).

To achieve the project objectives of increasing the accuracy of flood forecasts, extending flood warning time, and improving the flood management decision making, the project developed a series of integrated hierarchical "intelligent" systems. These included data acquisition, data transfer and management, flood forecasting, numerical modelling, hydro-metrological, flood management information, and options analysis systems. These "intelligent" systems are encapsulated into an overall decision support system (DSS) and accessed through a web-based visual display system. The DSS integrates with other Chinese complementary data and information systems and CWRC's daily operations and flood management system.

Keywords Yangtze River · flood management · integrated hydroinformatic systems · flood forecasting · options analysis · decision support system

H. Betts
SAGRIC International Pty Ltd.,
16 Carlisle Court, Hallett Cove, SA 5158 Australia

S. Markar
WRM Water and Environment Pty Ltd,
Level 5, Paddington Central, Brisbane, Qld., 4064 Australia

S. Clark
Water Technology Pty Ltd,
Unit 15 Business Park Drive, Notting Hill, VIC, 3168 Australia

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

34.1 Introduction

China has experienced massive floods in the Yangtze basin for centuries and is a serious source of concern for the Chinese authorities. The United Nations estimated over 3,000 people lost their lives in the disastrous 1998 floods when some 15 million people were rendered homeless, 5 million houses were destroyed, 22 million hectares were inundated, and 1.8 million hectares of crops totally destroyed. The total damage bill was estimated to exceed $US 20 billion (UNESCAP, 1999). Following these floods, the Chinese government sought Australia's assistance to identify potential methods to improve its flood management capability in the upper and middle reaches of the Yangtze basin between Chongqing above the Three Gorges Dam and thence down to Jiujiang as indicated in Fig. 34.1.

A Design Mission sent to China identified two key problems: the uncertainty in flood management decisions and the less than desired accuracy and lead time in forecast flood discharges and water levels (GOA and GOPRC, 2000). It was perceived that an "integrated and coordinated design that readily incorporates and digests real-time, forecast and historic flood data" was needed. This meant the development of "an integrated flood forecasting system; a flood response protocol system that defined response activities; a flood damage estimation system that evaluated the socio-economic costs of flood management options; a simple and effective internal consultation system for decision makers; and an effective visual display system at multiple locations that can interrogate, analyse and present underlying data

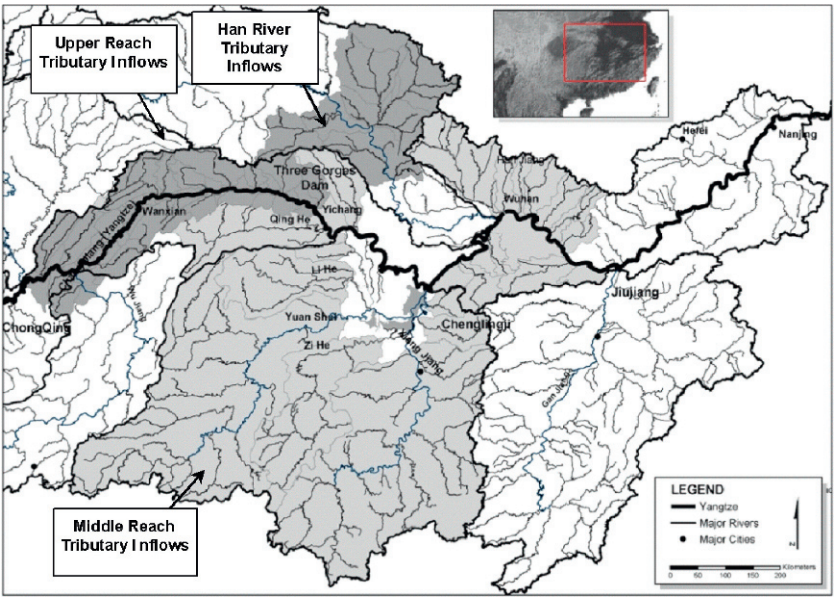


Fig. 34.1 Project area

to decision makers". The prime objective was to "develop an effective reliable and user-friendly decision support system for the upper and middle reaches of the Yangtze River". As this Hydroinformatics approach was new to China, it was set up as a demonstration project.

The overall project commenced development in Wuhan in March 2001 and was accepted as completed by the CWRC and AusAID in October 2005. This work (Betts et al., 2005) included the supply and installation of automatic water level and rainfall recorders which transmit their data by either satellite or telephone network to regional sub-centres – the design and construction of a microwave link between Danjiangkou and Xianfan in the Han River catchment, supply and installation of computer servers and database systems at CWRC headquarters in Wuhan and at 13 sub-centres, supply of 20 DGPS for river gauging and a mini-computer for climate modelling, the development of a new flood forecasting system (Markar et al., 2005) incorporating and linking a series of hydrologic and hydraulic models (Clark et al., 2004, 2005), and a decision support system (DSS).

34.2 Background

The Yangtze River, or Changjiang as it is known in China, 6,300 km long, is the longest river in China and the third longest in the world. The Yangtze links west and east China from the Tibetan Plateau to its estuary near Shanghai. The river provides excellent navigation in the lower and middle reaches, which are centres for agriculture and industrial activities and contribute more than 40% of the annual Chinese gross national product. The region is continuing to expand and grow.

Each flood season, the CWRC in conjunction with other State and Provincial agencies has to manage peak flows within the Yangtze main stem and ensure peak water levels and discharges do not exceed prescribed limits. Decisions may have to be taken to change the operation of key reservoirs, and when those options are no longer possible, perhaps divert flood peaks into any of up to 35 detention basins along the middle reaches of the Yangtze and some of its main tributaries. The most important of these basins for flood control is the Jingjiang Basin, which covers some 920 km^2 and is home to over 540,000 people. Flooding some of these basins involves the evacuation of inhabitants and providing alternative housing for several months and incurs extensive flood damage to agricultural land, villages, homes, and urban and rural infrastructures.

There are significant social and financial costs associated with operating detention basins, especially if the basins are opened unnecessarily (Betts et al., 2005). Conversely, if the basins are not opened during extreme flood conditions, significant risks exist that levees along the Yangtze could be breached at critical locations. If a levee fails in an uncontrolled way, then generally more extensive social and financial hardship is imposed on the affected communities. Deciding to flood any detention basin is a critical flood management decision. Such decisions become increasingly more difficult as floodwaters approach critical levels. Relevant decision support,

based on forecast scenarios, is required to guide the selection and implementation of the most appropriate action.

Improved understanding of the Yangtze basin river system behaviour has been achieved by enhancing data collection and modelling techniques to strengthen flood forecasting in terms of lead time and accuracy. The project has improved the precision and reliability of flood flow predictions and provides earlier warnings of impending flood risk. These in turn facilitate the flood control decision-making process for the selection and operation of reservoirs, warning and evacuation of those at risk, and the flooding of detention basins.

34.3 Information Systems Overview

The DSS developed by the project can be said to be vertically integrated in terms of data collection, information development, and presentation. However, it can also said to be horizontally integrated in terms of its functionality and absorption into the work practices of the CWRC's Bureau of Hydrology (BOH) and River Management Bureau (RMB) as outlined in Fig. 34.2.

The DSS relies on both recorded information and a series of modelling outputs that use hydro-meteorological data and flood management data stored in two separate databases. Operating within a web-based environment (using Microsoft's Internet Explorer) requires different approaches depending on the complexity of the respective information tasks.

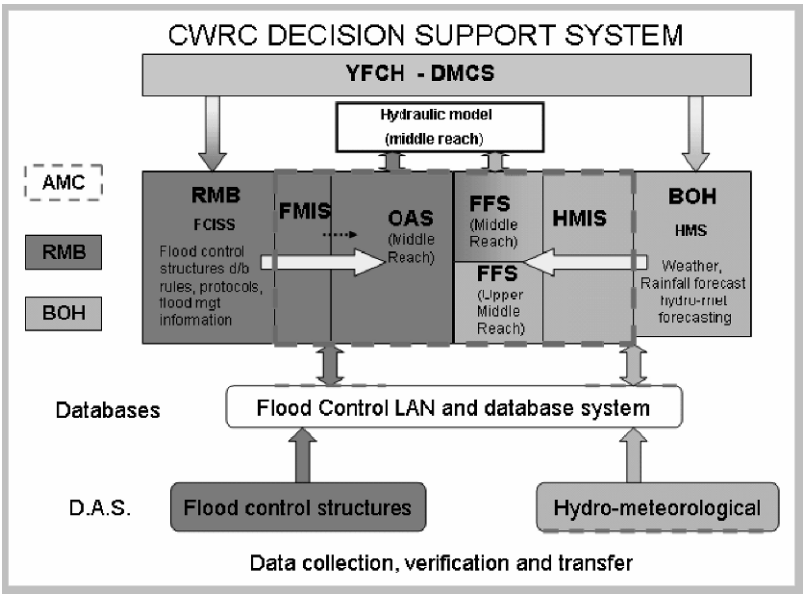


Fig. 34.2 Decision support system schematic layout

Acronyms used in Fig. 34.2 include

- YFCH – DMCS is the Yangtze Flood Control Headquarters Decision Makers Consultation System
- AMC is Australian Managing Contractor
- RMB is the River Management Bureau of the CWRC
- BOH is the Bureau of Hydrology of the CWRC
- DAS is data acquisition system
- FCISS is an existing RMB flood control information service system
- HMS is an existing BOH hydro-meteorological system

Component systems of the DSS are

- HMIS is the hydro-meteorological information system
- FFS is the flood forecasting system
- FMIS is the flood management information system
- OAS is the options analysis system.

34.4 System Architecture

The overall system can be accessed by client machines through a web server that interrogates tables in two databases (Sybase in a UNIX environment) as shown in Fig. 34.3.

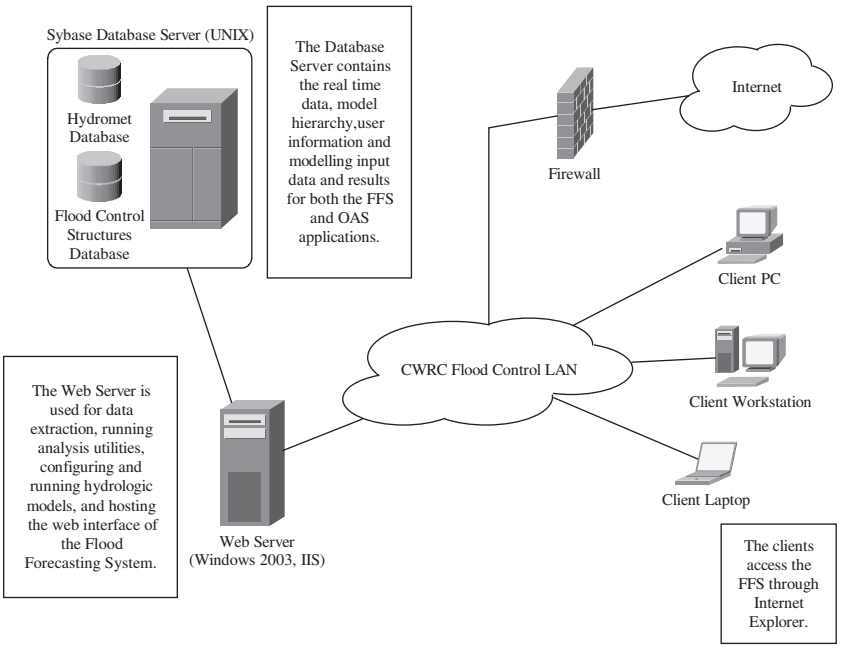


Fig. 34.3 System architecture

- The DSS website application is written using Active Server Pages (ASP) technology and hosted on a Windows 2003 server running Internet Information Services Version 6;
- The model data, utility programs, and static files required by the applications are stored on a separate share on the same web server that serves the website;
- The hydrological and flood control databases are Sybase Adaptive Server Enterprise (ASE) databases stored on dedicated HP-INIZ servers;
- The systems can run on any version of the Windows operating system with the ability to run Internet Explorer Version 5 or higher;
- The client systems require the MIKE-11 modelling package (DHI), Microsoft. NET framework v1.1, and Sybase ASE OLE-DB driver configured to connect to the database server;
- Users of the flood forecasting system (FFS) contained within the DSS can run a series of 120 hydrologic models in series. The modelling is undertaken on the web server with results being written to appropriate hydro-meteorological database tables. In this system, the web server retrieves real-time records, assembles the boundary files, and runs the models;
- The FFS also routes the results of the hydrologic forecast modelling through three MIKE-11 hydraulic models using a purpose-designed and constructed hydraulic modelling interface. Due to the size of the models, run time and data transfer constraints, this interface is installed on local client machines together with the MIKE-11 software. The interface retrieves hydro-meteorological data in forecasting mode, assembles boundary condition files, runs hydraulic models, and writes selected time series results to the database tables;
- The options analysis system also relies on hydraulic modelling but in this system is used to analyse the hydraulic impacts of changed reservoir operations and/or the deliberate flooding of one or more of 35 detention basins.

Further information on the each of the developed systems is contained in the sections that follow.

34.5 DSS Operational Overview

34.5.1 General

The DSS consists of four sub-components: a hydro-meteorological information system (HMIS), a flood forecasting system (FFS), a flood management information system (FMIS), and an options analysis system (OAS). This chapter discusses the HMIS, the FFS, and the OAS, the key outputs of the Yangtze DSS built by YRFCMP.

Both the FFS and the OAS are underpinned firstly by the data acquisition and communications systems (Betts et al., 2005), and secondly by numerical hydrologic and hydraulic modelling systems. More specifically, the FFS enables real-time

hydro-meteorological data obtained throughout the Yangtze catchment to be utilised by a series of hydrologic and hydraulic numerical models to produce 7-day forecasts of discharge and water levels throughout the upper and middle reaches of the Yangtze River. On the basis of these forecasts and pre-defined warning, alert and critical levels at various points in the Yangtze system, the OAS enables the identification of potentially dangerous flooding conditions, and it allows the effectiveness of various mitigation options available to managers to be investigated. The first of these mitigation option measures to be investigated would typically be the use of reservoirs to attenuate flooding from upstream areas. If these measures are not sufficient, the effectiveness of operating detention basins in addition to reservoir operations would be investigated. Once the decision has been made to actively utilise mitigation measures (through the OAS), the impact of such mitigation measures must be considered in future forecasts by the FFS.

The use of numerical models to provide this functionality is described in more detail in the following sections.

34.5.2 Hydro-meteorological Information System

The HMIS can display real-time, historic flood and forecast water levels, flows, and rainfall in map, graphical, or tabular form. Users have the ability to compare current and historic information, create isohyetal maps, view weather links, calculate flood volumes between dates or above user-defined flood levels or flows or critical flood levels. Station icons are colour coded to indicate the severity of the prevailing conditions. A calendar function/date picker also indicates the severity of water levels and is used to identify critical periods of flooding when searching flood records. The mapping system used in the DSS was especially developed to enhance download speed. It has the usual mapping functionality, and it uses colour coded icons and pop-up boxes to display data from the database thereby avoiding screen clutter.

34.5.3 Flood Forecasting System

34.5.3.1 General

The Yangtze River FFS is a web-based and fully integrated flood forecasting system, the heart of which is the flood forecasting models, both hydrologic and hydraulic. The flood forecasting system is described in detail in Markar et al. (2005).

Hydrologic models are used to forecast the tributary and local inflows into the main stems of the Yangtze and Han Rivers, and the Dongting Lake System. Hydraulic models are then used to forecast discharge and flood levels along the main stems of the Yangtze and Han Rivers, and the Dongting Lake System. The FFS links all the hydrologic and hydraulic models used to forecast discharge and water levels

in different parts of the Yangtze catchment into an integrated single catchment wide system.

To date, over 120 hydrologic models for contributing tributary catchments and 3 hydraulic models for the Yangtze main stem and the Han River have been incorporated into the FFS to model the hydrologic and hydraulic behaviour of the upper and middle reaches of the Yangtze River. Configuration data for these models, together with real-time hydro-meteorological data and rainfall forecasts are extracted from a central database and converted to model input files. The models are then run and the output files are stored, compared, and final forecast discharge and level scenario selected and managed within the above fully integrated web-based FFS framework.

34.5.3.2 Hydrologic Models

Figure 34.1 provides an overview of the contributing tributary catchments for which hydrologic models have been constructed for the upper and middle reaches of the Yangtze River.

From an operational perspective, the hydrologic models have been established within the FFS in such a way as to provide the following key operational features and functionality:

- Numerous tools for checking data from manual and automatic rainfall stations;
- Ability to vary the forecast period (7 days used as default);
- Ability to run no forecast rainfall, adopted forecast rainfall, and user forecast rainfall scenarios;
- Use "hotstart" files to minimise time associated with producing forecasts;
- Ability to rapidly run multiple hydrologic models within a forecast area for comparison purposes; and
- Numerous tabular, graphical, and map formats available for checking and comparison purposes.

34.5.3.3 Hydraulic Models

Figure 34.4 provides an overview of the middle reach hydraulic model (MRHM). The hydraulic model for the upper reach consists of a single channel extending from Chongqing to the Three Gorges Dam.

For real-time, operational forecasting purposes, the Danish Hydraulic Institute (DHI)'s MIKE-11 hydraulic modelling system has been utilised (http://www.dhigroup.com/Software/WaterResources.aspx). A custom built interface has been established to manage the operation of the hydraulic models. In essence, the interface is a tool that simplifies and automates the use of the hydraulic model in a real-time operational context.

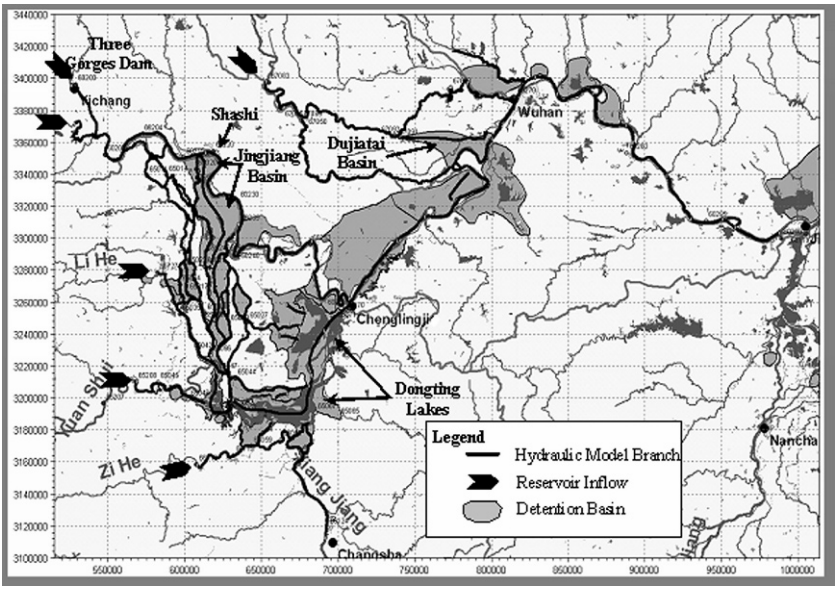


Fig. 34.4 Middle reach hydraulic model

From an operational perspective, the hydraulic models have been established within the FFS in such a way as to provide the following key operational features and functionality:

- Automated boundary condition extraction of either tributary inflows or direct runoff as predicted by the hydrologic models;
- Advanced tools for reviewing and modifying (if necessary) boundary conditions, both during the hindcast and forecast periods;
- Ability to accurately reproduce the large variation in water levels (in excess of 60m+) experienced through the Three Gorges section of the upper reach;
- Ability to handle hydraulic structures with complex operating regimes (e.g. the Three Gorges Dam and the entry gates to the larger detention basins);
- Ability to accurately reproduce the effects of the large lake systems (the Dongting Lakes) present within the middle reach;
- Facilities to automate the execution of the hydraulic models, including all potential variations in hydrological scenarios (e.g. no forecast rainfall, adopted forecast rainfall, and user forecast rainfall), for a number of different hydrologic modelling systems (e.g. any combination of URBS, XAJ, API models);
- Use of "hot start" files to minimise the time associated with producing forecasts;
- Facilities to utilise real-time observational data from key gauging stations (flow and level observations) to "update" the hydraulic model predictions in order to minimise model errors;

- Advanced, graphical results viewing facilities to enable users to check the simulations (including real-time observations), compare the results of varying simulations, and once satisfied, export the adopted forecasts to the database.

34.5.3.4 FFS Outputs

The key output of the FFS is a set of predictions (both discharge and level) at over 70 gauging stations throughout the upper and middle reach of the Yangtze River system. These predictions are written to the real-time hydro-meteorological database and are available for subsequent use within the OAS.

34.5.4 Flood Management Information System

The Flood Management Information System is linked to the CWRC's Flood Management Information Service System but also provides additional information such as the socio-economic impacts and information on the maximum reductions of peak water levels at over 40 locations within the river system whenever a particular basin is flooded (derived by hydraulic modelling), and a detention basins status management system. A socio-economic impact assessment procedure has been developed to determine the expected level of flood damage and a social impact index whenever detention basins are flooded. A literature review for the development of the socio-economic impacts assessment procedure may be downloaded from the project website www.yangtze.sagric.com. This information is also displayed in the OAS when comparing flood control options. The detention basin management system ("Basin Status" page) displays whether any basin is populated, people are evacuating, whether a basin has been breached or is flooding, and allows the user to amend any element. Such changes are also reflected in maps of the options analysis system (OAS).

34.5.5 Options Analysis System

34.5.5.1 General

The OAS has been developed as a tool to help decision makers assess the consequences of different flood management options in terms of the number of people to be evacuated, the cost of resulting flood damage, and the associated impact on flood levels and discharge at key locations. It is discussed in more detail in Betts et al. (2005).

More specifically, the OAS offers decision makers the following functions:

- A flood situation appraisal that shows the areas at greatest risk, suggests respective flood control options and allows high-level decision makers to amend those options;

- Allows users to rapidly investigate new reservoir release strategies or modify existing reservoir release strategies to mitigate excess flow volumes;
- Allows users to rapidly investigate the impact of operating detention basins if reservoir release strategies are insufficient;
- Quantifies the hydraulic and socio-economic impacts associated with flooding a single basin or series of basins at up to 40 locations;
- Provides a recommended strategy of basins to be flooded, the sequence of flooding, and a scheduling strategy according to existing or user-defined protocols;
- Provides the ability to reset strategies in accordance with current or forecast circumstances;
- Provides the ability to modify the recommended (default) strategy or develop alternate strategies;
- Once a strategy or strategies have been adopted, it allows detailed investigations using the hydraulic model;
- The ability to store metadata associated with decisions and review those decisions at a later time.

The same middle reach hydraulic model (MRHM) utilised within the FFS provides the basis for assessment of the impacts of potential mitigation strategies within the OAS.

34.5.5.2 Reservoir Operations

The impact of reservoir operations is incorporated within the OAS as simply a change in boundary condition to the MRHM. Within the middle reach, there are six major boundary condition inflows (including the Three Gorges Dam) that may be modified through reservoir operation.

In the first instance, the user may view boundary conditions (and implied reservoir operation) utilised in the forecasts produced by the FFS. If the user wishes to investigate the impact of altering release strategies, the appropriate boundary condition is altered, the MRHM re-run, and the results compared.

The usual release strategy to be investigated would consist of allowing impoundment of more flood volume to attenuate hydrographs passed to the downstream reaches. Note that it is also possible to investigate the consequences of pre-release strategies that aim to create additional storage within the reservoirs for anticipated future flooding from tributary catchments.

As the consequences of altering release strategies for reservoirs are far less damaging than the potentially catastrophic use of detention basins, the use of release strategies for flood mitigation purposes will always be investigated first.

34.5.5.3 Detention Basin Operations

There are over 35 detention basins within the middle reach of the Yangtze River and the decisions as to which basins to operate and the timing of these operations are

complex and vary from flood to flood. The CWRC has a set of operating "protocols" for these basins that have been developed through many years of experience with flooding in the middle reach of the Yangtze River. Figure 34.4 provides an indication of the location of detention basins within the middle reach of the Yangtze River.

Note that for most basins, the "operation" of a basin involves firstly the evacuation of the inhabitants followed by the destruction of a section of the dyke protecting the basin, allowing floodwaters to enter. Once the dyke has been opened, there are no means of controlling the flow until the levels within the basin equalise with those outside. Also note that it can take weeks to months to drain a basin once it has been flooded, imposing a severe cost in terms of displacement, lost production (industry and agriculture), and reconstruction. This additional storage available within the middle reach also changes the storage characteristics of the system, increasing the available storage until the basin is once again sealed off.

There are two basins that are not opened by simply breaching the dykes: these being the Jingjiang Basin and the Dujiatai Basin. Both of these basins have large gate structures which enable the inflows to the basins to be controlled. Inflow may also be minimised or stopped completely if conditions allow this.

The OAS provides the functionality to investigate the impact of operating detention basins. In the first instance, operation of detention basins according to the existing protocols is presented by the system. The user has the ability to alter the timing of basin opening, and in the case of the gated basins, altering the rules of operation to a certain degree. The user may also investigate the operation of other basins as alternatives to those suggested by the established protocols.

These operations are undertaken in a two-step process as follows:

1. A rapid, assessment of the effectiveness of various "scenarios" is undertaken at the website level, followed by
2. a detailed assessment of the most promising scenarios by detailed hydrodynamic modelling.

34.6 Typical System Operation

The results of a trial operation of the system are illustrated in Fig. 34.5.

These results show the predicted conditions at Shashi (see Fig. 34.4 for location) for a hypothetical flood emergency. The gauge observations at Shashi are an indicator of the potential exposure of the Jingjiang dyke to flood damage. If water levels at Shashi exceed the nominated critical level, the existing protocols indicate that the gate to the Jingjiang basin is operated such that excess flood volume is diverted into the basin in order to ensure the critical level at Shashi is not exceeded.

Four curves are shown in Fig. 34.5. The first curve corresponds to the water level predictions at Shashi produced using the FFS. These show that the critical level is predicted to be exceeded approximately $1\frac{1}{2}$ days after the time of forecast. It is then necessary to investigate potential mitigation options using the OAS.

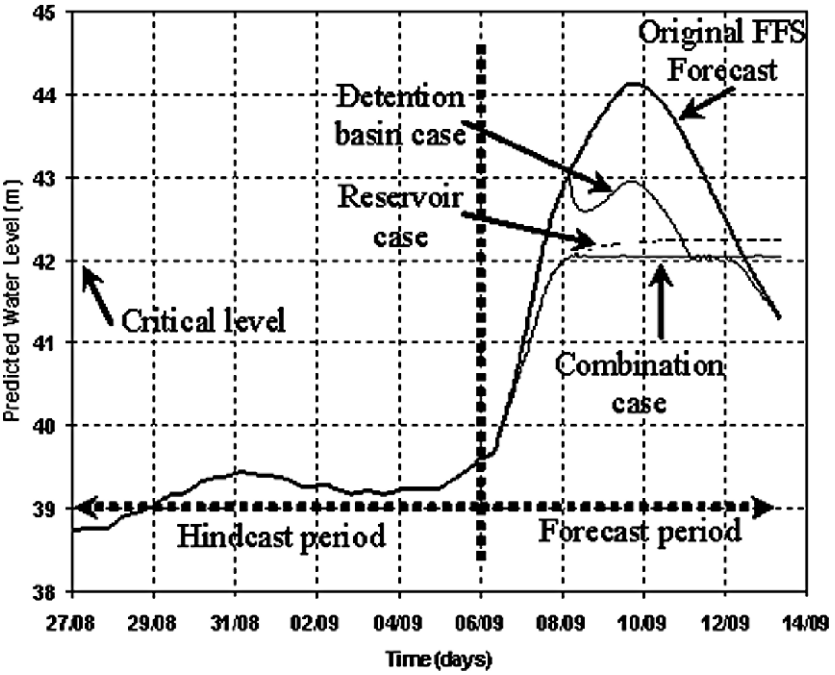


Fig. 34.5 Hydraulic results (water level predictions at Shashi) of the FFS and OAS for a hypothetical flood, trialing various reservoir and basin operation scenarios

The next three curves correspond to three different mitigation scenarios or cases that the user investigated as follows:

a. *Reservoir case:* Using the Three Gorges Dam to restrict outflow to the Yangtze at a rate such that levels do not exceed the critical level at Shashi
b. *Detention basin only case:* Using the Jingjiang detention basin to take excess flow volume
c. *Combination of reservoir and detention basin case:* Using the Three Gorges Dam as for (a) the Jingjiang basin and (b) in order to keep the predicted flood level at Shashi below critical level.

All the different mitigation cases considered have relative benefits and disbenefits or costs. Once these hydraulic results are exported to the OAS, the relative benefits and disbenefits may be quantified and presented in the OAS system so that decision makers can make an informed decision as to the best course of action.

More specifically, assessments may be made as to, amongst other things, the implications of impounding water within the Three Gorges Dam to the extent implied by (a) and (c). This may not be possible and would (in a real flood emergency) have to be the subject of discussions with the dam operator. As an alternative, the use of the Jingjiang detention basin without any flow mitigation from the Three Gorges Dam implies significantly higher damage as significantly more volume has to be

diverted to the Jingjiang detention basin. These benefits and costs are presented within the OAS.

Finally, once a decision is made, the adopted mitigation option details (reservoir release characteristics and detention basin operation details) are passed back to the FFS for inclusion in future forecasts.

34.7 Conclusions

In the hydro-informatics components of the project, new systems and software were developed using readily available technologies and applied in a manner that achieve the major objectives of the project and were done so in a manner that had never been attempted before.

Real-time, web-based systems have been established to firstly (through the FFS) improve the reliability, accuracy, and lead times of forecast flood discharges and flood levels along the upper and middle reaches of the Yangtze River, and secondly (through the OAS) provide a tool to assist decision makers to maximise the effectiveness of available flood mitigation measures.

It is now possible to produce 7-day flood forecasts with equal or better accuracy and reliability than the pre-project 3-day forecasts, and with further refinement, there is potential to make these forecasts even more accurate for most situations, which recur annually.

Additionally, the included decision support system (DSS) now provides a far more systematic and reliable means of managing floods and permits the CWRC to better inform all flood management and "at-risk" communities of pending risk and of the best management actions to be implemented to mitigate socially and economically damaging flood impacts.

The project has determined that real-time hydrologic and hydraulic numerical modelling is appropriate, accurate, and can be used to test semi-controlled flow diversions into detention basins.

These Hydroinformatic systems have been accepted by the CWRC and are being used on a daily basis through the flood season.

There is no doubt that the developed systems are replicable to other similar river basin systems as the overall system design is modular and can be tuned to meet the particular needs of each location or river system.

Originally intended as a flood forecasting and flood management system, the outputs can be used in the broader context of river basin management, orchestrating the use of reservoirs in concert, and applied to reconsiderations of land use for flooding, "higher" uses of land or the environment. It is also possible to adapt the existing DSS platform for water quality monitoring, modelling, and management.

Acknowledgements and Disclaimers This project has been funded by the governments of Australia and Peoples Republic of China under the "development cooperation program" with China. The Australian contribution is provided through the Australian Agency for International

Development (AusAID), and the Chinese contribution through the Changjiang (Yangtze) Water Resources Commission (CWRC) of the Ministry of Water Resources. The project is managed by SAGRIC International Pty Ltd in association with Coffey MPW Pty Ltd and Water Studies Pty Ltd, all members of the Coffey group of companies. The views expressed in this publication are those of the authors and not necessarily those of the Commonwealth of Australia. The Commonwealth of Australia accepts no responsibility for any loss, damage, or injury resulting from reliance on any of the information or views contained in this publication.

References

Betts HW, Sterling E, Clark SQ, Markar MS, Chen M, Huang W, (2005). Flood Management Decision Making in the Yangtze River Catchment, China. 8th International River Symposium, Brisbane, 5th–9th September 2005, Australia.

Clark SQ, Markar MS, Betts HW, Gooda M, Min Y, Zhang FW, Huang W (2005). Use of Numerical Modelling in Support of Yangtze River Flood Forecasting and Decision Making. 3rd International Symposium on Flood Defence, Nijmegen, 25th–27th May, 2005, The Netherlands.

Clark SQ, Markar MS, Min Y, Wu D (2004). Overview of Hydraulic Modelling of the Yangtze River for Flood Forecasting Purposes. 8th National Conference on Hydraulics in Water Engineering, Gold Coast, 13th–16th July 2004, Australia.

Government of Australia and Government of the People's Republic of China (2000) Yangtze River Flood Control and Management Project, Project Design Document.

Markar MS, Clark SQ, Gooda M, Min Y, Chen Y (2005). Improved Flood Forecasting for the Yangtze River in China. 8th International River Symposium, Brisbane, 5th–9th September 2005, Australia.

United Nations Economic Social Commission for Asia and the Pacific (UNESCAP) (1999). Water Hazards, Resources and Management for Disaster Prevention: A Review of the Asian Conditions, IDNDR 1991–1999, Bangkok, 23–26 February 1999. Source http://www.unescap.org/enrd/water_mineral/disaster/watdis5.htm#_Toc445864439 (downloaded 15th August 2005).

Chapter 35
Open Model Integration in Flood Forecasting

M. Werner

Abstract Operational flood forecasting systems have an established role in mitigating damage due to flooding through provision of timely flood warnings. Such forecasting systems are at the interface between developing modelling techniques, and practical use of these technologies, and therefore hold a clear challenge to hydroinformatics. Despite this challenge and significant attention from the hydroinformatics research community, adaptation of forecasting systems to new techniques is slow. This is partly due to most forecasting systems resulting from bespoke development of an operational environment around an existing model or set of models. Once established, adapting the system to changing needs is difficult both technically and organisationally. As an alternative to the model centred bespoke development, an open approach to integration of models is proposed. In this data centred approach, an operational flood forecasting shell is applied to provide both the operational setting of the forecasting system and an open framework through which models can be easily integrated. The decoupling of the models employed and the organisational setting facilitates adaptation of the forecasting system to both changing needs and integration of new modelling techniques and data sources.

Keywords Open model integration · flood forecasting · organisational challenges

35.1 Introduction

Flood forecasting and warning systems are employed operationally with what can be considered a simple objective. Through providing up to date, reliable information on forthcoming flood events to both relevant authorities and the public at risk, action can be taken to reduce damage due to the event. Operational flood forecasting systems are therefore recognised as an effective method of flood risk mitigation (Krzysztofowicz et al., 1992; Parker and Fordham, 1996; Haggett, 1998; Penning-Rowsell et al., 2000; De De Roo et al., 2003). The lead time with which reliable

M. Werner
WL | Delft Hydraulics, 2600 MH, Delft, The Netherlands

R.J. Abrahart et al. (eds.), *Practical Hydroinformatics.* Water Science and Technology Library 68, © Springer-Verlag Berlin Heidelberg 2008

information on forthcoming flood events can be provided plays an important role in the effectiveness of the forecasting and warning systems, as the longer the lead time the more effective any required response can be organised and executed.

To extend the lead time beyond that of the natural hydrological lag times, state-of-the-art systems incorporate some form of (model based) flood forecasting (Parker and Fordham, 1996). As a result of this clear need, significant research effort has addressed the development of modelling techniques for use in operational flood forecasting systems. Despite this research effort, illustrated in several papers included in this volume, most operational flood forecasting systems adopt a somewhat conventional modelling strategy, typically using a network of hydrological and perhaps hydraulic models (see e.g. Bürgi, 2002; Moore and Jones, 1998; Grijsen et al., 1992; Van Kalken et al., 2004). To understand the slow uptake of new research in modelling techniques, the operational setting of forecasting systems must be explored, as well as how systems that are used operationally have been developed. Most operational systems are tailor made, with their development centred around the particular modelling technique used. Whilst this may provide fit for purpose forecasting systems, the disadvantage is that once established, it is relatively inflexible to changing requirements or new research developments. The alternative is to adopt an open approach to integration of models in flood forecasting. This provides a clear advantage as the open concept allows flexibility to modelling techniques used, while maintaining continuity of the institutional process of forecasting.

In this chapter, the role of models within flood forecasting and warning is discussed and a flood forecasting platform that provides for such an open integration of models is briefly presented. The chapter details how the flood forecasting platform provides for the open approach, and discusses practical aspects of adopting the open concept, both from the perspective of the end user and the developer of the flood forecasting system.

35.2 Role of Models Within Flood Forecasting Systems

To understand the advantage of an open approach to model integration of models within flood forecasting systems, the role of these models is first studied. Haggett (1998) describes the flood forecasting and warning processes as a series of four stages (Fig. 35.1):

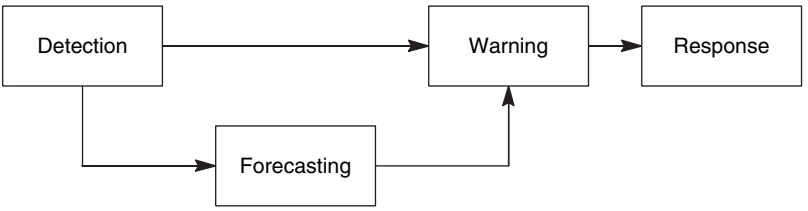


Fig. 35.1 Principal components of a flood forecasting and warning system

- Detection: This stage deals with real-time monitoring and data acquisition of processes that could generate a flood at the site(s) of interest. This includes hydrological and meteorological conditions, climate data, weather radar, etc.
- Forecasting: In the forecasting stage, predictions are made of the levels and flows, with particular attention given to the time of occurrence of threshold crossings and magnitudes of flood peaks. Typically this involves the use of hydrological models, driven using the real-time data gathered in the detection phase and forecasts of meteorological conditions.
- Warning: The warning stage is key to the success of operational flood warning. Using information from the detection and forecasting stages, the decision to warn appropriate authorities and/or properties at risk must be taken. Issuing of a warning may be initiated as a result of, e.g. a water level or a discharge having crossed a threshold in the detection stage, or as a result of the crossing having been predicted in the forecast stage.
- Response: Response to flood warnings issued is vital for achieving the aims of operational flood warning. An appropriate response must be taken following a warning to realise the potential of the warning system.

Although Fig. 35.1 shows an idealised schematic of the forecasting and warning process, it can be seen that the warning stage is primarily driven on the basis of monitoring, i.e. a warning is only issued if a monitored quantity has exceeded a threshold value. This is the structure of the most basic warning systems, where an explicit forecasting step is not included and flood warnings are issued on the basis of observations such as gauged rainfall and flows, combined with the judgement and experience of the forecasters (Cluckie, 2000). The figure also shows that if a forecasting stage is employed, then it is as a supporting step to the monitoring–warning process. It is in the forecasting stage that models are used to help extend the lead time with which warnings can be issued. Although the modelling technique used must be appropriately selected to reliably provide adequate information on which any subsequent warning is issued, this discussion illustrates that the role of models within flood forecasting and warning systems is a supportive one. This supportive rather than dominant role is reflected in the criteria used by Parker and Fordham (1996) in comparing the maturity of various operational flood warning systems in operation across Europe. Most of these criteria address the dissemination of the flood warning and the organisational embedding of the system. Indeed criteria were introduced to compare the importance of the flood warning step to the flood forecasting step, with systems where the forecasting step dominated deemed to be in an early stage of development, irrespective of how advanced the modelling technique used.

Despite this apparent supportive role of models, analysis of many research papers on flood forecasting systems shows significant attention is often given to the modelling techniques used and less to the process with which warnings are actually issued. If models are used in the forecasting stage, then these must be considered in the light of the complete process. Failure of flood warnings to reach the public, or a large number of warnings issued to the public that prove to be false due to

low accuracy of forecasts (Krzysztofowicz et al., 1992), will lead to performance deterioration of the flood warning service as a whole.

Whatever the appropriate modelling technique that reliably fulfils the information need of the flood warning stage, its application requires it to be integrated within an operational flood forecasting platform. This must then be capable of providing the required information clearly and concisely to the duty forecaster. A large number of forecasters may be involved, particularly as flood forecasting is often a 24/7 process. Additionally, these operational staff may not all be trained hydrologists. This means that when used operationally, the focus must be on the information provided on which possible issuing of a flood warning is assessed, and less on the merits of the particular model that provides this information. In other words, the model must be seen simply as a provider of information at one stage in the flood forecasting and warning sequence.

35.3 Open Model Integration vs a Model Centred Approach

The discussion in the previous paragraph shows that the role of models in operational flood forecasting is primarily supportive. Despite this, the traditional approach in establishing an operational system is to develop a hydrological model (using one or more modelling concepts), and subsequently to develop an operational shell around that model and its specific input/output data flow requirements. This evolution is quite logical as significant effort may be involved in establishing reliable models, and using these operationally is seen as a continuation of the development of the model. If this strategy is followed, the forecasting system is considered to be model centric. Additionally, the model can be considered to be closely linked to the flood forecasting organisation. This is because the shell, which forms the interface to the user(s), is specifically designed to cater for the requirements and processes of the particular model. Figure 35.2 schematically depicts this relationship.

At the outset of setting up the forecasting system, the model centred approach can be quite successful, and it often leads to rapid establishment of an operational system. It does, however, have a number of disadvantages. The most apparent of these is the versatility of the forecasting system as a part of the forecasting and

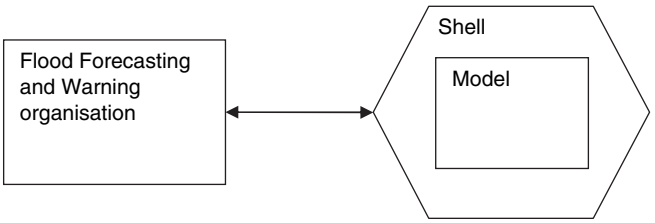


Fig. 35.2 Schematic relationship between a model centred shell system and the organisation that uses it

warning process when faced with changing requirements. When these changing requirements result in minor changes to the forecasting shell/model, the overall impact may be minor. However, required changes may be more drastic. Extension of the required lead time with which warnings are to be issued will, for example, influence the system as a whole. This means that the model may need review, perhaps even implicating the suitability of the model and subsequently requiring a different concept. A good illustration of this can be found in the case of the flood forecasting system for the Rhine used in the Netherlands (Sprokkereef, 2001). At a lead time of two days, the dominant process on which the forecast and subsequent warning is based is routing, and the original model and forecasting shell included principally hydrodynamic routing, with runoff generation processes addressed simplistically. With the extension of the lead time requirement to four days, the relevant processes to be modelled have shifted towards the runoff generation. This has resulted in a system of much higher complexity, and ultimately in the replacement of the entire forecasting shell and modelling system (though the original hydrodynamic model has been retained). Such replacement is not only a technical issue. It significantly impacts the forecasters using the system operationally, as well as the established forecasting procedures. Clearly the non-technical replacement process can be organisationally difficult and will often be met with resistance.

An alternative to developing an operational forecasting system centred around an existing model is to implement an "off-the-shelf" forecasting system. There are a number of suppliers "off-the-shelf" systems, but these typically dictate the use of a particular model, and should this not be the model originally invested in, then application of the "off the shelf" system will mean losing these perhaps significant material and knowledge investments. Additionally, organisations involved in forecasting may have close links with those involved in water resources and flood risk assessment, and the available models may serve multiple purposes and need to be managed as such. Developing additional models due to the constraints of a particular operational forecasting system is then unfavourable.

To alleviate the impact of changing requirements, which possibly results in changing the selection of suitable modelling techniques, an open approach to model integration is proposed. In this approach, the forecasting shell is entirely independent of the modelling techniques used, and it focuses on managing the forecasting processes and the required data flows. Where there is a requirement for information that is to be obtained from a hydrological/hydraulic process model, the shell system prepares the data for that model, runs the model and retrieves results from the model. Despite this close interaction with the model, it is important that the forecasting shell has no explicit knowledge about the model itself. Independence of the shell system to the models used is paramount, as this allows interchanging models when requirements change, without extensive reworking of the shell itself. This then guarantees that although the data provided to the forecaster may change as a consequence of the changing requirements, the process on how forecasts are made and disseminated stays the same. As a result, organisational changes are kept to those functional changes caused by the change in requirements, and limit required changes in managing the forecasting system as a part of the flood warning process.

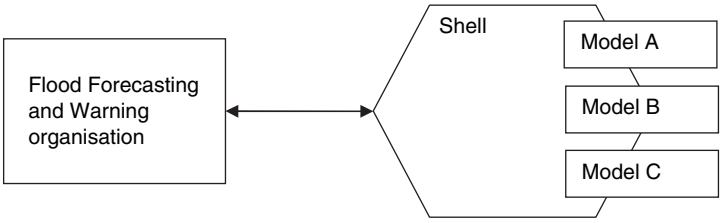


Fig. 35.3 Schematic relationship between an open model shell system and the organisation that uses it

An additional advantage of the independence of the operational shell is that there are no constraints on the models that are used. Should an existing model be available, with significant investment of knowledge and material, then this can be integrated, thus maintaining those investments, where appropriate existing models and newly developed models can be easily interchanged, or even used concurrently. This recognises the growing realisation that no single model may provide the most reliable results under all conditions. It also allows for a more rapid introduction of new techniques into actual operational systems. Figure 35.3 illustrates the independence of the models to the organisation, schematically indicating that the models are now integrated with the forecasting shell, but independent of the organisation.

35.4 An Approach to Open Model Integration: The DELFT-FEWS Forecasting Shell

An example of a forecasting shell that provides such an open concept to model integration is the DELFT-FEWS system (see Werner et al., 2004 for a full description). The architecture of the DELFT-FEWS system has been designed to provide an open framework that allows a flood forecasting system to be established to cater for the specific requirements of a forecasting authority. Through its modular structure it can, however, be easily adapted when requirements change. The modular approach has the advantage that many of the components used, such as the underlying models can be exchanged, without the need to change how the forecasting system is operated by its users. This allows for a much more rapid adaptation to advances in modelling techniques, without the added effort in organisational change. The system includes a wide range of intrinsic modules that deal with generic processing of data in the context of flood forecasting, including data validation, data manipulation, spatial and temporal interpolation, etc. A more extensive overview of the functionality provided is given in Werner et al. (2004).

The system provides a number of interfaces through which additional modules, including new models and forecasting techniques can be integrated. The most technologically advanced of these is that it allows users to extend the software through registering new Java classes that implement a defined software interface.

Although this gives the ultimate in openness and extensibility from a software engineering point of view, a more simplistic method of integration of new models and forecasting techniques is of greater practical relevance. The concept is one of a loosely coupled framework, where DELFT-FEWS makes the required inputs to the model available to the model, a model run is then initiated and the results are again made available by the model for importing into DELFT-FEWS. The format and protocols of communication between all models thus connected has been established in a so-called Published Interface. This interface defines the exchange of time series, states, parameter data, and metadata using XML-formatted files. Provided a model to be integrated in the forecasting shell implements the protocols and interfaces defined, it can be run operationally from the DELFT-FEWS shell, with the latter taking care of all operational data management and task scheduling activities. In practice this means that for each of the models run by the system, an adapter has been developed that picks up the XML-formatted data and metadata and transforms this to the native formats required by the model prior to the model run, and vice versa following completion of the run (see Fig. 35.4). Whilst the concept is technically simple, efforts to limit model-specific details to the model adapter and therefore keeping the data held in the forecasting shell as generic as possible have proven successful. Over 30 different models from a range of independent model suppliers have been integrated using the approach, thus proving its versatility. Examples of these are given in the next paragraph.

The advantage of a loosely coupled over a tightly coupled approach, which is often seen as the ideal in integrated environmental simulation solutions, is that the latter typically requires a lot of effort in integrating models, often with a requirement of the same programming language (Hoheisel, 2002). The loosely coupled approach taken here is much more flexible through the use of the simple XML interface. XML provides the advantage that for each of the data types to be exchanged (e.g. time series, metadata), an XML schema has been defined against which the XML data file can be validated. This means that if the XML-formatted data provided by the DELFT-FEWS system, as well as the XML-formatted data provided as an output of the model, complies with the schema definitions, then the model can

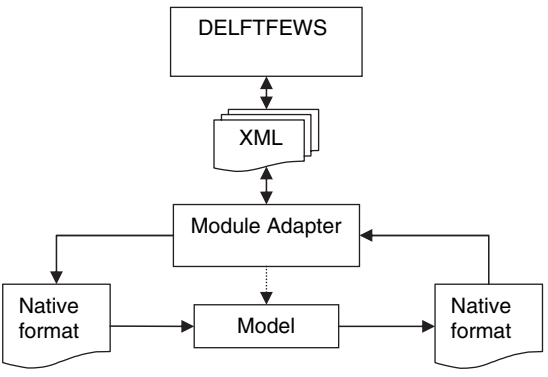


Fig. 35.4 Exchange of data between DELFT-FEWS and proprietary module formats

be integrated. As the XML schemas form a part of the openly accessible published interface, validation of the data exchange can be undertaken independently from both DELFT-FEWS and the actual model and its adapter. Again this has important organisational advantages, as the supplier of the model and of the operational forecasting shell (DELFT-FEWS) can remain independent. Typically the module adapter that implements the XML-published interface and maps this onto the proprietary native format (see Fig. 35.4) is developed by the module supplier, without necessarily needing to give open access to native module formats should this be undesired. The end user of the forecasting shell can even easily incorporate additional models, without any involvement of the shell supplier, provided the data exchanged can be shown to verify against the defined XML formats.

A third method of integration of models with the DELFT-FEWS shell system is through the OpenMI interface described in the paper by Fortune et al. in this volume (see also Gijsbers, 2004). This allows for all models that conform to this emerging standard in model integration to be easily incorporated in the forecasting shell, without the additional need of developing a model adapter. The variety of models supporting this interface is, however, relatively limited, and implementation of the interface requires significant software development capabilities.

35.5 Example of Models Used in Flood Forecasting Systems

Table 35.1 provides a list of selected operational forecasting systems currently in use that utilise DELFT-FEWS as the forecasting shell. Each of the model types listed is integrated with the shell using the loosely coupled approach in linking models described above. From the perspective of the forecasting shell system, there is no differentiation between these models other than the data passed and the executables called (note that in some cases the models are distributed 2D models, requiring exchange of 2D data time series). Figure 35.5 shows the user interface of the DELFT-FEWS forecasting shell used by the Institute for Inland Water Management and Waste Water Treatment (RIZA) to provide operational forecasts for the Rivers Rhine and Meuse at numerous fluvial sites in the Netherlands. All the operational systems listed in the table have a similar set up user interface and although the range of models varies, a comparison of how the models are employed, the processing of data within the forecasting shell, as well as how the systems are used operationally reveals that many of these are also very similar. This shows that the open model approach taken in the DELFT-FEWS system allows the process of forecasting to become largely independent of the actual model used.

The forecasting system used by the Environment Agency (NFFS) is an example of where the independence of the system with respect to the models employed allows the system to progress in a way that would not have been possible when taking the model centric approach. This system contains in fact eight independent regional

Table 35.1 Selected operational forecasting systems using DELFT-FEWS as the flood forecasting shell, and examples of the range of models used in each

Forecasting system	Lead time (h)	Models	Forecasting system
National Flood Forecasting System, Environment Agency (EA), England and Wales	2–24	ISIS (WS and Halcrow) Mike11 (DHI) PDM (CEH Wallingford) TCM (CEH Wallingford) NAM (DHI) MCRM (EA) KW (CEH Wallingford) DODO (EA) TRITON (PlanB, UK) PRTF (PlanB, UK) POL-2D (POL)	1D hydrodynamic 1D hydrodynamic Rainfall runoff Rainfall runoff Rainfall runoff Rainfall runoff Hydrological routing Hydrological routing Coastal lookup Transfer functions 2D hydrodynamic
FEWS- Rhine and Meuse, (RIZA), Netherlands	96–240	SOBEK (DH) HBV (SMHI) MLR (RIZA)	1D hydrodynamic Rainfall runoff Linear regression
FEWS Kamp & Donau, Austria	48	TU Modell (TU Vienna) Flux (Scietec Austria)	Dist. rainfall runoff 1D hydrodynamic
FEWS Pakistan, Federal Flood Commission, Pakistan	2–15 days	SOBEK (DH) Sacramento (DH)	1D hydrodynamic Rainfall runoff
EFAS, JRC, EU	10 days	LISFLOOD	Dist. rainfall runoff

WS, Wallingford Software; DH, Delft Hydraulics; DHI, Danish Hydraulics Institute; POL, Proudman Oceanographic Laboratories; SMHI, Swedish Meteorological and Hydrological Institute; EA, Environment Agency; JRC, Joint Research Centre

forecasting systems. The wide range of models used stems from the fact that the originally disparate forecasting systems have been migrated onto the DELFT-FEWS platform. Despite now having all these models operationally available, the objective of the Environment Agency is to in time reduce the amount of models applied so as to increase consistency on how forecasts are delivered. The selection of which models and model types are to be preserved can now be taken on the grounds of hydrological sensibility as well as management and maintenance considerations, rather than within the constraints of technical limitations.

35.6 Conclusions

This chapter looks at the role of models within flood forecasting and warning. The discussion points out that flood forecasting and the models employed to provide for these forecasts primarily have a supportive role in the warning process.

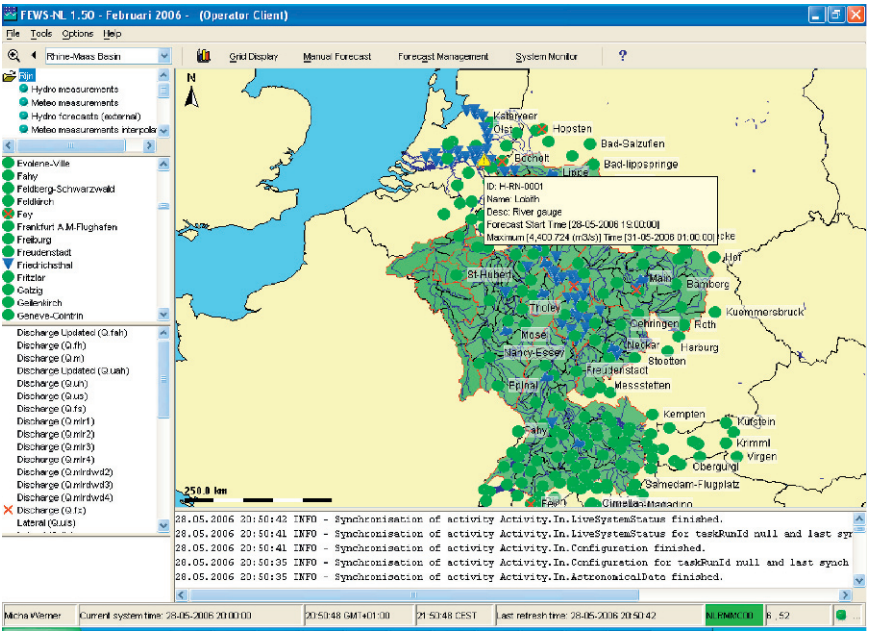


Fig. 35.5 Main user interface of DELFT-FEWS as used by the Institute for Inland Water Management and Waste Water Treatment (RIZA) in the Netherlands

As a consequence, the traditional model centric approach to establishing flood forecasting systems has the disadvantage that it is inflexible to changing needs. Changes such as a desired increase of forecasting lead time may result in the requirement to use a different model or set of models, which in the model centric approach would result in replacement of the entire forecasting system. Such replacement can have extensive organisational impact and will therefore be undertaken only reluctantly. Application of an open forecasting shell avoids this disadvantage of organisational change when the set of modelling tools used in forecasting is revised. The open approach can be achieved through a published file exchange format, with adapters developed to cater for the range of models to be integrated. By ensuring the models and the forecasting shell remain independent, both technically and organisationally, the open nature of the shell can be guaranteed. This open approach allows for the forecasting system to maintain its continuity as a supportive part of the flood warning process, while providing enough room for the models and tools used to be kept on par with the latest advances in modelling. These advances need not be limited to traditional hydrological and hydraulic modelling techniques, but can equally include new methods from hydroinformatics research, thus providing an easy springboard to bring such techniques into the operational arena.

References

Bürgi T (2002) Operational flood forecasting in mountainous areas - an interdisciplinary challenge. In: Spreafico M and Weingartner R (eds.) International Conference in Flood Estimation. CHR Report II-17. Bern, Switzerland, pp. 397–406

Cluckie I, (2000) Fluvial flood forecasting. Journal of the Chartered Institution of Water Environmental Management 14: 270–276

De Roo A de, Gouweleeuw B, Thielen J, Bartholmes J, Bongioannini-Cerlini P, Todini E, Bates P, Horritt M, Hunter N, Beven K, Pappenberger F, Heise E, Rivin G, Hills M, Hollingsworth A, Holst B, Kwadijk J, Reggiani P, Dijk M van, Sattler K, Sprokkereef E (2003) Development of a European Flood Forecasting System. International Journal of River Basin Management, 1: 49–59

Grijsen J, Snoeker X., Vermeulen C, El Amin Moh. Nur M, Mohamed Y (1992) An information system for flood early warning. In: Saul A (Ed.) Floods and Flood Management, pp. 263–289, Kluwer Academic Publishing, Dordrecht (The Netherlands) and Singapore

Gijsbers PJA (2004) The OpenMI Architecture – Details. In: Liong, Phoon & Babovic (Eds) 6th International Conference on Hydroinformatics, World Scientific Publishing Company, pp. 1819–1826

Haggett C. (1998). An integrated approach to Flood Forecasting and Warning in England and Wales. Journal of the Chartered Institution of Water and Environmental Management, 12: 425–432

Hoheisel A (2002) Model Coupling and Integration via XML in the M3 Simulation. In: Rizzoli AE and Jakeman AJ (Eds.) Integrated Assessment and Decision Support, Proceedings of the First Biennial Meeting of the International Environmental Modelling and Software Society, pp. 611–616, iEMSs

Krzysztofowicz R, Kelly KS, Long D (1992) Reliability of Flood Warning Systems. Journal of Water Resources Planning and Management, 120: 906–926

Moore, R, Jones D (1998) Linking Hydrological and Hydrodynamic Forecast Models and their Data. In: Casale R, Havnø K, Samuels P (Eds.) RIBAMOD River Basin Modeling, management and flood mitigation: Proceedings of the first workshop, pp. 37–56, European Community, EUR17456EN

Parker D, Fordham M (1996) Evaluation of Flood Forecasting, Warning and Response Systems in the European Union. Water Resources Management, 10: 279–302

Penning-Rowsell E, Tunstall S, Tapsell S, Parker D (2000) The Benefits of Flood Warnings: Real but Elusive, and Politically Significant. Journal of the Chartered Institution of Water and Environmental Management. 14: 7–14

Sprokkereef E, (2001) Extension of the Flood Forecasting Model FloRIJN. NCR Publication 12-2001, 2001, ISSN no. 1568234X (http://www.ncr-web.org)

Van Kalken T, Skotner C, Madsen H (2004) A new generation, GIS based, open flood forecasting system. In: Proceedings of the 8th National Conference on Hydraulics in Water Engineering, The institute of Engineers, Australia, ISBN 085825 850

Werner MGF, Dijk M van, Schellekens J (2004). DELFT-FEWS: An open shell flood forecasting system. In: Proceedings of the 6th International Conference on Hydroinformatics, Liong, Phoon and Babovic (Eds.), World Scientific Publishing Company, Singapore, pp. 1205–1212