Lecture Notes in Geoinformation and Cartography

LNG&C

Vasily Popovich
Christophe Claramunt
Manfred Schrenk
Kyrill Korolenko *Editors*

Information Fusion and Geographic Information Systems (IF&GIS 2013)

Environmental and Urban Challenges

НИЛ ООГИС
OOGIS RL

Springer

Lecture Notes in Geoinformation and Cartography

For further volumes:
http://www.springer.com/series/7418

Vasily Popovich · Christophe Claramunt
Manfred Schrenk · Kyrill Korolenko
Editors

Information Fusion and Geographic Information Systems (IF&GIS 2013)

Environmental and Urban Challenges

Springer

Editors
Vasily Popovich
St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS)
Saint Petersburg
Russia

Manfred Schrenk
Department of Urbanism, Transport, Environment and Information Society
Central European Institute of Technology
Schwechat
Austria

Christophe Claramunt
Naval Academy Research Institute
Brest
France

Kyrill Korolenko
Naval Sea Systems Command, Chief Scientist/Naval Undersea Warfare Center
Newport, Rhode Island
USA

ISSN 1863-2246 ISSN 1863-2351 (electronic)
ISBN 978-3-642-31832-0 ISBN 978-3-642-31833-7 (eBook)
DOI 10.1007/978-3-642-31833-7
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013946015

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

This volume collects the papers presented at the Sixth International Workshop "Information Fusion and Geographical Information Systems" (IF&GIS 2013), which was held from May 12 to 15, 2013, in Saint Petersburg, Russia. The workshop was organized by the Saint Petersburg Institute for Informatics and Automation, part of the Russian Academy of Sciences (SPIIRAS). IF&GIS is a biannual international workshop intended to promote the consolidation of efforts of academics and industrials from a wide range of disciplines, including computer science, geography, statistics, mathematics, hydrography, geomorphology, environmental, and urban fields. 2013 marks the 10-year milestone since the first international IF&GIS workshop was launched. This year, IF&GIS was specifically focused on environmental issues from global to local scales.

The objective of this workshop was to present the latest scientific and technological innovations, as well as geographic information system (GIS) applications to decision-making support systems intended for monitoring environmental and global warming problems and to monitoring, planning, and simulation aimed at sustaining the maritime presence in the Arctic. The scope of the Sixth IF&GIS workshop, and hence its proceedings, addresses several GIS and environmental research issues concerning modeling, analysis, information processing and visualization, as well as intelligent GIS applications to decision-making support systems. The papers selected by the International Program Committee reflect problems faced by multidisciplinary approaches to scientific domains to the extent that they address GIS fundamentals and their application to urban and environmental challenges.

While traditional topics at GIS conferences, such as ontologies and modeling on GIS and data security for GIS, are well represented and continue to be of importance, several new domains with an emphasis on urban and environmental challenges are now emerging. Maritime GIS is also at the cutting edge of novel applications intended to provide navigational safety so as to sustain a maritime presence in the Arctic, and it was addressed by several contributions presented at the workshop.

The submission process attracted 38 abstracts from 14 countries, and 26 abstracts were selected for full paper submission. After thorough reviewing, the Program Committee accepted 19 papers from 8 countries for publication, including two editorial papers. The accepted papers were assigned to the following five sessions: Ontologies and Modeling on GIS; Algorithms and Computational Issues for GIS; Data Security for GIS; Urban GIS; and Marine and Coastal GIS (for the Arctic Region). The IF&GIS 2013 program was further enriched by the contribution of our distinguished invited speaker: Assistant Professor Cosimo Stallo from Signal Processing and Satellite Navigation Institute of Electrical and Electronics Engineers (IEEE) Aerospace and Electronic Systems Society (AESS) in Italy, whose paper is also included in the volume.

The success of the workshop was assured by the combined efforts of sponsors, organizers, reviewers, and participants. We would like to gratefully acknowledge the contributions of the Program Committee members and thank all reviewers for their support and hard work. Our sincere gratitude also goes to all participants and all the authors of submitted papers. We are grateful to our sponsors, the Russian Academy of Sciences and the US Office of Naval Research Global (ONRGlobal), for their generous support. Finally, we also wish to extend our gratitude to Springer's team, managed by Dr. Christian Witschel and Agata Oelschlager, for their help and collaboration.

May 2013

Vasily Popovich
Christophe Claramunt
Manfred Schrenk
Kyrill Korolenko

Contents

Part I
Invited Paper

Using Space-Based Technology for Smart Resource Management during Disaster Early Warnings

Cosimo Stallo, Marina Ruggieri, Sabino Cacucci, and Donatella Dominici

Abstract To improve the decision process of first responders just before and just after a disaster, effective access to distributed unstructured disaster information and an infrastructure for precise and accurate disaster information access and retrieval could represent new and important tools. Moreover, they could help improve the performance of disaster prediction tools developed in any country in the world. This chapter describes a system based on integration of space and terrestrial technologies that aims to provide useful information to first responder organizations for smart management of disasters, both in the early warning phase and just after it.

Keywords: Global positioning system (GPS) · Swarm intelligence · Disaster early warning · Earth observation (EO) · Radio frequency identification (RFID) · Geographic information system (GIS)

1 Introduction

Humans face natural hazards at different scales in time and space. These hazards affect life and health: they have a dramatic impact on the sustainability of society, especially in societies that are vulnerable because of their geographic location,

C. Stallo (✉) · M. Ruggieri
Electronic Engineering Department, University of Rome Tor Vergata, viale del Politecnico 1 00133 Rome, Italy
e-mail: cosimo.stallo@uniroma2.it

S. Cacucci
Sistemi Software Integrati, Viale del Lavoro, 101 74123 Taranto, Italy

D. Dominici
University of L'Aquila, via Campo di Pile ex Optimes, L'Aquila, Italy

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_1,

poverty, or both. The first decade of the twenty-first century has been characterized by a lot of natural disasters, such as earthquakes (e.g., Sumatra-Andaman in 2004, Kashmir in 2005, Sichuan in 2008, L'Aquila in 2009) followed by landslides (China in 2008) or tsunamis (Indian Ocean in 2004); floods (e.g., West and Central Europe in 2002; China in 2007; Taiwan and Philippines in 2009); cyclones (e. g., Katrina in 2005; Nargis in 2008) and several others. On 12 January 2010, Haiti was struck by a violent earthquake; its epicenter was located near the capital city Port-au-Prince. According to the Haitian government, more than 111,000 people died in the earthquake, 194,000 were injured, and 609,000 became homeless (Buscema and Ruggieri 2011).

It is essential to study and develop very efficient tools in order to forecast these natural disasters, which can cause economic and social damage anywhere in the world. In particular, the United Nations, with the support of the European community, is now trying to develop initiatives for the prediction and early warning of such events (Buscema and Ruggieri 2011).

To provide an early warning of a disaster and improve the first response just after it, proper and prompt information should be provided to first responders. What kind of information should be provided to first responders just after the crisis to improve their response? How can this information be distributed in an efficient way?

This work describes an integrated system that aims to provide the following:

(a) Useful information to civil protection and first responder organizations for early warning of a disaster and just after it. This information includes the following:

- probability of strategic infrastructure collapse within a certain time and updated maps of infrastructure safety;
- information on people and goods (e.g., through sensors and radio frequency identification [RFID] technology) to help rescue teams to be more efficient and timely in their relief attempt;
- information on roads (e.g., through satellites) to help technical assistance teams get to the intervention place at the right time;
- updated disaster maps with an associated new risk level and probability of a new catastrophe strike within that region by a certain time;
- information about relief centers where people can go to get safety;
- information about how to contact and/or find relatives and how to leave messages.

(b) Improving the understanding of natural disaster processes, which will ultimately lead to mitigation of damage from this natural hazard;
(c) Early warning mechanisms for crisis reparedness;
(d) Emergency recovery and relief;
(e) Disseminating information to scientific and non-scientific users, who could use the data for their research to test new models or algorithms;

(f) Monitoring of potential dangerous situations on different scales, globally and locally, using satellite technology (Constellation of Small Satellites for Mediterranean Basin Observation [COSMO] SkyMed integrated with Global Navigation Satellite System [GNSS] solutions) instead of traditional solutions, which usually are more expensive and not affordable for small communities;
(g) Integration of different information developing a real-time, web-based geographic information system (GIS) platform (e.g., World Wild Web) that is accessible and well filtered, both for monitoring and risk maps.

Interoperability between Navigation Communication (NavCom) and earth observation (EO) systems can be used for collecting and distributing the data through a swarm network.

The system adopts a modular approach that will enable its application to different kinds of disasters (in particular, flooding and earthquakes) given a proper customization. The effectiveness of the project will be validated through trials involving different user groups who may be involved in the process of natural disaster response.

2 Novelties

The main novelties of the system are:

1. an intelligent and innovative distribution of data through a NavCom and EO integrated system to activate a swarm network for early warning;
2. a smart identification of the type of data to be gathered and distributed. These data could be used for disaster risk mapping, including tools and products enhancing mitigation and preparedness, thereby supporting adaptation strategies and prevention capacities. The useful data to be recorded in the database can be categorized as:

- Real-time data:
  - Local Data

    from earth: sensors, RFID technology (Developers' Alliance for Standards Harmonization of ISO 18000-7), smartphones;
    from the sky: global positioning system (GPS) and synthetic aperture radar (SAR) data for monitoring risk areas;

  - Global data:

    from earth: local and global geological data;
    from the sky: local and global meteorological data;

- Survey based:

– interviews of affected people conducted by anonymous forms distributed via a web-based portal;

– data collected from call centers on abnormal events, data collected from first responders and people after a disaster (interviews, short message service (SMS) from mobile phones);
– historical survey;
– geological survey.

3. new and advanced NavCom systems design for emergency rescue applications (involving space system and terrestrial enhanced wireless/mobile radio systems) in order to manage the first phases after a disaster. In fact, the use of extremely high frequency (EHF) [30–300 GHz] and, in particular, W band frequencies [75–110 GHz] are ideal for developing highly secure, high-bandwidth applications as Public Safety and Disaster Recovery (PSDR). There are many advantages coming from EHF use: assurance of reliable communications in a nuclear environment, minimal susceptibility to jamming, and the ability to achieve smaller, secure, high bit-rate beams by using small-size antennas (Stallo et al. 2010).

2.1 Data Collection Through Space-Based Technologies

Space-based technologies in the form of GNSS networks aided by augmentation systems, such as the European Geostationary Navigation Overlay System (EGNOS) and EO (by using COSMO-SkyMed) are the key contributors to measuring surface deformation.

Integration of these space geodesy monitoring techniques with other disciplines can be used to generate reliable hazard maps and define the so-called disaster preparation zone. Here, important local changes/deformations are used to designate a zone where future disasters have a high probability of occurring.

EO data can be useful not only to define the disaster preparation zone, for which other information such as disaster precursors and seismic data are needed, but also to detect and determinate possible earth surface deformations. These deformations involve changes in the properties of the crust (density, electrical resistivity, changes in groundwater levels, and other geochemical precursors) that can be measured using various techniques.

The most important thing to be noted is that all deformations are accompanied by the building up of tension; for this measurement, GPS techniques have recently made a strong contribution. The EO data can be derived from GNSS observations, high-resolution satellite images, SAR, and photogrammetry of proximity (Dominici et al. 2011).

First, the GNSS data can be obtained from permanent stations (PS) distributed throughout the considered area. Each station is equipped with both a receiver and a geodetic antenna; its main task is the continuous collection of all visible satellite signals and the recording of their code and phase over time, for 24 hours a day and 7 days a week. Each PS is also equipped with a local acquisition system guided by specific software, which permits the correct registration of all information coming from satellites (codes, phases, ephemerides). This information is broadcasted in real time by the transmission and receiver hardware (H/W) of the permanent station. Authorized users can access on the data recorded during the last hours for the entire area covered by the service (both by radio modems and via cellular phones).

All the positioning data are stored on customized software, creating time series and also detecting correlations in the ground deformation monitoring of the interested area. Finally, all of this information, integrated with high-resolution satellite images, can be used during both the pre- and post-emergency periods. Both multi-spectral and panchromatic images can be used.

For example, the World View platforms have been implemented with more bands than are traditionally used (red, blue, green, infrared). In addition the Worldview 2 satellite is also designed with sensors for the yellow band, "red edge" (705–745 nm), and "coastal" (400–450 nm). These characteristics allow end users to discriminate many more features than is possible with conventional sensors. In both the pre- and post-hazard period, this feature can permit an extremely detailed detection of territorial information, including damage distribution and documentation.

2.2 Data Collection Through Sensor Networks

Data collection and the exchange of information are easily performed through the development of swarm network architectures, wireless networks, and advanced sensors (Engelbrecht 2005). The territory security requires new paradigms in order to design decentralized and cooperative complex systems that adapt to an evolving scenario. If the information is widespread throughout the territory, the decision quality improves and the answer time is drastically reduced. The use of the "swarm intelligence" means using autonomous individuals who are able to cooperate with others and adapt to environmental variations. In the context of a disaster such as an earthquake, a swarm network can help during the prediction phase and initial rescue period. The swarm logic should be applied to a sensors/actuators network, which is able to self-organize in response to a seismic event.

The system-level procedure of a swarm sensor/robotic network should exhibit three functional properties, which come from natural swarms and thus are desirable properties of such systems:

- *Robustness*. The system should be able to operate even during disturbances from the environment or the malfunction of its individuals. The loss of an individual must be immediately compensated for by another one. Moreover, the coordination process is decentralized; therefore, the destruction of a particular part of the swarm is unlikely to stop the whole operation.

- *Flexibility*. The swarm agents should be able to coordinate their behaviors to perform tasks of different natures.
- *Scalability*. The swarm should be able to work under a wide range of group sizes and support a large number of elements without considerably impacting the performance.

A certain number of coordination mechanisms are considered here; once again, they take inspiration from nature. Two of the main mechanisms are self-organization and stigmergy:

1. *Self-organization*. In real systems made up of real nodes (i.e., robots, sensors), the concept of self-organization is crucial because it is the basic paradigm behind the autonomous behavior of the agents. Therefore, a swarm system does not need a central station driving the whole system because it is able to configure itself. In biology, self-organization is defined as the process in which the global patterns of a system emerge solely from numerous interactions among the lower-level components of the system itself. This is common in natural systems. Studies on self-organization in natural systems show that interplay of positive and negative feedback in local interactions among individuals is essential. The positive feedback cycle is balanced by a negative feedback mechanism, which typically arises from a reduction of physical resources. In addition to these mechanisms, self-organization also depends on the existence of randomness and multiple interactions within the system. The self-organization models of social insects and animals have already been used as inspiration sources. In addition swarm sensor networks can be considered as the engineering utilization of the self-organization paradigm. In fact, in real networks, self-organization is the capability of a system to change its organization in response to environmental changes without explicit external commands.
2. *Stigmergy*. In biology, stigmergy is a process typical of ants, in which pheromone trails are released to drive other ants to a food source. In real applications, when real agents, such as robots, are involved, this behavior is the property of the agent/node to modify the environment in which it is moving, in order to reach a specific goal.

Finally, the geo-referentation of a swarm agent system is significant for simplifying the organization of the swarm toward the final formation. In this process, the agents are aware in real-time of their position and can move according to it (Hadim and Mohamed 2006). A key role, in this sense, is played by the positioning accuracy and precision of GNSS, particularly when aided by an augmentation system, such as EGNOS.

In this regard, the URTV has developed two projects:

1. an application in which a GPS receiver is associated with each agent (as an external USB device) to perform certain missions during the phases of risk for the pre–post disaster that have a high level of risk for human operators; this

device is called the Software-to-Hardware Pen-Drive (SoHa-Pen) (Stallo et al. 2010; Barbera et al. 2010).

2. an application based on the use of the satellite compass for very accurate monitoring of strategic buildings or infrastructures: this project is called Satellite Compass.

2.2.1 Adaptive Information System for Prevention and First Response

This section describes the design of the Adaptive Information System for Prevention and First Response (AISPR). AISPR provides an infrastructure that is able to route the proper information to people in the proper places and times during a disaster. The principal aim is to support the operator in making the best decision.

AISPR is an autonomous network of heterogeneous multi-sensor nodes that are able to distribute information toward diversified users. These data are selective, provided only to enabled users who have the proper authorization at that particular moment. Such distribution manages different levels of information criticality, such as, reliability, availability, promptness, and security.

The pervasive nature of the AISPR allows one to achieve a complete, top-level view of the disaster phenomenon. The AISPR for a single infrastructure can be generalized to obtain an AISPR covering an larger area (Figs. 1 and 2).

The density and adaptive ability of the nodes, along with the serverless architecture, make the AISPR system intrinsically robust and able to function even with some malfunctioning nodes; in this sense, it exploits the swarm intelligence. Each single node is equipped with sensors that provide disaster and structural information, referring to the place where it is deployed. Such information can be

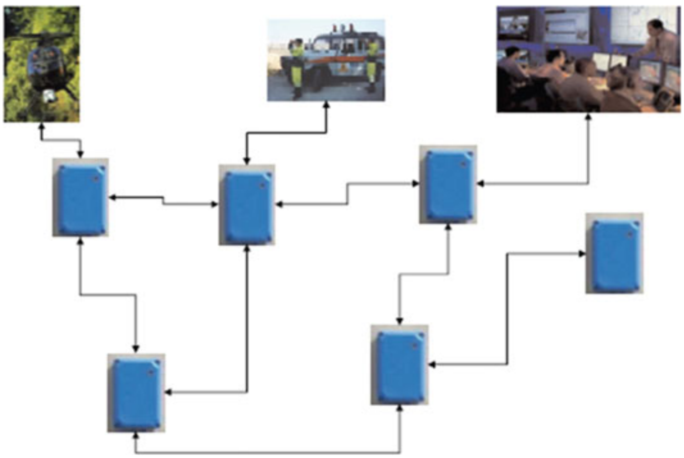


Fig. 1 AISPR network

Fig. 2 AISPR node in a building

used both by mobile and fixed operators. Information is time stamped and geo-referenced to give a detailed and updated picture of the territory.

Each node can also drive devices such as switches, which act autonomously to prevent further damage from occurring if pipes or power wires break, as well as lighting and emergency systems, which can automatically activate logically related key assets for the rescue operators.

AISPR can also be used as an ad-hoc communication network to exchange data among different operative nodes or as a database to collect data on events that occur in a building before a catastrophic event. In addition, there is an interface between the AISPR and a risk analyzer, to "make intelligent" the collected information. In designing this sensor network, the swarm model will also be considered. A swarm sensor network could represent an effective solution for a network that must adapt to new conditions after a disaster (for instance, some of the nodes are not usable anymore) and eventually perform elementary first actions (i.e., closing the gas distribution).

The main tools used for data processing are:

- artificial adaptive systems, such as artificial neural networks, which manage extremely complex information in order to predict disasters and to create maps of disaster risk factors;
- data collection and exchange of information, which are easily performed through the development of swarm network architectures of advanced sensors.

Finally, a fundamental coordination service must be provided by institutional organizations. AISPR can, in fact, make available a great amount of data, but all these data must be exploited in the proper way to guarantee effective coordination and synergy between civil protection, police, and fire brigades in the operative phases.

An online portal is the focal point for both dissemination and collection of information related to disasters. It also will serve as one of the bases for creating the "lessons learned" database (Fig. 3).

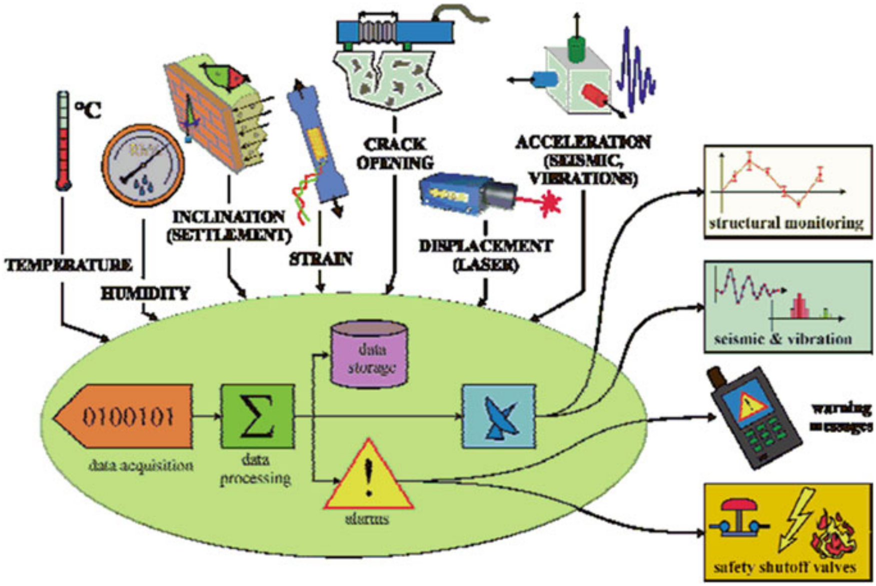


Fig. 3 AISPR sensors and actuators

3 System Architecture

The system is based on a three-tier architecture consisting of the database tier, the portal tier, and the service tier. Figure 4 shows the high-level architecture of the integrated system for early warning and disaster management. The building blocks of the proposed architecture include:

- the online portal;
- the system database supported by semantic analysis tools and an artificial adaptive system for data analysis;
- the sensor network;
- the GNSS and SAR networks;
- GIS/management information system (MIS) subsystems.

The data flows are shown in Fig. 4; the heterogeneous data coming from different sources, such as sensors, GPS, satellite measurements, folk information, interviews, etc., are collected through the online forum. This amount of data could be structured or not: Structured data are directly stored in the database, whereas unstructured data are processed using a semantic analysis tool and structured in order to be stored. Scientific measurements conducted in the test area will also be stored in the database.

All online forum subscribers will be able to access the database in order to find useful information. Moreover, the database will be supported by an artificial

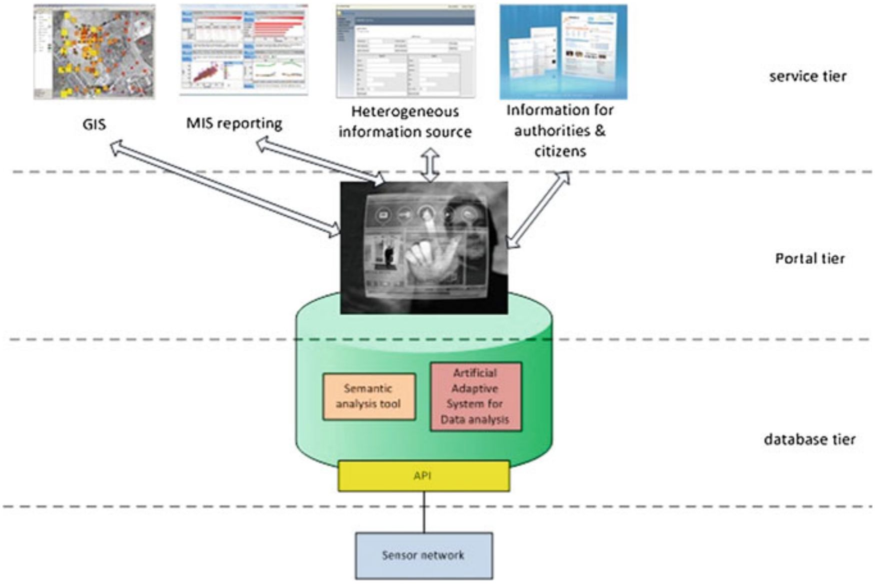


Fig. 4 System architecture

adaptive system for data analysis that could be managed by selected users in order to extract new information, such as the disaster risk of a specific area or more efficient procedures for disaster first response.

The data collected from the online forum and the sensor network will be used to test the artificial adaptive system, which will be fine-tuned based on the requirements of selected users.

As an example of the capabilities of the proposed architecture for improving first-response activities, let us analyze a typical earthquake disaster scenario. Right after the disaster, a small amount of new data, directly coming from the affected area, is available:

- "on-ground" sensor data (even if some sensors are not working after the earthquake);
- airborne or satellite sensor data: this information is not current but is usually a few hours old (typically updated data are available a few hours after the seismic event, when the satellite sensors next "observe" the disaster area);
- firsthand information, coming from people directly affected by the seismic event; these data are available as phone calls or text messages to civil protection, firemen, or other entities;
- information coming from the civil protection operators (or other entities) working in the disaster area.

The data are collected in the database, as gathered through:

- the online forum;
- a direct link with the sensor network;
- a direct link with selected users;

As an example, let us consider a building collapse event (FP7-Space 2012). This event can be registered by area sensors and by people living in the surrounding area; therefore, the following data should be available:

- a direct call with a message such as, “A large old building in front of my house, 5 Main Street, collapsed.” Using semantic analysis, the following information can be stored:
- Lat: 42°18′05’’N
- Lon: 12°25′17’’E
- Note 1: possible road obstruction
- Note 2: possible presence of victims trapped into building debris
- Vibration energy registered by a sensor near the collapsed building

Therefore, these data could be directly forwarded to civil protection working in the area (or forwarded after a first processing) and used by the operator to identify possible road obstructions (or collapse) or the level of danger in rescuing trapped people—all of which would improve first-response activities. At the same time, the data coming from the civil protection operators can be used to update the data stored into the database together with data coming from airborne/satellite sensors. All of this updated information is of utmost importance in the coordination of disaster response activity. Moreover, the creation of a “lesson learned” database will be useful, not only right after the crisis but also to enhance rebuilding and restoring activities.

4 Conclusions

This chapter described a database that can be used to gather and exchange information about disasters (earthquake or flooding, in particular). The previous experiences of first responders, people involved in disasters, and scientists studying disaster processes can be collected through a tool made available in an online portal). Our project will identify the types of data that should be gathered and how they should be processed to get useful information for managing the post-crisis phase. The database will allow for better preparedness and more effective responses to the future disasters, thus improving the capability to restore normal activity after a crisis situation.

References

Barbera S, Stallo C, Savarese G, Ruggieri M, Cacucci S, Fedi F (2010) A geo-referenced swarm agents enabling system: theory and demo application. IEEE First International Conference on Systems, Modelling and Simulation (ISMS 2010), 27–29 Jan 2010, Liverpool, England, pp. 100–104

Buscema M, Ruggieri M (2011) Advanced networks, algorithms and modeling for earthquake prediction. River Publishers, ISBN: 978-87-92329-57-8, 2011

Dominici D, Galeota D, Gregori A, Rosciano E, Elaiopoulos M (2011) Using new techniques of geomatic and structural control in the old city center of l'aquila after the April 6, 2009. Earthquake Proceedings of Joint International Symposium on Deformation Monitoring, Hong Kong 2–4 11 2011

Engelbrecht A (2005) Fundamentals of computational swarm intelligence. Wiley, Hoboken

FP7-Space2012-1 (2010) Collaborative Project Proposal, ENCELADUS

Hadim S, Mohamed M (2006) Middleware: middleware challenges and approaches for wireless sensor networks. IEEE Distrib Syst Online 7(3)

Stallo C, Cianca E, Mukherjee S, Rossi T, De Sanctis M, Ruggieri M (2010) IR-UWB for multi-gigabit communications beyond 60 GHz. Telecom Sys J 55:1

Stallo C, Barbera S, Savarese G, Ruggieri M, Cacucci S, Fedi F (2010) A geo referenced swarm agents enabling system for hazardous applications. UK Sim 12th International Conference on Computer Modelling and Simulation, Emmanuel College, Cambridge University, (UK), 24–26 Mar 2010, pp. 104–109

Part II
Editorial

Intelligent GIS Conceptualization

Vasily V. Popovich

Abstract The concept of an intelligent geographic information system (IGIS) is not new. However, the term represents a complex set of ideas and methods that cannot be combined into one method, theory, or concept. This chapter attempts to describe this new domain in geographic information system science and technology. Here, we present selected results of theoretical investigations and technological developments to share the vision of IGIS.

Keywords GIS · IGIS · Artificial intelligence · Inference machine · Ontology

1 Introduction

The history of the intelligent geographic information system (IGIS) is comparatively short. The first references to IGIS can be found in literature from the mid-1980s on the geographic information system (GIS), in which the intelligent subsystem was considered to be intended for solving a certain class of tasks, mainly related to data mining and cartography.

GIS applications for decision-making support systems (DMSS), which were considered to be a convenient interface and a key technology, created new requirements and challenges for conventional GIS. The requirement of intelligence is imperative for obvious reasons. One of the main challenges is the complexity of GIS, which is a key factor in computer science development, particularly in the process of software development. The entire evolution of the software development process from first-level languages to the concept of the object-oriented approach (OOA), which includes analysis, design, and programming, is based on

V. V. Popovich (✉)
St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS), 39, 14 Linia VO, St, Petersburg 199178, Russia
e-mail: popovich@oogis.ru

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_2,

dealing with complexity (Booch 1995) . OOA applications inspired the development of large and complex systems. However, some problems were exposed in the application of this methodology, such as in business logic development for a subject domain or a system of subject domains combined in one project. The business logic development process for a subject domain stretches into an infinite chain of software development cycles, according to Edward Yourdon (1989). These conditions thus threaten the implementation of heterogeneously distributed system development as a whole.

In addition, the complexity of decision making in a DMSS is a challenge, especially in real-time systems. This complexity is mainly caused by the human factor phenomenon (Brooks 1975). The intellectualization of the software development process and the decision-making process created a universal approach to solving specified groups of problems. All of these reasons were key factors in the development of GIS intellectualization.

IGIS is defined here as GIS that includes integrated tools and/or systems of artificial intelligence (AI) (Popovich 2003, Popovich et al. 2011). IGIS incorporates a number of basic components, such as:

- an inference machine (IM) and expert system (ES);
- a knowledge-based (KB) system (ontology);
- a visual environment for developing classes and objects of a subject domain;
- a visual environment for developing the models (script for acts) of objects functioning in GIS;
- a system for scenario implementation in real time or/and a user-defined arbitrary scale with a visual display as symbols or images on the background of electronic maps;
- recommendations for decision makers during the scenarios playing within the real system on research design, business games, situation analysis, and learning and training of personnel.

The proposed approach considers an ES and ontologies to be a system of AI. Thus, ES is used dually as a conventional ES, which assists the decision makers, and as a system to manage various processes working under control of some "scenario."

ES applications in GIS led to the emergence of many new scientific directions and technologies, such as:

- visual development of geospatial subject domain ontologies;
- development of ESs based on rules and objects;
- knowledge representation of the spatial environment based on ontologies.

Our experience indicates that IGIS applications increase the development speed and reliability of general and special software. IGIS also increases the efficiency of various information systems, including those for decision making on various levels, including strategic, operative, and tactical.

Many scientific methods and technologies related to intellectualization are united under the common name of AI. AI has been defined as "the scientific direction that poses and solves the problems of hardware or program modeling of human activities that traditionally are considered intellectual" or as the "property of intelligent systems to carry out functions, for instance, the creative ones, which traditionally are considered to be a human prerogative" (Sorokin et al. 2005). However, the notion of AI can have different meanings in different languages. For example, consider the remark made by Dr. T. A. Gavrilova, the chairman of St. Petersburg Chapter of the Russian Association for AI:

> for Russian-speaking readers …in English the word-combination AI does not bear that slightly fantastic anthropomorphous tinge that it has acquired in quite unsuccessful Russian translation. The word intelligence means "ability for reasoning", and not "intelligence" for which there exists an English analogue—"intellect."

This chapter discusses the systems and methods of AI rather than "intellect" itself.

Scenario is an important notion for IGIS. Here, the scenario is considered to be the selfsame algorithm with a capacity for parallel execution in some of its branches; the scenario has an ontological representation and is interpreted by the IM. The presence of such basic systems as the ES, IM, and scenario approach opens up high-grade and flexible possibilities for IGIS use in DMSS. In addition, AI could be used in the conventional AI sense as well as a real-time control system. AI applications in IGIS have the following advantages:

- the possibility of developing the business logic of the decision maker in the form of scenario ontology, thus assuring independence of software code and the possibility of visual development and correction for each scenario in the system's development and operation;
- a sharp decrease in cost for business analytics development;
- a decrease in DMSS possession cost and depreciation of DMSS possession;
- a decrease of the number of errors in business logic;
- the possibility of a fast, practically instantaneous, change of business logic, thus avoiding the traditional software upgrade cycle.

In addition to the intellectualization problem, IGIS solves some other problems, as demonstrated by the history of GIS formation.

1.1 Universal Model of Data

Historically, the majority of GIS researchers and producers are oriented toward the production of geographic chapter maps, as was dictated by the old cartographic tradition that faithfully served mankind for many centuries. Upon the appearance of computers, the situation changed dramatically; nevertheless, the newly emerged capacities were slowly introduced into practice. The old tradition preserves the whole cycle, from user interfaces to data models and exchange formats. The production (printing) tool of chapter maps penetrated into other subject areas, such as modeling systems, automated systems of decision-making support, and many others that are far from map production. The notion of layers is one of the main concepts of "chapter" cartography that found a wide application in electronic cartography. Layers became the basis for many data models and cartographic formats. As long as GIS was primarily used as the means for cartographic information visualization, the problem was insignificant; however, implementation of GIS in modeling systems and DMSS constituents obviously indicated the unpromising use of the layer notion and layer-based technologies.[1]

It is worth noting that the situation in cartographic production changed dramatically. According to the National Oceanic and Atmoshpheric Administration, the USA began using a new system of electronic cartography that incorporates the chapter maps printing system as one of the functions within the integrated cartographic system instead of as a separate technologic process.

This chapter focuses on the scientific and technical aspects of GIS development as a tool for decision-making support in real-time (or close to real-time) systems. In the considered systems, the electronic imitation of chapter maps does not work because the system should first and foremost assist the user in the decision-making process rather than simply displaying information in a user-friendly graphical form. More importantly, the system should not overload the decision maker with too much information, which may be redundant and hinder the user's perception instead of facilitating it.

To implement GIS in DMSS, detailed attention should be paid to GIS development basics, starting with data models (Popovich et al. 2004). Data models are the base for the subject area description—the main logical and mathematical scheme that is the foundation for all data, information, and knowledge storage. Eventually, the data model determines the complete GIS configuration and architecture.

The main assumption in data model development is that operation with the environmental information (subject area) cannot be accurately specified by a certain aggregate of analytical and/or other functions; that is, the unambiguous (isomorphic) transformation of one parameter into another based on the set cannot be determined. The considered approach matches the Immanuel Kant thesis "thing-in-itself" (Ding an sich), which has become increasingly more understood in a broad sense (Kant 1889). In general, certain phenomena can be observed such that any phenomenon stays partially incognizable. In this regard, information about the phenomenon should be replenished continuously from outside sources, such as a measuring apparatus and/or data source, so that analytical and statistical concepts concerning the entity and phenomenon are continuously refined and edited. For example, in digital mapping, real physical parameters such as temperature, salinity, pressure, and speed measured by sensors should be considered rather than abstract parameters such as current, isobaths, cyclones, and ice, which are "assumed" by man. Some conventional and/or other abstractions could be

[1] Useful model for the presentation of information in the GIS.

exclusively formed based on primary data. This is important because the isomorphism condition is rarely used in data transforms. Furthermore, the situation could occur when other processing methods are applied to already processed data, thus causing an impetuous increase of data inaccuracy. The formal substantiation of these ideas can be found in higher algebra model theory (Tarski 2004).

Avoiding the layer notion and receiving (when needed) direct access to measurements requires an introduction to the abstract data model or environment universal model (EUM). Such a model is based on an abstract space extraction in the form of a multidimensional vector, allowing for adjustment to a definite subject area. This model accounts for environmental changes by conventional electronic navigation maps, topographic maps or specifically generated subject area maps, diagrams, reports, objects, and three-dimensional (3D) dynamic images based on modern computer paradigms or other information technologies.

1.2 General Architecture of IGIS

Depending on the theoretic research and practical implementation, the major requirements for IGIS general architecture could be defined as follows:

- OOA should be used in IGIS as a primary approach instead of a cartographic (topographic, topological, etc.) approach for describing maps and objects on maps. It should include complex object support, class property inheritance, encapsulation, polymorphism, overlapping, reboot of functions (object models), etc. Here, object orientation does not mean that IGIS is written using object-oriented tools; the idea is that the same mechanisms should be rendered to the system user in an interactive mode to allow for organization of the space–time data.
- Different IGIS users should be given an opportunity to simultaneously attribute any objects on the electronic map to various classes and to construct multilevel "subjects" (classification systems with different bases), possibly for each user's application or definite research.
- Maximal openness of IGIS architecture assumes that, by default, IGIS represents the mechanisms of data exchange with external systems based on open data formats and the possibility of IGIS function extension. The presence of an internal restricted language is insufficient; it is required that IGIS provides access to its functions and internal data from external programs, written by other developers based on the developers' application programming interface. Distributed and rapidly developing systems should be able to be integrated with other developments existing in different subject areas.
- It should be possibile to establish all necessary topological relationships between the objects on the digital map. The importance of system support for the whole spectrum of topological relationships between objects is so high that, if the above mechanisms are omitted, IGIS turns into a map editor. However,

IGIS is primarily intended for analysis of information distribution in space and for modeling situations and processes as needed for prognosis. In particular, without the inter-objects and inter-layers topological relations the problems of modeling (analysis and prognosis) the development for complex network systems.

- IGIS should be easily managed in a distributed environment (including the Internet).
- The presence of a well-developed system ensuring rights of access to cartographical and semantic data is mandatory. So far, in most modern IGIS, a well-developed system that ensures rights of access to cartographical information does not exist.
- Embedded IM and ES should be included.
- An embedded database (DB) system (ontology system) should be included.

Figure 1 schematically depicts the general architecture of IGIS (Popovich et al. 2005).

The central part of IGIS is a KB that encompasses an aggregate of ontologies describing the subject area entities and their relationships. The other part of the KB is an aggregate of objects representing real entities of the subject area. An ES that includes IM and a set of logical rules is an important part of IGIS. Other architecture components are quite conventional for GIS. The structures of some constituents will be further considered in detail.

The object server is intended to support the uniform object model of the whole system. The server's main functions are to give access to the subject area objects and perform operations on them, eventually building objects based on the subject area meta-model stored in ontology, destructing objects, and modifying the properties of objects.

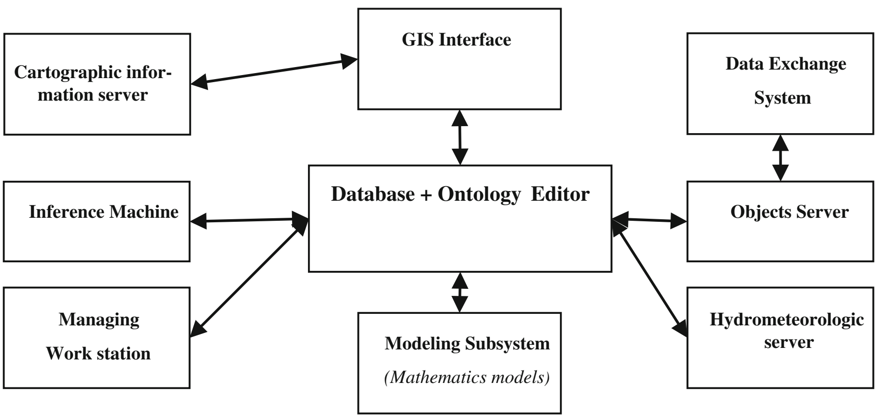


Fig. 1 General architecture of IGIS

The GIS interface is composed of minimal set of constituents, executing interactions between the system and the user and integration of software constituents in order to support an appropriate system functionality. The GIS interface renders a set of interfaces for interaction with the other software constituents at a technological level. The GIS interface represents the current situation on the electronic map and arranges for interaction between the user and application. To adjust the whole system and the GIS interface to a certain subject area, plug-ins are used to change the appearance of the interface accordingly. Localization of the user's location, workstation, keen client, worksheet, or communicator is performed via the GIS interface.

The complex architecture of IGIS cannot be realized within one programming system or software development environment. Organization of different levels of servers is required to support interactions between all constituents.

2 Service-Oriented Architecture

As previously discussed, IGIS incorporates a number of interacting concepts, ideas, approaches, and technologies. Theoretically, an abstract IGIS theory can be the subject of separate research, resulting in the construction of a system with rigorous axiomatic approaches. Moreover, practical orientation is one of the important factors in GIS science. In this regard, special attention should be paid to the problem of different approaches' interaction on a technological level and to developing ready-made applications on a theoretical level. In light of the above, let us consider the idea of interaction as proposed by the concept of service-oriented architecture (SOA), which is currently an advanced concept in information systems' interaction.

2.1 General SOA of IGIS

To realize the SOA, it is necessary to logically construct the architecture and then arrange for its realization through a certain set of technologies, products, and platforms for different purposes. SOA consists of three major logic components:

- *Users*—someone using the service proposed by a supplier;
- *Suppliers*—someone offering a definitive service or function;
- *SOA infrastructure*—a basis for arranging a normal consumer–supplier interaction.

IGIS users might be heavy desktop users, or they may use interfaces such as web browsers, mobile devices, portlets, or other services generalizing information arriving from lower-level servers. The GIS interface is the main interface for the heavy IGIS user. The GIS interface is a software component for the visual representation of space data of various geographic digital formats and the subject area's objects stored in databases. It supports user–system interactions and provides the interface necessary for solving applied problems. The GIS interface allows the user to map the cartographic information in two-dimensional and 3D formats. Plug-ins are provided to get IGIS attuned to a definite subject area. These plug-ins accordingly adjust the specifications of the functional tasks being solved by the system.

SOA infrastructure incorporates the services for supplier–consumer interactions and the service backup. The service backup contains different components supporting the SOA mechanisms' normal functioning. It is intended for developing the SOA infrastructure for further deployment of technical services, including five standard functions: directory, security, management, orchestration, and semantics. A detailed description of these functions is beyond the scope of this chapter.

The service suppliers define the service query format and publish it for detection and multiple uses. The following entities may be included by the service suppliers in IGIS:

- cartographic information server;
- hydrometeorological information server;
- object server;
- IM;
- simulation and mathematical model server;
- administration (management) server;
- interaction with external systems server.

The services in IGIS can be divided into three main groups:

1. *Application services*. Application services provide data and control parameters for organizing common work in one unified computer space. In our case, we are operating with many heterogeneous applications in different layers: one workstation, local network, or global network.
2. *DBs, KBs, and ontology services*. DBs and KBs and ontology subsystems require many kinds of services. A DB system is heterogeneous and as a rule is distributed inside a local or global network. Therefore, for every part of a DB, a particular kind of service should be selected. The DB environment forms unified data, information, and knowledge spaces. This is a key issue in developing large-scale practical applications and systems in IGIS.
3. *End user's services*. The basis for the end user's services is a web interface, except for real-time applications.

Let us consider some of these groups in detail.

2.2 Application Services

Application services realized in IGIS can be conditionally divided into subgroups based on these service suppliers.

Cartographic information servers provide the following applications:

- accessing spatial data in different formats, including Shape, S57, SXF, Open-StreetLayer, and VPF;
- geodesic functions for a given projection and Earth model;
- visualizing cartographic data;
- importing/exporting spatial data to/from different formats, etc.

Administration (management) servers provide the following applications:

- resource distribution;
- managing access for the user and other services;
- managing the IGIS tuning and operating modes;
- IGIS operation journaling, etc.

Hydrometeorological information servers provide the following applications:

- selection of and attuning the channel to receive hydrometeorological information from a source via transmission control protocol, file transfer protocol, and e-mail.
- filtering the flow of recipient information in accordance with the required types of hydrometeorological weather reports;
- sentence analysis of the recipient's hydrometeorological information reports according to the international (World Meteorological Organization) and regional code forms;
- decoding and storing the values of given meteorological parameters;
- controlling the server's given operation processes.

Modeling and mathematical models' servers provide the following applications:

- universal time;
- mathematical problems (e.g., search theory, radio location, hydroacoustic);
- multilevel data processing (Popovich and Voronin 2005);
- simulation;
- 3D representation of results;
- object recognition and classification as well as tactical situations.

Interaction with external systems servers provide the following applications:

- receiving information from external mobile objects and systems (e.g., transportation monitoring systems, permanently functioning network of oceanographic stations based on drifting gauge-buoys [ARGO project], and others);

- receiving locations of sea and river vessels in the World Ocean based on automatic identification system data;
- receiving locations of aircrafts based on the data from Automatic Dependent Surveillance-Broadcast system's transponders;
- receiving locations of spacecraft and calculation of their coverage zones based on open-source data and a number of other services, as determined by a definite subject area.

The object server sustains the unified object model of the whole system. The main server's services provide access to the subject area objects and operations with the objects, generate objects based on the subject area metamodel, store the ontology, destruct objects, and change the objects' properties.

2.3 Database, Knowledge Base, and Ontology Services

The functions of providing data access are assigned to DBs, KBs, and ontology. The KB is an important component of any intelligent system. KBs provide the data necessary for the functioning of the IGIS ES component. The ES is a main reason why GIS can be called "intelligent." It is intended to seek techniques to solve problems in a certain subject area based on DB records and on a situation described by users. The ES can serve to solve two problems. The first problem is conventional for ESs: making recommendations in situations that present difficulties in decision making. The second problem consists of managing (controlling) the complex modeling modes. The ES core—IM, using, for example, the Rete (Forgy 1982) algorithm—can describe complex concurrent processes by sets of simple rules while providing high levels of managing (control) and modeling efficiencies. The ES uses ontology services in addition to DB services. Ontology is intended for overall and detailed formalization of the IGIS subject area by using certain conceptual schema. Interaction between the KB and ontology (Gruninger, 1995) objects is a "frame" intended for representation of concepts and their links (relationships) within the application's subject area. The universal IGIS should load the various scenarios in the KB and, thus, be attuned to different subject areas. DB services and ontologies are consumers of BD services that furnish information for their functioning.

As a rule, information in IGIS is divided into the following groups:

1. Cartographic data
2. Object data
3. Data received from external sources
4. ES scenarios

Because each of the above groups is intended to realize different functionalities, the requirements each group is expected to meet are rather diverse. Cartographic data are aimed at the user-oriented visual representation of information in the form of geographic maps (Figs. 2, 3). The map is generated based on data sets. Several sets may be used while solving certain tasks, such as sets containing sea and ground maps, vector graphs of satellite shots and drawn roads, and topographic maps with superimposed height nets. The sets are divisible by types, including vector, raster, and regular value matrices; each set has its own format.

Object data contain knowledge about the dynamic information model of the subject area. The dynamic information model in the considered case is organized in accordance with a certain rules system totality of situation elements; at each time instance, it contains data matching factual parameters of the objects and medium of their functioning (Popovich et al. 2005).

2.4 End User Services

IGIS provides the end user with a great number of services, such as:

1. Access to cartographic information
2. Access to hydrometeorological information
3. Mathematical problem modeling and solving (in theory of search, hydro-acoustics, radio location, etc.)

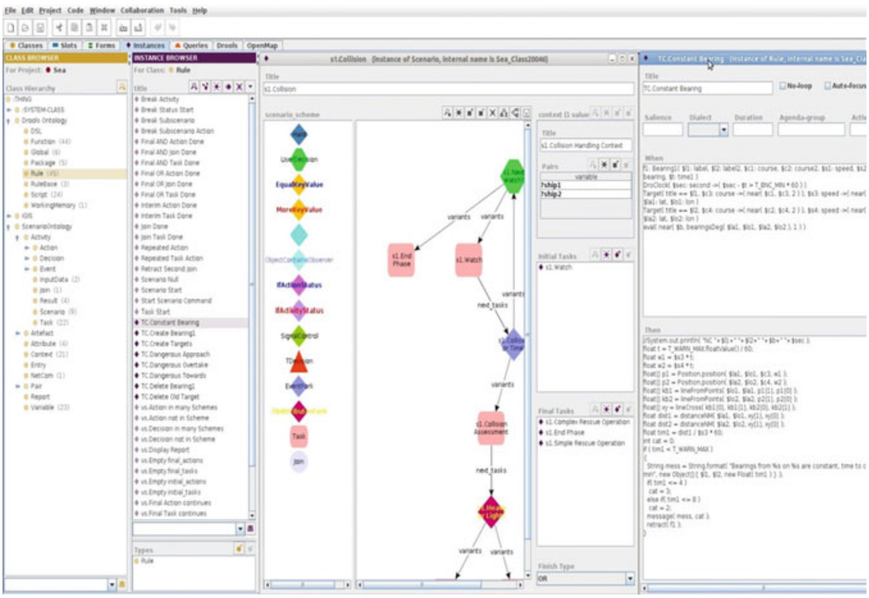


Fig. 2 Visual medium for scenario development

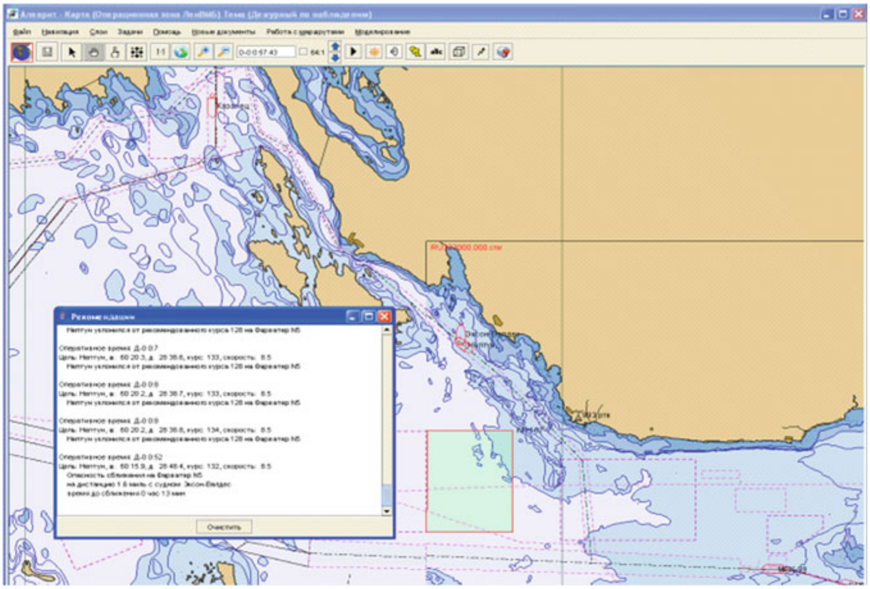


Fig. 3 An example of expert system operation in the form of interactive prompting

The dynamic information model of the subject area underlies all users' services. As mentioned, the model provides access to all objects of the subject area and allows one to solve all IGIS applied problems. Access to the model elements is performed through an aggregate of users' services of the object server. The object server is specified by the following service groups:

1. Subject area universal description;
2. Inheritance mechanism, also including multiple inheritance of objects;
3. Separation of an object's stationary and transient data;
4. Universal mechanism of relationships;
5. Access to the history of the objects' property states;
6. Information filtering for different user groups.

The subject area universal description describes objects within the uniform model of data representation from unified standpoints. The subject area universal description by default assumes access to the objects' classes with an indication of their peculiar properties along with their values. It also lists possible object states based on business process requirements, expressed through the values of an object's congruent properties and through the list of relationships, binding the objects while realizing the business function.

A variety of problems/tasks from different subject areas that are being solved by IGIS may involve the same information model's objects; this fact requires different object behaviors. Therefore, while solving tasks of this kind, the object can be classified by different attributes. The mechanism of multiple inheritance lets such objects be described in the information model.

Different behaviors of the same objects in different tasks/problems leads to one more issue. In the simultaneous modeling of different business processes, the properties of the same object may take on different values. The mechanisms separating constant and transient data avoid the conflict of simultaneous altering of the same object properties. The services accessing ontology objects and the topic objects are responsible for this task solving.

The universal mechanism of relationships describes the complex loaded links between different objects of the information model that arise while realizing business processes.

To analyze the information model's dynamics and characteristics of its state change in-depth to separate object properties and (when needed) to restore the state to the given time instance, it would be necessary to use the mechanism storing the history of object property states.

GIS information resources may be available to a great number of users of different levels and access rights. In order to solve certain problems, the different user groups will need different information about the same processes and participating objects. Services filtering information for different groups of users are intended to confine access to the necessary information.

The overall aggregate of specific problems being solved by IGIS is constructed based on the above services of the object server.

3 Ontology

Ontology is not a new phenomenon in computer science, but it is often asked why ontology is needed for a particular project. Here, we present some arguments in favor of using ontology in IGIS:

- creating a uniform system of subject area notions and their relationships;
- forming a uniform space at data, information, and knowledge levels;
- providing a uniform space for the users, applications, and data (knowledge) bases;
- assuring the possibility of knowledge reuse and multiple use within the subject area;
- explicit formalizing of suppositions and assumptions within the subject area;
- separation of knowledge and live data within the subject area;
- knowledge analysis within the subject area.

Development of uniform information fields within the IGIS constituents, as well as within other applications, is the priority task of ontology approach implementation. Existence of the universal model of environment and uniform ontology systems significantly facilitates the use of different cartographic data sources, such as shape formats, C57, VPF, CXF, and others. The service system is also ordered and does not cause any conflicts at interactions between various applications or agents. A well-thought-out ontology system also simplifies the problem of data transfer between various models, because it does not need special mechanisms or format development.

3.1 Knowledge Representation by Ontology

Determination of subject area knowledge is an important property of the ontology approach. Here, the main point is that the knowledge system is separated from realization (i.e., from the software code), which is impossible in conventional OOA. Thus, quite a flexible system is possible, which allows for altering business logic without interfering with the substantial regulations in the subject area.

It should be noted that data analysis in the subject area is possible when the terms' declarative specifications exist. The terms' formal analysis is extremely valuable for the existing ontologies' reuse, as well as their extension.[2]

As a rule in computer science, the subject area ontology is not a target in itself. Ontology development is somewhat similar to the determination of data sets and their structure when intended for use by other programs. The idea itself is somewhat analogous to OOA and earlier approaches, such as the concept of data abstract types. When realizing the task-solving methods and developing software components, the use of KBs as ontology data is rather effective. This dramatically simplifies the development of software code and architecture of the subject area classes.

Notions such as classes, as realized in the ontology software system, become information objects and are transferred from the category of software code to the category of data that could be manipulated by IGIS, as well as by the interacting environment.

The proposed approach has many advantages:

- subject area object classes and their relationships are described as data and not as software code, which significantly increases the software system's flexibility;
- alteration of the class characteristics does not imply a necessity to change the system software code; consequently, maintenance of the class libraries is simplified;
- class libraries can be stored in DBs, which means that the system's DB structure becomes simpler and does not require any alteration to class descriptions.

A number of languages have been developed for meta-class descriptions. The best-known languages are briefly described here.

---

[2] OWL Web Ontology Language Guide URL: http://www.w3.org/TR/2004/REC-owl-guide-20040210/#import.

Consortium WWW (W3C) developed resource description framework (RDF), a language for knowledge coding. This technology was used for the first time in forming web-page content. Since then, the use of RDF has been extended. In general, it can be used to perform global navigation when searching for data or knowledge needed for software components, executing an information search, or realizing definite functions.

Based on the four-level meta data model and unified simulation language, the unified modelling language developed a special language, Web Ontology Language (OWL), which was intended for metadata exchange between various modeling tools via the Internet. This language allows one to describe any ontology of the subject area.

3.2 Open Technology for Ontology Development

Currently, the following systems for ontology management are in wide use: Protégé, SUMO (Suggested Upper Merged Ontology), KAON2 (The Karlsruhe Ontology and Semantic Web tool suite), LarKC, and CYC. The functional capacities of these systems are different, and it is somewhat difficult to single out uniform criteria for comparison. Each of these systems has some advantages and disadvantages, which could be applicable in solving a definite class of problems.

The Protégé system can be considered as an example. It is a free distributed ontology editor and a tool intended for KB development.[3] The system was developed by a research group in medical informatics at Stanford University. The system has two versions: a system based on graphs of descriptive logic (DL; Protégé-OWL) and a system of frame-based representation (Protégé-Frames).

Protégé, based on the frame model of knowledge representation OKBC (Open KB Connectivity), is intended for ontology development, visualization, and manipulation using various languages for ontology descriptions. Classes, instances, slots (describing the properties of instances and classes), and facets (specifying additional information about slots) are the main constituents of such a system.

Protégé has an open architecture that enables the extension of its functionality based on modules called plug-ins. It has an extension adapted for editing models stored in different formats (standard, textual, in DB JDBC, UML, languages XML, XOL, simple HTML ontology extensions [SHOE], RDF and RDFS, DAML + OIL, OWL). So far, more than 60 modules of the kind are developed for Protégé.

Protégé can be used as a self-contained subsystem. Conjointly with GIS, the system can play various roles, such as a subsystem in the subject area of business analytics, ES, and/or IM. In some cases, Protégé can be used as an IGIS kernel.

[3] Protege. www.protege.stanford.edu.

3.3 Ontology for Subject Domains

Even relatively simple IGIS applications can hardly be squeezed into the frame of a single subject area. Therefore, when dealing with a real application, a system of ontologies should be implemented rather than just one ontology. To understand how to determine the subject area list or the domain list, consider the principles of OOA (Booch 1995). When developing large distributed systems, division into several ontologies is an extremely important issue and a series of principles should be followed:

- all class sets of different ontologies must be of a disjointed nature;
- ontology development, editing, and modification should be performed by the subject area professional rather than the programmer;
- ontology interface systems should be altered only for exceptional cases (detection of conflicts and/or errors, system upgrades, etc.).

3.4 OWL and GML

The web ontology language OWL can describe the classes and their relationships that are proper for Web documents and applications. OWL ontologies may contain descriptions of classes, properties, and their instances. OWL allows for the correct determination of the terms declared in one application (system) in another system, independently of the systems' technical and work scenario specifications. Based on DL, OWL provides tools intended for the logical description of notion semantics (sense); therefore, the information can be coordinated for use by people as well as by applications in different information systems (Websites, DBs, ESs, DMSS, etc.). Additionally, the OWL language supports the complete computer processing of Internet content better than XML, RDF, and RDF Schema (RDF-S); therefore, it provides an additional terminology dictionary along with formal semantics.

SHOE was the earliest language for Web ontology representation. DAML + OIL (DAML + OIL, 2001) was an OWL predecessor; it used the integrated projects of DALM (DARPA Agent Markup Language) and OIL (Ontology Inference Layer) to form the upper level of languages intended for Web ontology description.

3.5 Data and Workflow

Each document in the subsystem of an electronic workflow can be considered as a separate information flow that has its own model of data representation. The component task consists of transforming the information contained in the documents into a form matching the geoinformation system ontology. This task solving is complicated by the fact that documents contain verbal or weakly structured information. Therefore, it is necessary to accurately interpret all introduced notions and to impart meaningful content to all objects and relationships. Weakness of structures enlisted in model development requires a rigid semantic mapping between the objects being operated with.

Signs of a document's verbal elements are determined to be two dimensional or static; the signs' combination can only be linearly ordered (more accurately, monolinearly ordered) and discrete (i.e., the signs in combinations preserve the separability property).

Signs of the structural components are determined to be one or two dimensional. Their combination should be discrete and polylinearly ordered; otherwise, the order and relationships between characters and signs are imposed by net, hierarchical, relational, and discrete parametric schemata.

Thus, the subsystem of an electronic workflow is intended for the documents' information transformation and its further use in geoinformation systems.

As a rule, the workflow subsystem is adjusted to a definite subject area in accordance with the developed system's purpose.

4 Scenario Approach

The scenario is one of the key IGIS notions. This notion describes various processes almost seamlessly as compared with, say, an algorithm. In a way, the scenario is a generalization of an algorithm.

4.1 Scenario Definition

In the spatial environment being modeled by GIS, various processes take place, including spatial ones (which are the most complex). The process scope encompasses natural process and various human activities. These processes can be complex and include interactions between many heterogeneous objects. The process complexity may be hierarchical, when more global processes are a result of conjoint running and mutual influence between separate, specific processes, which in turn are decomposed into more elementary processes.

Different events can be external manifestations of spatial processes in various points of space, as well as some changes in space.

The following events can be separated:

- appearance and disappearance of objects in different space points;
- motion of point objects on different trajectories;
- change of elongated objects in shapes and sizes according to different laws, etc.

For visual computer modeling, the spatial processes can possibly be described by scenarios.

Let us give some definitions:

- *Scenario* can formally be determined as a sequence of stages and decisions.
- *Stage* is an aggregate of elementary actions performed sequentially or concurrently.
- *Decision* is a point where the process flow may change its direction depending on certain conditions arising at the current moment. So, formally the decision is determined as an aggregate of branches (directions of scenario continuation). Realization of the decision-making procedure (i.e., a choice of direction for further scenario development) can be different: it can be automatic (program) or manual (including the human assistance).
- *Actions* are "construction" blocks for scenarios. They represent specific elements of the scenario participants' actions, which may have different realizations. Some events or changes in space may take place as a result of actions (Fig. 2).

Unlike algorithms, the actions belonging to some stage can be executed sequentially as well as concurrently.

Using mathematical modeling and simulation, it is theoretically possible to model processes with any desired accuracy. The reason is that any action could be described by its *particular scenario* whose elementary actions could in turn have their particular execution scenarios, and so on ad infinitum. Consequently, in scenario decomposition, it is important to determine the level of elementary action detailing. One of the OOA principles—namely, the redundancy depletion at the event or entity formal description—should be kept in mind.

The detailing level first should be determined by modeling objectives, not just by computer tool capacities. The detailing level of the constituents of separate processes should be almost the same; otherwise, the advantage of detailed modeling for some actions would be nullified by rough modeling for other actions. In other words, the same scale order should be considered. In mathematics, such an idea could be illustrated by infinitesimals of different orders.

In visual modeling, the scenarios of spatial processes are played back on a computer in real or arbitrary time. In GIS, such modeling results are displayed on the background of a map as different spatial events or changes. The results of multiple playbacks of scenarios in spatial processes could be used to elaborate their statistical estimations and substantiate decisions for definite action modes. The scenarios are used not just for modeling; they can also be extremely fruitful in formalization of the subject area's business logic and for realization of real-time systems.

4.2 Expert Systems

An expert system is an aggregate of the interacting software tools using expert knowledge (i.e., experts' or specialists' knowledge) to assure highly effective problem solving within a certain subject area. The previously mentioned software systems, as a rule, represent knowledge symbolically, can explain the reasoning of (study) processes, and are intended for the subject areas where humans need to be specially trained for years to acquire the necessary skills. For an ES, the development technology itself requires a specific form of interaction between the ES developer (usually called a knowledge engineer) and one or several subject area experts. The knowledge engineer "extracts" from the experts' procedures, strategies, and empiric rules, which are later used in problem solving and are embedded into ESs.

Standard composition of an ES integrated in GIS consists of the following main constituents:

1. KB, which contains facts (data) and rules (or other data representations) that use these facts as a basis for decision making.
2. IM (mechanism), which is used in the search for the process of problem solving based on knowledge from the KB.
3. User interface, which supports external interactions with the ES. Taking into consideration the ES integration into geoinformation systems, the user interface would be first and foremost the GIS interface.

Apart from these components, the GIS-integrated ES may include some other software tools, as determined by specific characteristics of the problems to be solved.

The KB is undoubtedly the major and most valuable constituent of the ES for IGIS. The process of development differs from the development of conventional software tools, determined by a specific characteristic of the special information type of knowledge.

At least five steps can be singled out in the process of transforming data into knowledge:

- *Knowledge is true*. The system containing knowledge trusts it completely: its statements are true. Should the system reveal certain faults in some of its data fragments, the fragments would immediately lose the status of knowledge and would be deleted or properly modified;
- *Knowledge is abstract*. Disregarding its empirical origin, knowledge is separated and extracted from the subject content and separated and removed from reality;
- *Knowledge is interpretable*. Knowledge tends to express its authenticity in some definite subject realization;
- *Knowledge is active*. Knowledge possesses an intrinsic capacity to be activated and to assign on its own will the states to conjugated information structures;

- *Knowledge is structured.* Knowledge is inside a stable frame originated by diverse structural units and levels.

4.3 Inference Machine

A number of IMs currently exist, with the Rete algorithm being one of the most well-known. The Rete algorithm was developed in 1979 by Dr. Charles L. Forgy at Carnegie-Mellon University, USA (Forgy 1982).

Historically, Rete algorithm-based IMs have been implemented in ESs. These systems are called "expert" because the experts' knowledge in some subject areas is represented there via certain rules. This knowledge mainly has an empiric nature and is accumulated based on the experts' experience in certain subject areas. At the rules' actuation, certain facts can be verified—say, whether the current distance between vessels has become less dangerous or if the distances can be calculated.

In IGIS, the ES may be used to solve the following problems:

- To acquire new information quality at different processing levels in accordance with the JDL model (e.g., measurements, signals, objects, traces, tactical situations, threats);
- To solve some complex hydrometeorological problems, such as prognosis, forecast, subject maps, etc;
- To assure a vessel's navigation safety;
- To manage tactical situations within the zone of shipping control system responsibility;
- To assist the decision maker with emergency management and many other tasks.

Within the frames of conventional use, the IM deals with *empirical* rules put into the ES DB.

However, Rete algorithms, along with empirical knowledge areas, can be successfully used in well-structured knowledge areas where rigorous mathematical theories containing axioms and formal rules exist. The Rete algorithm ideally matches to get complex parallel processes modeled. Processes of this kind are being studied by geoinformation scientists.

4.4 Inference Machine as a System Supervisor

To start, let us consider a comparatively simple task of monitoring navigation safety. To facilitate the given example, consideration of only the reciprocal vessels' collision, particularly the "high-seas version," will be used. This problem could be reduced to defining the distance between all pairs of vessels being monitored and comparing distances with the safe distance. Using standard programming, this task solving is performed by a dual cycle: externally according to all vessels and internally according to vessels yet in question. At a total number of 100 vessels, this requires 5,000 verifications in one iteration.

How often should such verifications be performed? The answer depends on the current distance between the vessels in pairs and their approach speed. In close proximity, even a verification each second might be insufficient; however, verifications might be done once an hour when vessels are far apart (say, hundreds of miles). Nevertheless, an algorithm based on a dual cycle should work at a frequency to assure the navigation safety (i.e., with a maximal frequency). Thus, such a solution is as ineffective as the case of the IM linear realization.

Using an IM based on the Rete algorithm, this task could be optimally solved by one rather simple rule. The verifications' frequency would functionally depend on current distances between vessels and their speeds. The above rule in CLIPS language is as follows:

```
(Rule 1)
(Current - Time ?cur.)
(Check-Dist. (vessel1 ?v1)(vessel2 ?v2) (time ?time) )
( test ( < ?time ?cur. ) )
=>
( if ( < ( Distance ?C1 C2 )( Foul-Distance ?v1 v2 ) )
then
( assert (Risk of collision (vessel1 ?v1)(vessel 2 ?v2) )
else
( assert (Check-Dist. (vessel1 ?v1)(vessel 2 ?v2)) (time
( + ?cur. ( min - Heading in - time ?v1 ?v2 ))) )) ) )
```

The rule content will be reduced to:

1. Fact (*Current time? cur*) is given each second (or with other maximal frequency).
2. Slot value (*time*? *tm*) for a fact. (*Check-Dist*) stores the astronomical time when it is necessary to run a regular verification of a distance between the given pair of vessels. Should the current time override the astronomical time, the rule would trigger.
3. At the rule triggering, the distance between vessels is verified against the fact of the distance becoming shorter than the dangerous distance. Matching functions are evoked to calculate the above distances.
4. If the current distance between vessels became shorter than the dangerous distance, then the fact alerting the collision danger is confirmed.
5. In the opposite case, the fact is confirmed that would evoke the distance retesting in a time specified by a function: *min-time-appr*.

4.5 Expert System as the Core of the Decision-Making Support Subsystem

Implementation of the ES as a tool for GIS-based managerial support and decision making assures the appearance of qualitatively new possibilities in modeling geospatial processes and in substantiation of complex managerial decisions. These new possibilities include the following:

- Fast recognition of typical situations in the applied subject area and a well-grounded offer for an operator (user) of the matching actions;
- Assurance of the operator's (user's) self-control of actions and decisions in regard to the management of applied subject area processes;
- Monitoring the state of controlled processes in the subject area over different criteria in real time;
- Arranging the information and reference assistance for the operator (user) at the stage of adaptation to IGIS functional capacities;
- Intelligent analysis of spatial-time activity of heterogeneous moving objects;
- Visual development of functioning models (scenarios, actions) for the objects in GIS;
- Playback of objects' action scenarios in real and arbitrary times as accompanied by visual mapping in the form of conventional signs on an electronic map background;
- Recommendations for decision makers in scenarios playing at the stage of research design for systems, business games, situation analysis, personnel learning, and training.

ES is used as a standard tool for a decision maker's intelligent support and as a system intended for managing various processes, such as modeling.

4.6 Case Studies of the Scenario Approach

The results of ES work based on the scenario approach can be represented to the user as interactive promptings or dialogues within a specified research area. Figure 3 depicts an example of an ES composed of interactive promptings. Obviously, it would be hardly possible to visually distinguish the ES work from the script language. However, the advantages of the proposed approach become obvious when accounting for the fact that scenarios and DBs can be developed in a visual medium using the full power of OOA. Figure 3 depicts the tactical situation of modeling special actions within a closed area; resulting from the previous situation analysis, the ES embedded in GIS generates prompts for decision makers (Fig. 4).

These prompts could be accompanied by voice and some other effects, such as visual cues. Luckily, the scenario allows the realization of practically any functionality and logic for making the most complex decisions based on first-order predicate algebra.

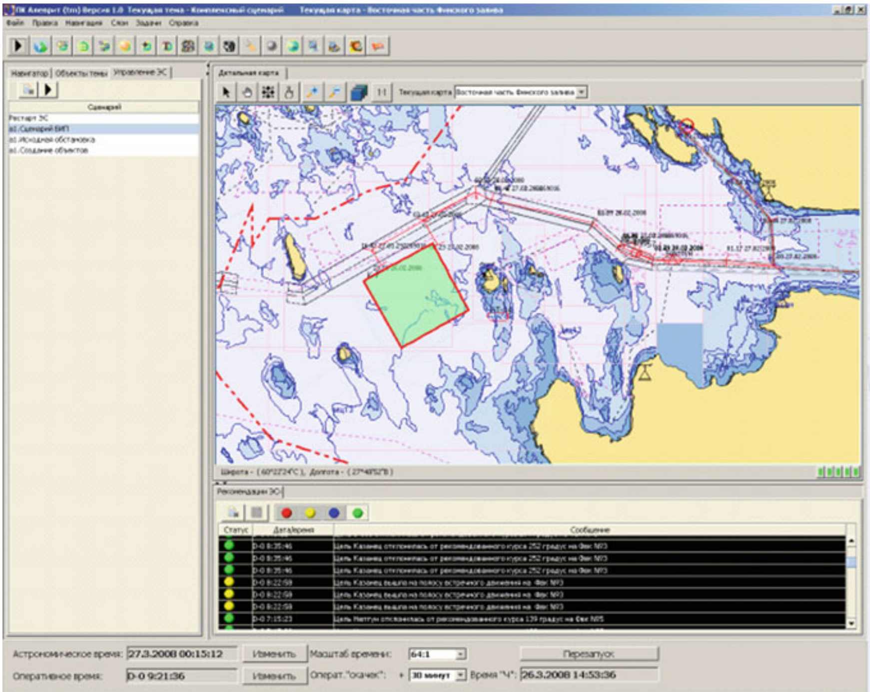


Fig. 4 The general structure of a DMOS based on a JDL model

5 Case Study

As an example of the ideas proposed in this chapter, the realization of the maritime surveillance and recognition systems (MSRS) developed by SPIIRAS and SPIIRAS-NTBVT Ltd. will be discussed. The given system architecture was developed based upon the concept of data harmonization, integration, and fusion in IGIS (Popovich and Voronin 2005, Popovich 2009).

The process of information harmonization assumes a distribution of main notions (concepts) and their relationships (ontologies) over the matching subject areas and/or responsibility areas. For instance, the partition could be executed over existing knowledge areas, such as hydroacoustics, radiolocation, theory of search, etc.

Information integration allows the fusion of information from heterogeneous sources and access to information resources for the current task being solved (modeling). The distinguishing feature of information integration is the fact that the result is always aimed at solving a definite class of problems.

In the literature (e.g., Blasch 2002), the notions of data fusion and information fusion are often separated. Data fusion is understood as the combination of organized data intended for analysis and decision making, whereas information fusion is the combination of data intended for knowledge acquisition or as a process of fusing data from heterogeneous sources. The concept of “data fusion” was introduced in the beginning of the 1990s (Blasch 2002). In IGIS, fusion is understood as the appearance of a new quality of information rather than an improved quality of information (see Fig. 5).

MSRS is used for information, model, and graphic support while typical functional tasks are being solved by managing officials at all situation coverage levels. The system is an information base for an integrated automated control system at all control levels and stages for different customers: military, emergency management, vessel traffic managers, regional governments, etc.

MSRS is capable of information and technical integration with necessary information sources, observation stations, data acquisition, and processing stations at horizontal and vertical levels. An example of a visual representation of the air, surface, and coastal situation for the Northern Baltic Sea is shown in Fig. 6.

The main functions of MSRS include the following:

- Integrated information space generation for specific tasks and analysis of the current situations. An example of a visual representation of navigation situation based on the S57 format is shown in Fig. 7.

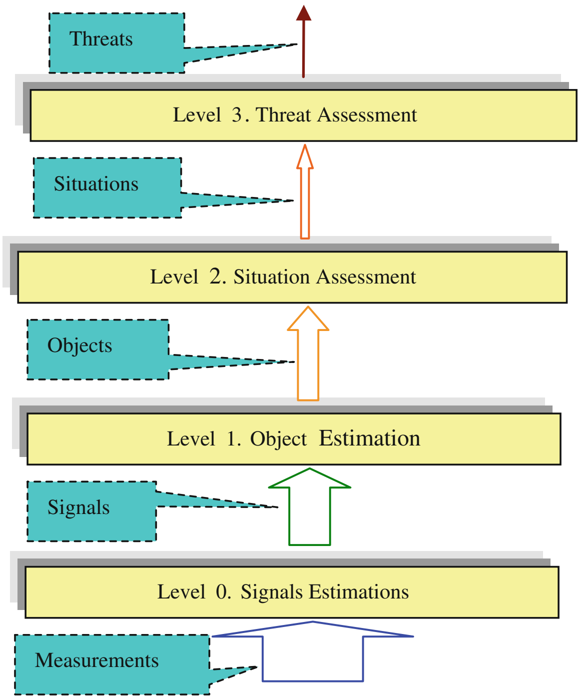


Fig. 5 Information fusion levels

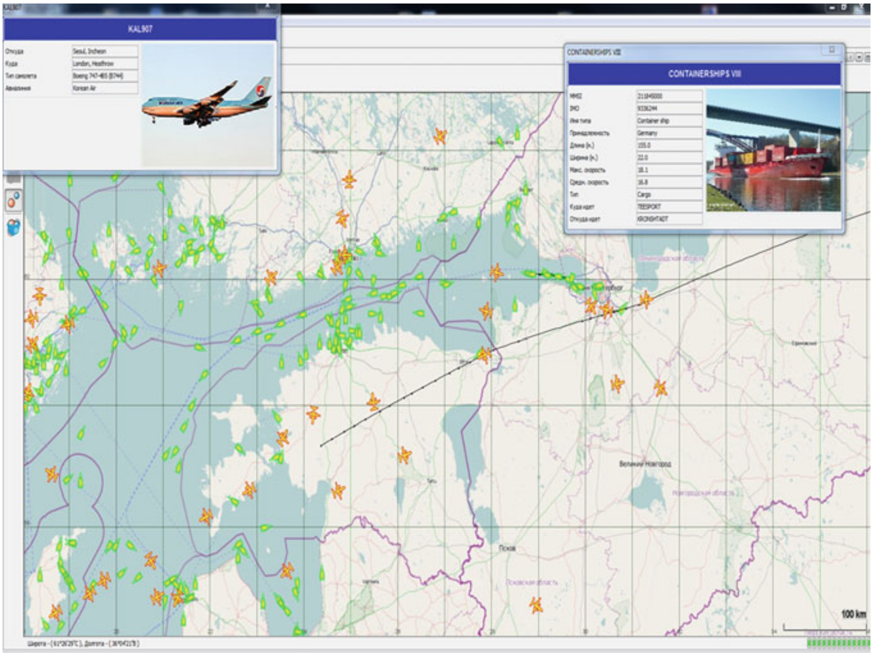


Fig. 6 An example of a visual representation of air, surface, and coastal situations

- Processing of heterogeneous information in a unified coordinate environment at the data entry pace, as well as its timely distribution and transfer to consumers at a desired pace and scope (see Figs. 6, 7).
- Continuous monitoring of moving targets, assessment of their state and environment (including weather conditions), and elaboration of summary data about the current operational situation in the zone of responsibility of the typical observation station.
- Early detection of dangerous targets, situations, and threats, as well as intelligent support of decision making using the scenario-based approach.
- Generation of recommendations on the adoption of adequate measures for strategic, operational, and tactical levels to prevent potential threats (weather, ice conditions, the threat of terrorists and pirates), based on the current situational data.
- Management of mobile surveillance equipment for operational data validation and refining.
- Simulation for assessing and predicting situations during the planning phase for optimal decision making.
- Surveillance system optimization at the construction and functioning stages, based on simulation and modeling using means and methods of artificial intellectuality.

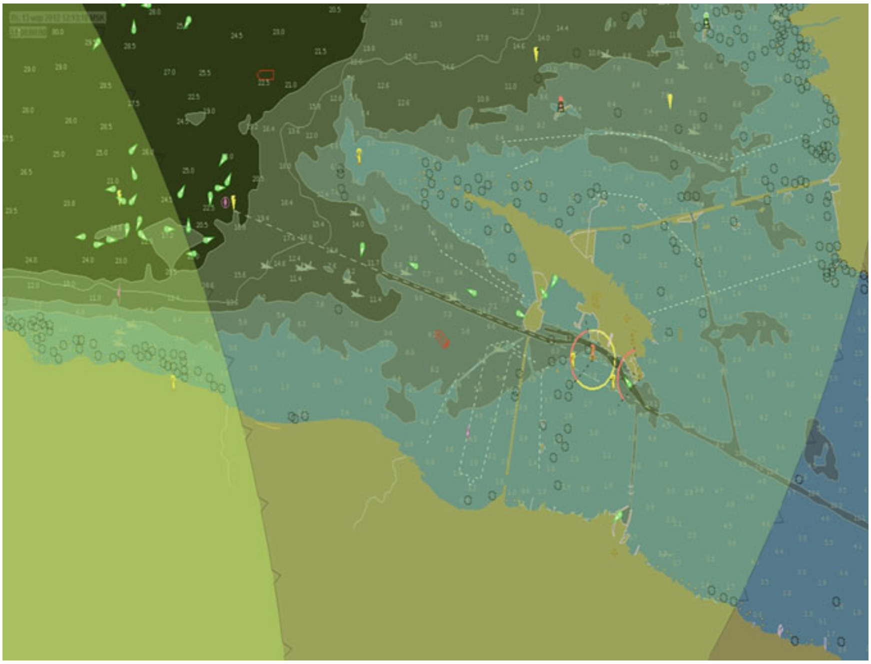


Fig. 7 An example of a visual representation of a navigation situation based on the S57 format

The system is capable of real-time (information update cycle: 1–5 min) tracking up to 100,000 moving targets. It is possible only by using IM. The scenario approach allows one to change business logic on the fly while the system is working.

Let us consider a very interesting and important subsystem of MSRS—the sonar calculation subsystem (SCS) (Popovich et al. 2009). SCS provides calculations (modeling) of acoustic fields for the estimation of sonar range, calculation and simulation of observation zones, and tactical situation conduct based on sonar data. The results of these calculations allow one to resolve a number of challenges related to the optimal placement of a positional sonar's optimization search operation, such as fishing, searching for sunken objects, searching for moving underwater and surface targets, etc.

The IGIS interface allows a 3D + t picture of transmission losses (see Fig. 8) and target detection zone distribution on a digital map. ES and IM help end users to understand current tactical situations and make the right decisions. MSRS and SCS contain different sets of ontologies, but they work together under the system of scenarios and sets of mathematical and simulation models.

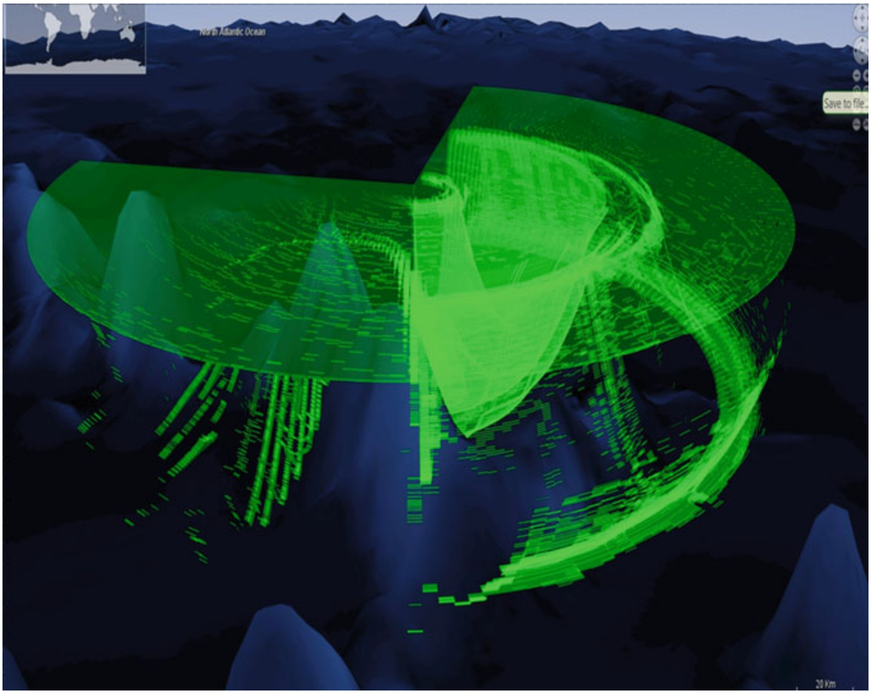


Fig. 8 3D visual representation of sonar's capabilities

6 Conclusion

The ideas presented in this chapter regarding IGIS could be joined in a specific subject domain. IGIS theory is one more attempt to fight complexity. This complexity is considered in two aspects: software development complexity and the complexity of decision-making support in some given location at given time. The proposed approach conditionally could be divided into three interrelated constituents:

1. Universal model of geospatial data: The core of the new approach in geoinformatics that allows for abandoning the old chapter maps heritage.
2. An intelligent subsystem that incorporates at least a KB using the ontology approach and an IM using algebra of first-order predicate logic.
3. A scenario approach based on the concept-generalized notion of algorithms. This approach opens up a wide variety of possibilities in formalization and automation of the subject areas' business logic.

Practical approbation of the proposed theoretical and technological ideas realized through serially developed information systems (heterogeneous and distributed) encourages the realization of the given ideas in a number of subject areas related to geoinformatics and software system development in general.

References

Blasch E (2002) Fundamentals of information fusion and applications. Tutorial, TD2, Fusion

Booch G (1995) Object solutions: managing the object-oriented project. Pearson Education, New Jersey, ISBN 0-8053-0594-7

Brooks F (1975) The mythical man-month. Addison-Wesley, Cambridge, ISBN 0201835959

DAML + OIL (2001) Reference description URL: http://www.w3.org/TR/daml+oil-reference

Edward Yourdon (1989) Modern structured analysis. Prentice Hall, New Jersey

Forgy C (1982) Rete: a fast algorithm for the many pattern/many object pattern match problem. Artif Intell 19(198):17–37

Gruninger M (1995) Methodology for the design and evaluation of ontologies. In: Proceedings of the workshop on basic ontological issues in knowledge sharing, IJCAI-95, Montreal

Kant I (1889) Kritik der reinen Vernunft. 1781

Popovich VV (2003) Concept of geoinformatic systems for information fusion. In: Proceedings of the 1st international workshop on information fusion and geographic information system, St. Petersburg, 17–20 Sept 2003

Popovich V (2009) Concept for corporate information systems based on GIS technologies development. In: Proceedings of IF&GIS-09, St. Petersburg, Springer//Lecture Notes in Geoinformation and Cartography, 17–20 May 2009, p 371

Popovich V, Voronin M (2005) Data harmonization, integration and fusion: three sources and three major components of geoinformation technologies. In: Proceedings of IF&GIS, St. Petersburg, 25–27 Sept 2005

Popovich VV, Shpakov MV, Shaida SS (2004) Ontology modeler for geographic information systems applications. In: Proceedings of 10th EC-GI&GIS workshop. Warsaw, 23–25 June 2004

Popovich VV, Potapichev SN, Sorokin RP, Pankin AV (2005) Intelligent GIS for monitoring systems development. In: Proceedings of CORP2005, University of Technology Vienna, Vienna, 22–25 Feb 2005

Popovich V, Aviloff C, Ermolaev V, Zinyakov Y, Leontev Y, Korolenko K, Popov O (2009) Sound propagation modelling on intelligent GIS base. In: Papadakis JS, Bjørnø L (eds) Proceedings of UAM 2009, June 21–26 2009, Nafplion

Popovich V, Vanurin S, Kokh S, Kuzyonny V (2011) Intellectual geographic information system for navigation safety "Systems", vol 25(8), pp 29–31, Aug 2011

Sorokin R (2005) Application of artificial intelligence methods in geographic information systems. In: Ivakin Y, Sorokin R (eds.) Proceedings of international workshop information fusion and geografic information systems (IF&GIS2005), St. Petersburg, 25–27 Sept 2005, pp 105–114

Tarski A (2004) Life and logic. Anita Burdman Feferman; Solomon Feferman, Stanford University, California. Hardback. ISBN: 9780521802406

Ten Years of the International Workshop "Information Fusion and Geoinformation Systems"

Rafael Yusupov and Yan Ivakin

Abstract In 2013, it will be 10 years since the first international workshop "Information Fusion and Geoinformation Systems" (IF&GIS). Over the past decade, the IF&GIS workshop has become a noticeable and important scientific forum, internationally recognized by professionals working with geographic information systems and many environmental and urban applications. This chapter considers the essential results and major accomplishments of the IF&GIS workshop.

Keywords Intelligent geoinformation systems (IGIS) · GIS integration and fusion · Artificial intelligence · Data and knowledge · Environmental and urban GIS · GIS technologies and software

1 Introduction

''Information Fusion and Geoinformation Systems'' (IF&GIS) is a biannual international workshop, intended for the consolidation of efforts of scientists in the theory and applications of geographic information systems (GIS), GIS integration and fusion, data and knowledge for GIS applications. IF&GIS is a scientific forum, internationally recognized by academics, professionals from the software industry, and many environmental and urban fields. The tenth anniversary of the international IF&GIS workshop is the right time to define the first significant results of its work in GIS research and related areas. The main results of the workshop include the following:

- emergence of an original subject dedicated to knowledge-based GIS, reflecting a strong demand from practitioners and the current software industry;

R. Yusupov · Y. Ivakin (✉)
SPIIRAS, St. Petersburg, Line 14, 39 199178 Petersburg, Russia
e-mail: ivakin@oogis.ru
URL: http://www.oogis.ru

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_3,

- wide recognition of representativeness as well as professional and research level of the workshop as an international platform for exchange of views between scientists engaged in the development of GIS, information fusion methodology, procedures and means for intelligent GIS (IGIS), and GIS-based software;
- making of special format in the workshop holding, varying from the traditional forms of similar scientific events to be oriented to the needs of scientists doing environmental and urban research, as well as to the needs of professionals doing applied research and accounting for the interests of software application users;
- assurance of future scientific prospects, with the possibility of involving a wider range of participants working in the above-specified and related research areas.

Overall, these results demonstrate the scope of the IF&GIS workshop to develop an independent strategy of scientific views on IGIS, range of information integration problems based upon them. This chapter presents the results of past IF&GIS workshops to demonstrate the essence of this strategy as a whole.

2 Subject Concepts

Subjects of the international IF&GIS workshop harmonically combine highly specified issues of research in the subject area of GIS with elements of artificial and hybrid intelligence, data, and knowledge fusion in reference to the corresponding GIS applications, along with actual problems of applications for the desired and related areas of research. Thus, the invited speakers' presentations and the corresponding scope of regular sessions, which often include presentations from world-class researchers, consider quite important results from basic research. Some examples are given below:

2011:
Jean-Claude Thill (USA) "Is Spatial Really That Special? A Tale of Spaces" (Thill 2011);

Bin Jiang (Sweden) "A Shot Note on Data-Intensive Geospatial Computing" (Jiang 2011);

2009:
Gennady Andrienko (Germany) "Visual Analytics for Geographic Analysis, Exemplified by Different Types of Movement Data" (Andrienko 2009);

2007:
Gabriel Jakobson (USA) "Situation Management: Basic Concepts and Approaches" (Jakobson 2007);
Christophe Claramunt (France) "Maritime GIS: From Monitoring to Simulation Systems" (Claramunt 2007);

2005:

Mieczyslaw Kokar (USA) "Ontology Development: A Parameterization" (Kokar 2005);

Katia Sycara (USA) "Geo-Spatial Reasoning Context for High Level Information Fusion" (Sycara 2005);

Manfred Schrenk (Austria) "Planning in a Rapidly Changing World—Challenges and Thoughts on 'Dynamic Planning'" (Schrenk 2005);

2003:

Vasily Popovich (Russia) "Concept of GIS for Information Fusion" (Popovich 2003).

Moreover, presentations delivered by workshop participants in the regular session allowed a variety of applied aspects of information fusion and GIS to be covered. In particular, these presentations focused on the issues of information fusion in logistics (Smirnov et al. 2003), solving environment management problems (Evertsen et al. 2005), planning and design of urban development (Schrenk and Pozare 2007), knowledge-based search of GIS objects (Prokaev 2005), solving track analysis problems (Makshanov and Prokaev 2007), and many other scientific and practical issues. Traditional sessions were combined with workshopes, such as, "Harmonization, integration and fusion of data, information and knowledge in GIS," "Ontologies and software technologies for GIS and GIS applications," and "Maritime GIS," with nonrecurring sessions about actual issues in modern life, such as "Operations research and theory of search for mobile objects in GIS applications"—2005, 2007; "GIS as the basis for monitoring systems"—2007; "Corporate information systems and GIS"—2009; "GIS based monitoring of terroristic threat"—2009; "Prospective maritime GIS research areas"—2011. These sessions highlight the diversity of the scientific subjects and identity of the international IF&GIS workshop.

Consistency of the main participants' composition, in combination with progressive growth of the IF&GIS Program Committee, ensures preservation of the integrity and evolvement of the workshop's subject concepts. The program committee has grown from seven members in 2003 to 44 members in 2011. Today, the program committee includes leading scientists and application specialists in the area of IGIS from all over the world, including representatives of St. Petersburg Institute for informatics and automation of Russian Academy of Sciences (SPIIRAS) and representatives of a wide range universities from USA, Canada, Sweden, France, Ireland, Belgium, Germany, Great Britain, Russia, Mexico, Poland, China, and Switzerland; representatives of research institutes and companies from Austria, Germany, USA, and Greece; and representatives of the Russian and French Naval Academies. The Program Committee has focused on specifying certain crucial issues to be included in the workshop scope, as well as thoroughly reviewing the submitted materials and resulting proceedings. Their work has allowed the workshop to achieve a consistently high reputation in the world research community, scientific press, and many international research institutes.

3 Scope and Scientific Level

Over its decade of its existence, the international IF&GIS workshop has significantly expanded its scope, in both quantity and quality. The number of workshop participants has increased by a factor of 5–6. The focus of interest of specialists participating in the workshop and the number of young scientists, and software developers participating in the event's activities constantly increases. Also continuously expanding is the geography of interested universities, research institutes, and different companies whose representatives take part in IF&GIS sessions. Such growth of the workshop has allowed it to be held not only in the home city of its founder, SPIIRAS (St. Petersburg, Russia), but also it in other cities of Western Europe and Russia. In this regard, IF&GIS-2011 was remarkable: it was held in Brest (France), within the scope of wide- range multi-conference "SaferSeas". This fact also contributes to the workshop's popularity and facilitates its accessibility for more GIS specialists around the world.

Moreover, the increasing popularity of the international IF&GIS workshop has not decreased its scientific level at all. The foremost credit in this respect should be given to the program committee, science editors, and the reviewers of the workshop proceedings. The session chairmen and the workshop session participants also greatly contribute, and their efforts are obviously aimed at maintaining the event's high level of professionalism. Active discussion of all presentations at the workshop sessions and thus expert analysis of scientific concepts is the target and the purpose of IF&GIS. Namely, the efforts of all those who participated in preparing and holding the international workshop allow the achievement of a combination of subject diversity, system integrity, and the highest scientific level of materials, as published in IF&GIS proceedings. (The workshop proceedings are traditionally published after each workshop). To illustrate this thesis, Fig. 1 shows the ratio of the number of presentations delivered at the workshop sessions and published in the proceedings.

The cooperation established by the Organizing Committee with the international scientific publishing house Springer (Germany) largely contributes to maintaining the workshop's scientific level. Since 2007, the workshop proceedings have been published by Springer in its series "Lecture Notes in Geoinformation and Cartography". To a great extent, this has expanded the sphere of influence of the international IF&GIS workshop; at the same time, it preserves the essentially high level of scientific and publishing standards. All the workshop materials are useful and available to scientists and the users, developers, and marketing specialists in the GIS technologies area. The apparent increase of the number of young workshop participants from among the universities' postgraduate students, who are starting their professional careers, serves proof of the fact.

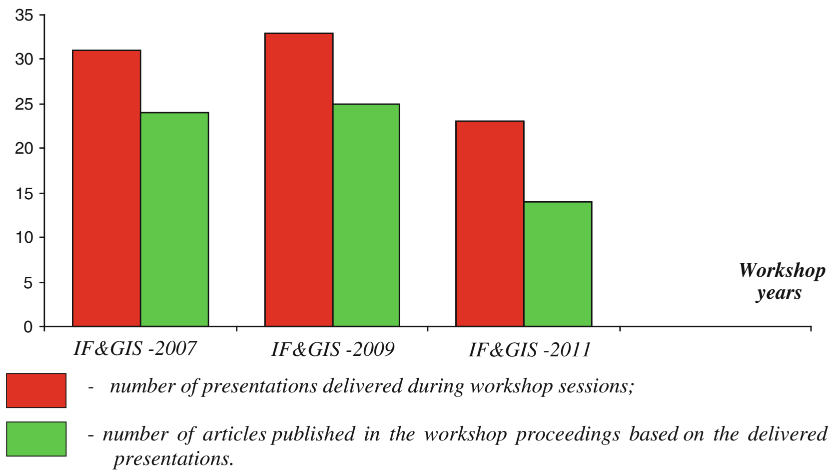


Fig. 1 The ratio of the presentations delivered and published in the workshop proceedings

4 Format Originality

Establishment of a special workshop format that differs from traditional forms of similar scientific events is a distinctive feature of IF&GIS. This manifests itself in several ways:

- longstanding maintenance of the international science workshop status and format, without transforming the event into a multisession scientific conference. This status and format are the most convenient for professional interactions of specialists and top-level scientists;
- rejection of parallel subject-oriented workshop sessions in favor of successive subject-dedicated sessions. This gives the workshop participants an opportunity to be a part of any workshop activity and, thus, successfully combine their scientific and general development interests;
- combining traditional workshop science sessions with various application-oriented public events.

It is appropriate to mention here some of the public events that took place over the years:

- 2005: a demonstration of practical implementation of geoinformation technologies that were represented at the workshop in the software–hardware monitoring system “Ontomap,” developed by SPIIRAS;
- 2007: an exhibition of the leading research organizations and commercial companies involved in geoinformation products and services, as represented in the St. Petersburg information technologies market;

- 2009: tutorials by Gabriel Jacobson, dedicated to situational management, and by Vasiliy Popovich and Ruslan Sorokin, dedicated to SPIIRAS intellectualized GIS;
- 2011: participation in the exhibition "SaferSeas" and collective session with the 4th international conference on "Geospatial Semantics" (GeoS 2011) (Claramunt et al. 2011);
- Each year, the traditionally broad workshop scope has focused on the up-to-date issues of the current global geoinformatics development stage and its methodology, methods, practices, and technologies. The highlighted topics included the following:
  - 2003: current issues in search and track analyses for GIS applications;
  - 2005: methods and technologies for GIS intellectualization;
  - 2007: situational modeling and situational analysis using GIS and technologies;
  - 2009: geoinformation technologies for developing corporate information systems;
  - 2011: GIS and digital ocean—problems of ensuring safe and efficient human activities in the world's ocean using state-of-the-art geoinformation technologies;
  - 2013: GIS applications for environment and urban planning research, as well as for current works in the exploration of the Arctic and the adjacent regions;
- the priority of representation at the international workshop sessions of scientific results in basic research over particular engineering developments;
- openness of the workshop format, i.e., continuous search for new intelligible and popular ways to present current, highly sophisticated, and capacious geoinformatics concepts to the scientific community.
- The Organizing Committee does not consider the existing format of the IF&GIS to be permanent and invariable; therefore, continuous work is being performed and the participants' suggestions are accepted with due regard for introduction of new, unorthodox forms of running certain workshop activities. At that, the general architectonics of the organization of the Program and Organizing committees, reviewers, and editing group who conduct of the main workshop sessions remain the same. Thus, the classic, strict scientific workshop form and flexible modern innovations are well balanced within the workshop format.

5 Future Prospects

The founders and organizers of the international IF&GIS workshop associate the workshop's future prospects with continuous qualitative improvement. This growth under the current conditions is manifested by the possibility of engaging more and more participants who work professionally in geoinformation and related research areas; in the search for subject concepts and their presentation at the sessions; in publishing in the proceedings of new, unique scientific concepts; and in advanced developments and the quest for program-technical solutions. Also, this growth will manifest itself in expansion of the range of events, held within the IF&GIS scope; expansion of the geography of the workshop venues; and expansion of the number of organizations, companies, and universities represented by the workshop participants. Possible future directions of the workshop development will be enhanced by cooperation with international and domestic scientific publishing houses:

Development of relationships with the scientific publishing house Springer-Verlag GmbH has been planned. Within the frames of the above cooperation, the IF&GIS workshop will serve as an approbation platform for publishing a scientific monograph covering the intellectualized GIS problems. Further, this project may evolve into a number of scientific developments, represented and evaluated at the international IF&GIS workshop;

It is intended to establish business contacts aimed at future publishing of workshop proceedings by such scientific publishing houses as Pergamon Press (USA) and Nauka (Russia).

Needless to say, the current state and future development of IF&GIS are inseparable from the sponsors' support. So, it is worth noting that throughout the entire decade of the workshop's existence, its continuous sponsors have been and remain:

- Office of Naval Research Global (USA);
- The Russian Academy of Sciences (Russia).

Periodic sponsor support in financing the workshop operations was received from:

- Federal Agency for Science of the Ministry of Industry, Science and Technology of the Russian Federation (Russia)—2003;
- US Army Research Laboratory, European Research Office—2003, 2005, 2007;
- the Russian Foundation for Basic Research (RFFR)—2009;
- Naval Academy Research Institute (France)—2011.

The workshop information support was provided by:

- Binghamton University (USA);
- Fort-Ross Company, official representative of the Russian software developers association ROSSSOFT (Russia);
- The popular science magazine *Machines and Mechanisms* (Russia);
- Science journal *Information and Space* (Russia);
- Science magazine *Information Management Systems* (Russia);
- Science, research and educational, interdisciplinary journal specializing in informatics, automation and applied mathematics *SPIIRAS Proceedings* (Russia);
- CEIT ALANOVA GmbH (Austria);

The permanent service agent of the workshop is the Russian company Monomax. The founders and the Organizing Committee of the international IF&GIS workshop thank the above companies for longstanding cooperation and support and express hope for their continuation. The Organizing Committee also hopes and continuously works on expanding the range of the workshop sponsors, friends, and supporters in promotion of GIS advanced research ideas and technologies to the state-of-the-art software applications and geoinformation services market. The accumulated experience in organizing and holding this presented international workshop, with its high scientific authority and large number of partners around the world, allow the IF&GIS Organizing Committee to look ahead with confidence, with a clear perspective for perfection and development.

6 Conclusion

By summarizing the main significant results of the international IF&GIS workshop, one can state a single resumptive fact: IF&GIS is an accomplished representative scientific forum. Over 10 years of its existence, it has made its way from a temporary event to a regular, authoritative scientific gathering of scientists and professionals specializing in geoinformatics and applied GIS technologies. Today, it is a developed institution of public discussion and evaluation of scientific concepts, with solid future prospects. This fact allows the organizers to make further ambitious plans and set the priorities for its growth.

The overview of the main IF&GIS establishment milestones, given in this chapter, does not claim to be exhaustive. The workshop, as a scientific and social event, is the fruit of labor of hundreds of people who contribute their efforts, knowledge, and skills for the workshop to take place. Further IF&GIS development will be the best acknowledgment of and reward for their endeavors.

References

Andrienko G (2009).Visual analytics for geographic analysis, exemplified by different types of movement data. In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 17–20 May 2009, pp 3–18

Claramunt C (2007). Maritime GIS: from monitoring to simulation systems. In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 27–29 May 2007, pp 34–44

Claramunt C, Levashkin S, Bertolotto M (eds) (2011) Proceedings 4th international conference geospatial semantics, Brest, 12–13 May 2011

Evertsen G, Tollefsen T, Jokulsson G, Finnseth A (2005). Integrated relevant information and geographical maps in a flood management systems. In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 25–27 Sept 2005, pp 35–41

Jakobson G (2007) Situation management: basic concepts and approaches. In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 27–29 May 2007, pp 18–33
Jiang B (2011). A shot note on data-intensive geospatial computing. In: Proceedings international workshop information fusion and geographic information systems: towards the digital ocean, Brest, 10–11 May 2011, pp 12–17
Kokar M (2005). Ontology development: a parameterization. In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 25–27 Sept 2005, pp 15–17
Makshanov A, Prokaev A (2007) Empirical bayes trajectory estimation on the base of bearings from moving observer. In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 27–29 May 2007, pp 323–334
Popovich V (2003). Concept of geoinformatic systems for information fusion. In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 17–20 Sept 2003, pp 83–97
Prokaev A (2005) Intelligent geoinformation systems based information integration in the objects' search tasks. In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 25–27 Sept 2005, pp 54–64
Schrenk M (2005). Planning in a rapidly changing world—challenges and thoughts on «dynamic planning». In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 25–27 Sept 2005, pp 6–12
Schrenk M, Pozare W (2007) «Centore MAP»: combining X-border information from multiple sources for planning purposes. In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 27–29 May 2007, pp 96–110
Smirnov A, Pashkin M, Shilov N, Levashova T, Krizhanovsky A (2003) Knowledge logistics in open information environment: KSNet-approach and portable hospital configuration case study. In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 17–20 Sept 2003, pp 134–144
Sycara K (2005).Geo-spatial reasoning context for high level information fusion. In: Proceedings international workshop information fusion and geographic information systems, St. Petersburg, 25–27 Sept 2005, pp 13–14
Thill JC (2011). Is spatial really that special? A tale of spaces. In: Proceedings international workshop information fusion and geographic information systems: towards the digital ocean, Brest, 10–11 May 2011, pp 3–12

Part III
Ontologies and Modeling in GIS

Semantic Retrieval of Geospatial Datasets Based on a Semantic Repository

Julio Vizcarra, Miguel Torres, and Rolando Quintero

Abstract In this chapter, we propose a methodology to semantically search geospatial datasets by means of their metadata. It consists of structuring a semantic repository in order to provide the inclusion mechanisms of distributed data to be retrieved, as well as the extraction of those geographic objects with respect to their conceptual similarity. The approach proposes a conceptual measure called the Conceptual Distance algorithm to determine the conceptual similarity among concepts. The features of geographic objects are conceptualized in an ontology on the domains: thematic, spatial, and temporal. The ontology is populated by using Federal Geographic Data Committee specification with instances of geographic objects distributed in several locations on a network. In the retrieval process according to a query, the objects are presented as a ranking list, which provides other results that are semantically close by means of radius of search and it avoids empty results. The approach has been implemented in a web system called *SemGsearch*.

1 Introduction

Nowadays, there is a large volume of collected geographic data that has been obtained by different technologies, such as global positioning system (GPS), satellite images, geographic databases, and maps in analog format, among other

J. Vizcarra (✉) · M. Torres · R. Quintero
Intelligent Processing of Geospatial Information Lab-Centre for Computing Research, National Polytechnic Institute, Mexico City, Mexico
e-mail: rvizcarrab08@sagitario.cic.ipn.mx; kaivzro@gmail.com

M. Torres
e-mail: mtorres@cic.ipn.mx

R. Quintero
e-mail: quintero@cic.ipn.mx
URL: http://www.cic.ipn.mx

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_4,

sources (Sabbouh et al. 2007). It is not only new spatial information systems but also data collection technologies that are becoming more sophisticated. In addition, geospatial data are an important issue in any decision support system (DSS); they can be considered as crucial features for planning and decision making in a variety of applications. It is important to mention that the development of technologies for integration, tools for management, and analysis have increased in recent years (Heywood et al. 2006).

On the other hand, there is not agreement on the representation of geospatial semantics in maps and generally in geospatial representations. For example, different organizations have different levels of accuracy in drawing certain lines, points, or polygons on a plane, in order to represent cities, water supply wells, or altimetry points in the network, transmission lines, road infrastructure, and so on (MacEachren 2004). However, there is no consensus or agreement between organizations or groups of experts about the meaning, semantics, or ontology of these conceptual representations; therefore, each organization has a particular conceptualization. For instance, organizations use the term "artificial lakes," whereas others use the term "dam". The problem now is not how to accurately represent a geographic feature. Rather, the problem occurs when the user is presented with a spatial representation or geographic database but the user understands the naming (semantic annotation) or other cartographic information made by a different organization or another user. This need for a semantic union is the basis for obtaining true interoperability and exchange of geospatial data among different users. Therefore, geographic information systems (GIS) are not exempt from this problem in order to manage and process these data (Zhan et al. 2008).

Most GIS were not originally designed to work with semantic processing. Therefore, there are some problems of interoperability, and the integration of heterogeneous geospatial data sources cannot be achieved in these systems (Buccella et al. 2007). This is mainly because each GIS provides specific requirements for the representation of its data, such as its format and specific query language (Bishr 1998).

The problem in information retrieval is that users can be easily overwhelmed by the amount of information available. The processing time of irrelevant information in documents such as geospatial databases, geospatial images, text, and tables retrieved by an information retrieval system is very chaotic. This condition is the result of inaccuracies in the representation of the documents in the repository, as well as confusion and imprecision in user queries because users are frequently unable to express their needs efficiently. These facts contribute to the loss of information and the provision of irrelevant information.

According to the problems mentioned above, this chapter proposes a methodology for semantic search of geospatial data by means of their metadata. The metadata are structured into a semantic repository, adding a priori knowledge of thematic, spatial, and temporal domains in order to provide mechanisms that include distributed data to be retrieved and the extraction of these geographic objects with respect to their conceptual similarity from several servers.

The chapter is organized as follows: Sect. 2 presents the state of the art related to the work in this field. Section 3 describes the proposed methodology to semantically retrieve geospatial information. In Sect. 4, the experimental results are depicted. Finally, the conclusion and future works are outlined in Sect. 5.

2 Related Work

In this section, several approaches that are used for information retrieval from different data sources have been analyzed to solve the problem of semantic heterogeneity. Some of these works have exploited the metadata, designing descriptors of the information contained in the repositories. Other projects have proposed the use of ontologies. However, the common characteristic is the intelligent search, for which models have been developed to retrieve an object or concept from various sources.

In Janowicz (2007), a theory of similarities to be used in modeling web services is presented. The implementation of the theory is performed using the framework for WMSL, which is based on the intersection of logical descriptors. For the process to give a measure of similarity of a concept, it extracts the attributes or properties of its specification in the WMSL (Sabbouh et al. 2007). Later, a matrix is formed in order to compare with other concepts and their respective matrix. For identical pairs, it is necessary to assign a weight, similar or different for each pair. In Malik et al. (2010), a framework of semantic annotations is proposed. In Kiryakov et al. (2004), an approach to integrate and retrieve information from distributed heterogeneous resources on the Web is described. The semantic annotations defined by the knowledge base within the knowledge discovery processes and control over it are called entities, which are generally interconnected to define a process of discovery of pieces of knowledge or data with little or no human assistance. Another case is presented in Boley et al. (2010), which studied the design, syntax, and implementation of semantics and integration within e-business (Amor 2000). The integration is performed by reconstructing the notion of object relation with shared components. There are high-level languages to describe knowledge and integrate it into the semantic web using positional and slotted techniques and artificial intelligence. A language that defines the concepts and semantic relations between them in order to integrate accurate data between different sources is created (Vilches et al. 2009).

On the other hand, the work presented in Santos et al. (2003) is focused on the discovery and retrieval of spatial data in distributed environments on spatial data infrastructures (SDIs). The discovery and recovery tasks are based on three steps. In the first step, the user's search terms are mapped to concepts in domain ontology based on the hybrid ontology approach (Liang et al. 2007). In the second step, the concepts are extended on the basis of the hierarchy of concepts in the domain ontology. The third step consists of the expansion of the query, in which adequate descriptions of geographic information are sought and returned to the users. If the results are adequate, the search is over.

3 The *SemGsearch* Methodology

To achieve a semantic search, it is necessary to process geospatial data sets at a conceptual level. However, a geographical object can be described in many ways depending on the degree of knowledge, abstraction, and interpretation desired. The conceptualization of a domain is used to integrate information. In the geographic domain, we propose to apply the GEONTO-MET approach (Torres et al. 2011) in order to build geographic ontologies.

Additionally, we propose the Conceptual Distance (DIS-C) algorithm to compute the distance between concepts defined in the retrieval process when the user performs a query. This algorithm avoids obtaining empty answers when the exact concept is not found in the sources. Therefore, as a response, users will obtain a concept that is conceptually closer. The conceptual structure and stages of the *SemGsearch* approach are shown in Fig. 1.

- **Conceptualization**. This stage consists of building the knowledge base, represented directly by an ontology. This conceptual structure stores information related to the geographic domain. It also computes the conceptual distance table as a basis for the custom query.
- **Synthesis.** This stage involves the instantiation of objects that represent geographical concepts of the domain by using the metadata that describe each data source. In other words, the stage has the function of populating the ontology with geographic objects located within several data sources that compose the semantic repository.
- **Analysis**. At this stage, the repository is used for semantically searching geographic objects on domains, using the concepts related to the query request.

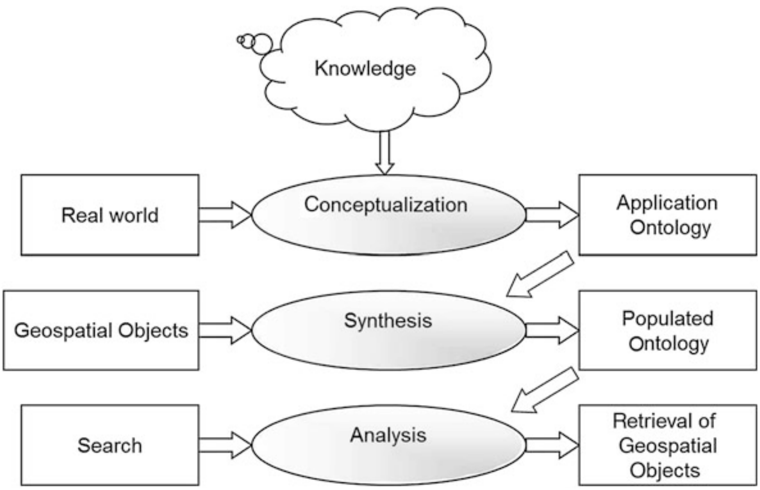


Fig. 1 Conceptual structure of *SemGsearch*

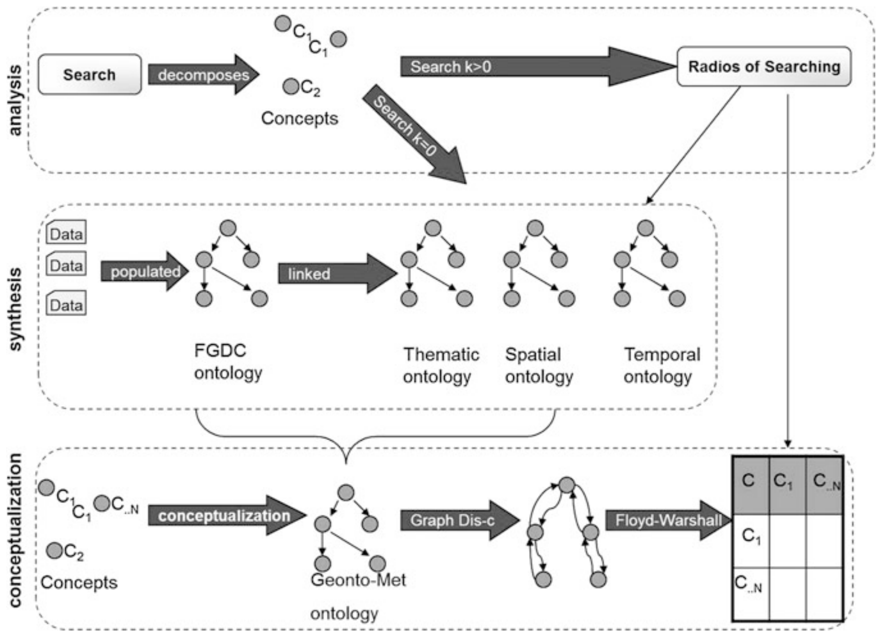


Fig. 2 Conceptual framework of *SemGsearch*. FGDC, Federal Geographic Data Committee

The conceptual framework of the three stages is depicted in Fig. 2. The conceptualization stage uses the application ontology that contains the knowledge base of the domains described in the conceptualization. By applying a conceptual similarity algorithm, we build a graph that determines a distance value to each conceptual relationship from the ontology built by GEONTO-MET. Finally, the Floyd–Warshall algorithm, described in (Floyd 1962), is used to determine the smallest value or cost of each concept or node to another, indicating a similarity value. This algorithm's performance is reduced with large number of nodes because it works with matrices. The synthesis stage selects the geospatial data from the Federal Geographic Data Committee (FGDC) (2012). This specification defines (by means of metadata descriptions), the main features of a geographic object to be retrieved; they are used in the population task for reading the metadata file, which describes a geographic object. Later, an instance for each geographic object is created. After this process, each of these objects is linked to the other three domains—thematic, spatial, and temporal—by means of their metadata keywords in order to semantically retrieve geospatial data in the next stage.

In the analysis stage, the searching is performed by decomposition from the query to concepts. Additionally, the vector created by the concepts will be used for searching concepts within the ontology together with the geographic object linked in the synthesis stage. Moreover, the retrieval process is based on the expansion of the conceptual distance among concepts, increasing the radius. The first approximation is to find equal or exact concepts for the conceptual distance $k = 0$. Later, each domain $k > 0$ is extended from the resulting Floyd–Warshall table in the conceptualization.

3.1 Conceptualization Stage

In this stage, the domains that define geographic objects are conceptualized. It is possible to use a variable number of domains; however, in this work, we only used three domains: thematic, spatial, and temporal, which were conceptualized using the GEONTO-MET methodology. If the number of domains is increased, the semantic granularity of the ontology is improved and the retrieval process is enriched. This fact allows a query to be refined and thus become even more specialized.

3.1.1 *DIS-C: Conceptual Similarity Algorithm*

The conceptual distance is defined by the space that separates two concepts within a specific conceptualization (Abu-Hanna and Jansweijer 1994). The main idea of the algorithm is to establish the distance value of each relationship within the ontology and translate it into a weighted directed graph, where each concept becomes a node and each relationship becomes a pair of edges.

The classic step for computing the minimum distances is to find the distance among concepts that are not directly related. Thus, the steps to compute the conceptual distance are the follows:

1. For each type of relationship, it is necessary to assign a conceptual distance for each relationship within the conceptualization. For example, if it has the relation "is" and "road is a communication route," we can set the distance of "road" to "communication route" and "communication route" to "road." As an illustration, we can establish that *distance(road, communication_route)* $= 1$ and *distance(communication_route, road)* $= 0$. Formally, $K(C, \Re, R)$ is a triplet that defines a conceptualization, where C is the set of concepts, $\Re$ is the set of relationships types, and R is the set of relations in the conceptualization. For each type of relationship $\rho \in \Re$, it has to set the values of δ^{ρ} to the relationship ρ in the normal sense, and $\overline{\delta}^{\rho}$ to the relationship ρ in reverse sense. In the example above, $\delta^{is} = 0$ and $\overline{\delta}^{is} = 1$.
2. The directed graph is created as $G_K(V, A)$ for the conceptualization K. First, it is added to each item $c \in C$ as a vertex in the graph G_K, i.e., $V = C$. Now, for each relationship $a\rho b \in R$, where $a, b \in C$, edges (a, b, δ^{ρ}) and $(b, a, \overline{\delta}^{\rho})$ are added.

3. Once the graph has been built, we apply the Floyd–Warshall algorithm to process the minimal distance between two nodes. The table of minimum distances between each pair of vertices, which are directly mapped to the concepts in the ontology, is created. Therefore, the result is the conceptual distance disseminated to all concepts in the conceptualization *K*.

3.1.2 *Application of Conceptual Similarity in GEONTO-MET*

GEONTO-MET (Santos et al. 2003) is composed of three axiomatic relations: "is," "has," and "does." The relation "is," as we mentioned in the example "road is a communication route," establishes the distance of "road" to "communication route" and "communication route" to "road." It is defined by *distance(road, communication_route) = 1* and *distance(communication_route, road) = 0*. Thus, we propose that if $a(is)b \in R_R$, then $\delta^{is}(a,b) = 0$ and $\overline{\delta}^{is}(a,b) = 1$. The relation "has" defines properties in which the distance is inversely proportional to the number of occurrences of the concept that "has" presents. For example, if an urban area "has" a street of the first order, then the conceptual distance between the concept "urban area" and the term "street" will be inversely proportional to the number of streets that the urban area presents. That is, if $a(has)b \in R_R$, then $\delta^{has}(a,b) = \frac{1}{R(p)}$, where $R(p)$ is the occurrence number of the property $p = a(has)b$ in R_R (this value is normally 1). On the other hand, the conceptual distance of "street" to "urban area" will also be inversely proportional to the number of "streets"; otherwise, "urban area" is directly proportional to the number of properties that "urban area" contains (streets, buildings, parks, etc.) If $a(has)b \in R_R$, then $\overline{\delta}^{has}(a,b) = \frac{card(P(a))}{R(p)}$, where $P(a) = \{x | a(has)x \in R_R\}$ for any concept $x \in C$ and $R(p)$ is the occurrence number of the property $p = a(has)b$ in R_R.

The relation "does" defines abilities. The conceptual distance is computed in both senses of the relationship, inversely proportional to the number of times that one ability is referred to by one concept. Likewise, in the inverse relationship, it is directly proportional to the total number of abilities that a concept contains.

For instance, we consider the ontology depicted in Fig. 3, in which the blue arrows indicate the relation "is" and the red arrows indicate the relationship "has." The conceptual distance is defined between the pairs of concepts by the DIS-C algorithm. As a result, a weighted directed graph is created (see Fig. 4). Once the graph is generated, it is necessary to compute the shortest distance $\Delta_C(a,b)$.

Once the graph with the weights corresponding to the conceptual distance is obtained, the next step is to transform it in a matrix. Later, it is necessary to measure the axiomatic relations. To illustrate the procedure, we have in the example above the (*concept 1*, *concept 2*) related under the relation "is,"

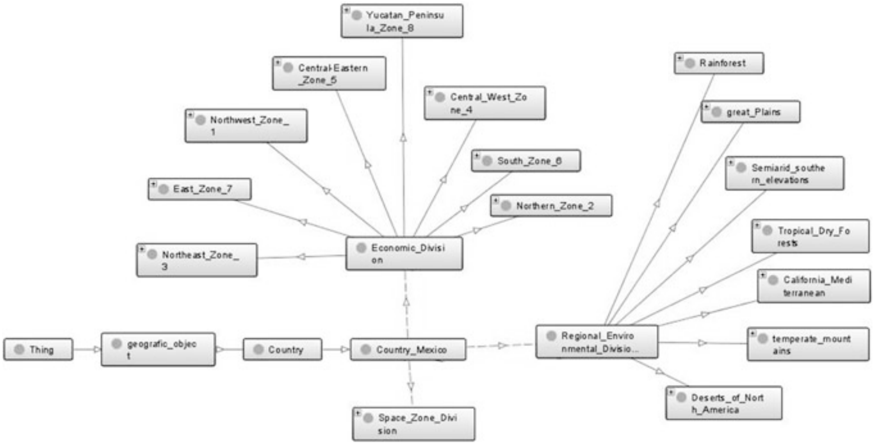


Fig. 3 Geospatial ontology for the division of a country (Mexico)

according to DIS-C, the "concept 1" to "concept 2," the conceptual distance or weight $w = 1$ and 0, conversely.

In the case of (*concept 2*, *concept 4*) related under the relation "has" the "concept 1" to "concept 2", the conceptual distance w is the weight $\frac{1}{R(a \rho b)}$, that is, it has 1/1 and conversely $\frac{card\,(P(a))}{R(a \rho b)}$, is 1/1. Finally, the concepts (*concept 2*, *concept*

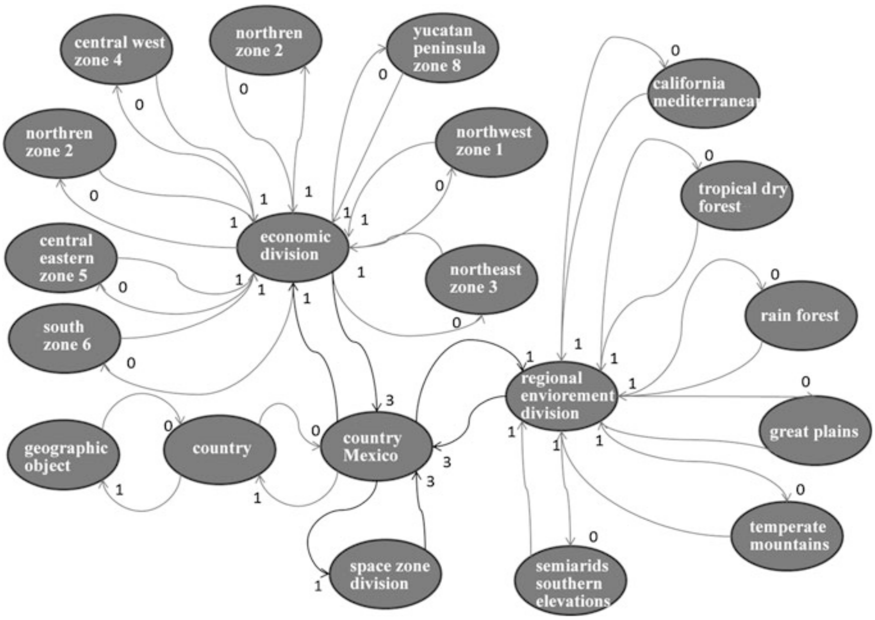


Fig. 4 Weighted directed graph obtained from applying the DIS-C algorithm

Table 1 Matrix generated by Floyd–Warshall algorithm using the DIS-C graph, indicating the shortest path

Concepts	C1	C2	C3	C4	C5
C1	0	0	1	1	1
C2	1	0	1	1	1
C3	3	2	0	3	3
C4	3	2	3	0	3
C5	2	1	2	2	0

3) under the relation "does" the weight w of "concept 1" to "concept 2" is $\frac{1}{R(a\rho b)}$, it has 1/1 and conversely $\frac{card\,(P(a))}{R(a\rho b)}$ is 2/1. The next step is to determine the lowest weight w between two concepts a and b belonging to a given domain ontology. The process is carried out using the Floyd–Warshall algorithm, which considers as input the resulting graph of the algorithm as a DIS-C matrix between concept weights w, C_x, C_y another belonging to the same domain indicating the shortest path or minimum weight between any pair of nodes C_x, C_y that belong to the graph DIS-C. The matrix that indicates the minimum weight of a concept C_x to a concept C_y is described in Table 1. In addition, we present the DIS-C algorithm in Table 2.

3.2 *Stage of Synthesis*

The synthesis stage generates instances from the concepts described in the ontology (previous stage). The task is performed for each site or server where there are data sources. It is described as follows:

1. For each data source located in the repository, the description of geographic objects is obtained using the metadata file that accompanies each geographic object; it is mapped on FGDC domain (see Fig. 5).
2. For each domain of metadata, the set of keywords is identified. These keywords are mapped with concepts in the others domains (see Fig. 6).

Table 2 The DIS-C algorithm

1. $G_O \leftarrow$ weighted directed graph
2. $R_R \leftarrow$ relations in O
3. For all $a\rho b\pi c \in R_R$ do
4. If $\rho = is$ then
5. Add $(a, b, 0)$ to G_O
6. Add $(b, a, 1)$ to G_O
7. Else if $\rho = has$ then
8. Add $(a, b, \frac{1}{R(a\rho b)})$ to G_O
9. Add $(b, a, \frac{card\,(P(a))}{R(a\rho b)})$ to G_O
10. Else if $\rho = does$ then
11. Add $(a, b, \frac{1}{R(a\rho b)})$ to G_O
12. Add $(b, a, \frac{card\,(H(a))}{R(a\rho b)})$ to G_O
13. End if
14. End for
15. Return Shortest path (Go)

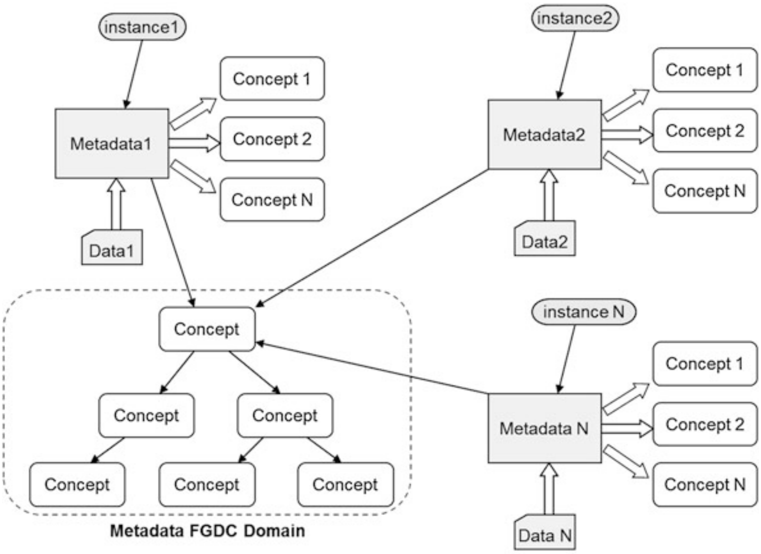


Fig. 5 Instances of geospatial data in the Federal Geographic Data Committee domain

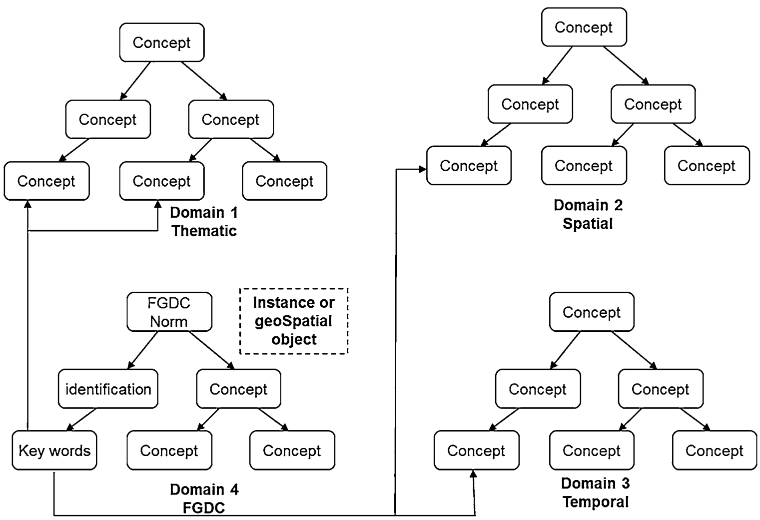


Fig. 6 Geographic objects linked in the ontology. FGDC, Federal Geographic Data Committee

3.3 Stage of Analysis

In the analysis stage, the tasks of semantic searching and retrieval as well as the results display are carried out. The stage is composed of five basic processes:

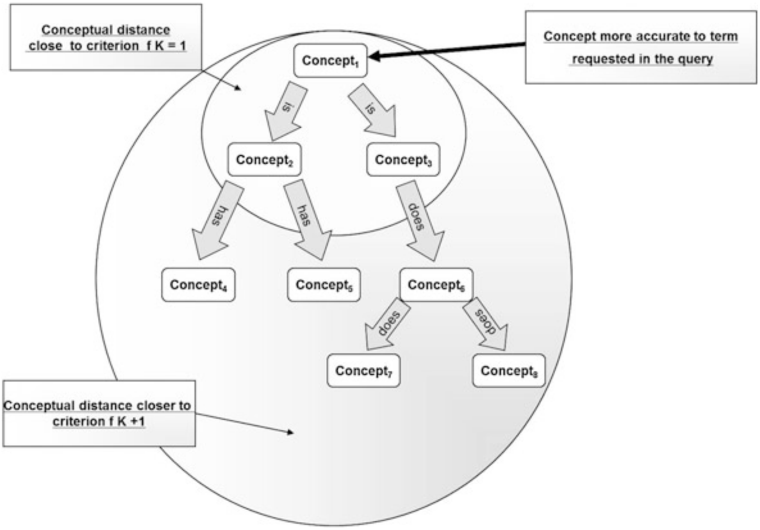


Fig. 7 Conceptual distance extending the searching range

1. The concepts and their domains are obtained from the query; after that, a vector that defines the geographic objects is created.
2. With the previous resultant vector, the next step is to determine each concept to its respective domain.
3. If the concept in the query has an exact match with a concept in the domain, this is the closest conceptual distance, i.e., $k = 0$. The process continues, extending for $k \geq 0$ (see Fig. 7).
4. When the radius are increased to values $k \geq 0$, more concepts are obtained; thus, several instances of geospatial objects in the data sources are retrieved. This procedure is applied to each domain, then a list of concepts that are semantically related is created by means of the intersection among domains (see Fig. 8).
5. The last task in the search is to return the entire set of instances from the previous procedures for $k = 0$ that represents an exact match and, after that, for values $k > 0$ that are semantically close (see Fig. 9).

4 Results

As a case study, the *thematic domain* is composed of urban areas, communication infrastructure, and forest resources using the specification from INEGI (Mexican National Institute of Statistics, Geography, and Informatics 2012). The *spatial domain* refers to Mexico with different classifications: regional division by CONABIO (Mexican National Commission for Knowledge and Use of

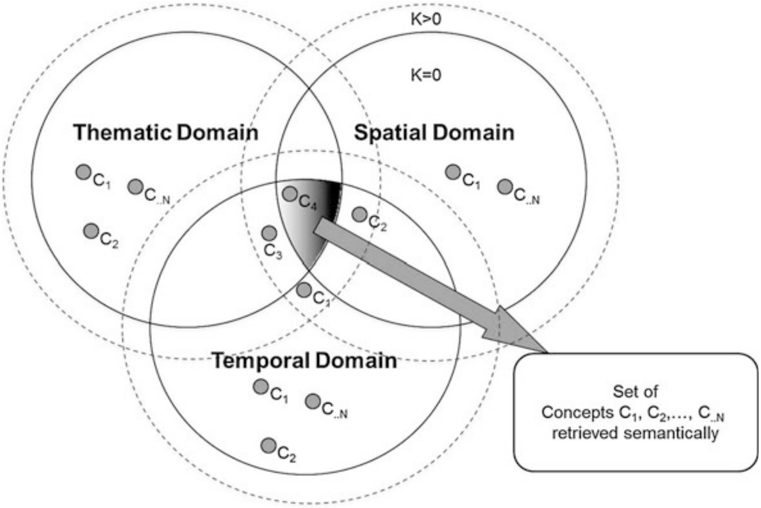


Fig. 8 Semantic intersection among the sets of geographic concepts

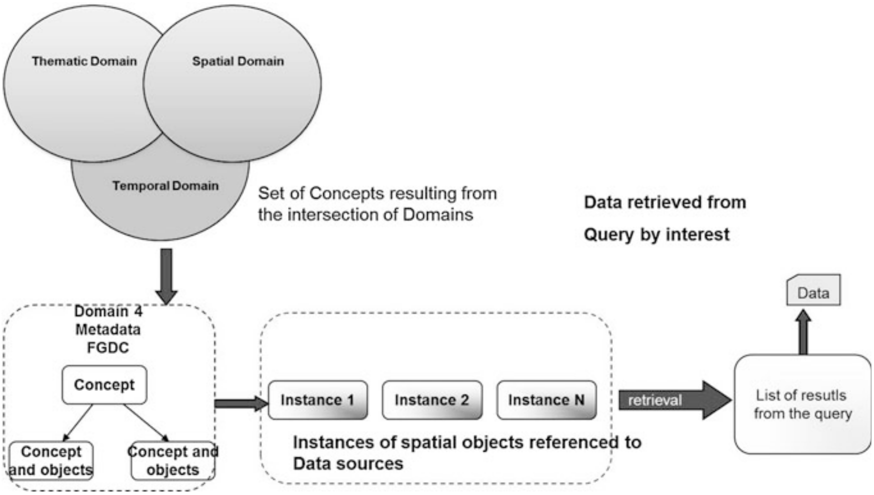


Fig. 9 Instances retrieved from the repository. FGDC, Federal Geographic Data Committee

Biodiversity 2012), spatial distribution area by CFE (Mexican Federal Electricity Commission 2012) and economical area as in the classification from INEGI. Finally, *the temporal domain* is conceptualized taking into account the Gregorian calendar.

The semantic repository is structured on *SemGsearch* by means of an administrator interface, which adds servers and loads domains that are built under GEONTO-MET in the OWL format (McGuinness et al. 2004). The system allows the

Fig. 10 *SemGsearch* administration interface

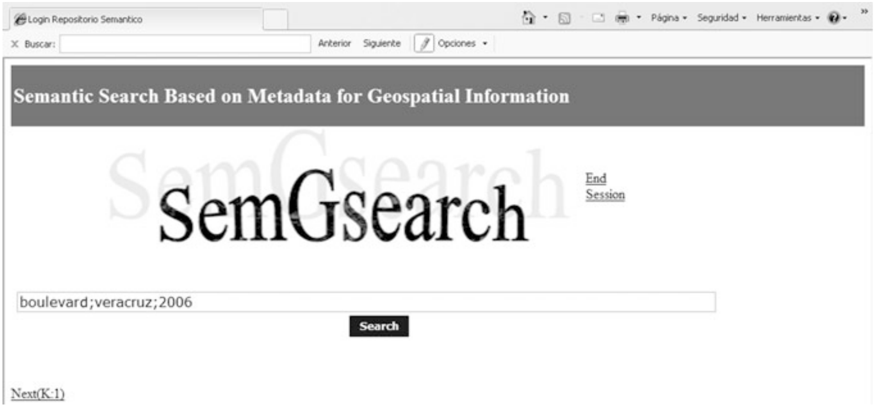


Fig. 11 Semantic search: thematic "boulevard," spatial "Veracruz, Mexico," year "2006"; $k = 0$

possibility of querying the metadata files of the geographic objects located in the servers. The *SemGsearch* administrator is depicted in Fig. 10.

As an example, the next query retrieves the geospatial object "boulevard" in the location of "Veracruz state, Mexico" (see Fig. 11). Because "boulevard" at "Veracruz" is not stored on any server, this query will not retrieve any instance with an exact match ($k = 0$). In this case, to avoid empty answers, the conceptual distance is applied in order to retrieve similar concepts according to the conceptualization.

Now, for a value $k = 1$, the semantic search is increased in the partition of the ontology, according to the conceptualization, within the spatial domain "Eco-

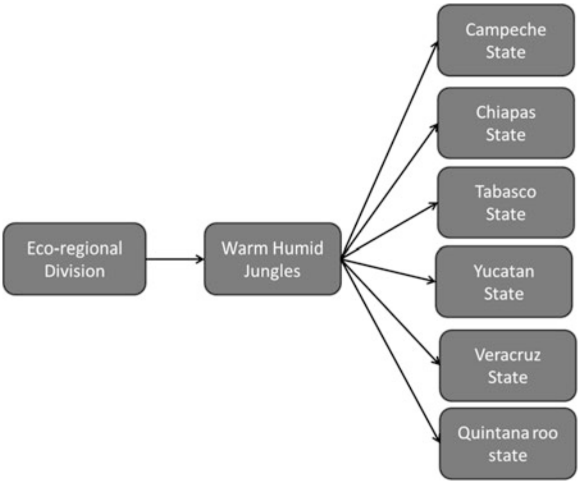


Fig. 12 Ontology partition of the class "Eco-regional Division" and "Warm Humid Jungles"

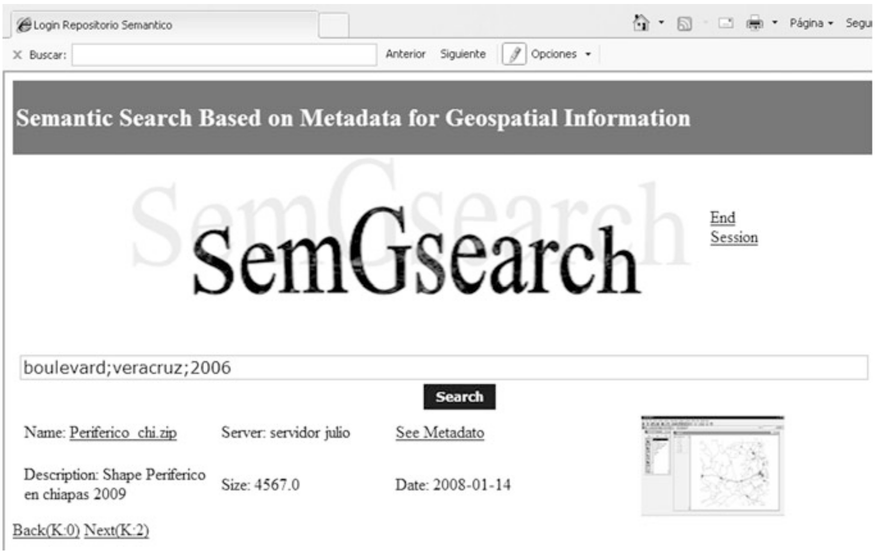


Fig. 13 Semantic search: "boulevard," "Veracruz," "2006" with $k = 1$

regional Division." There is a similarity under the classification of "Warm Humid Jungles": the concepts "Chiapas state" and "Veracruz State" are the closest states semantically with respect to the others (see Fig. 12).

According to the increment of the conceptual distance for $k > 0$, the next search is made as "Boulevard" that is located in "Veracruz," this geospatial object is

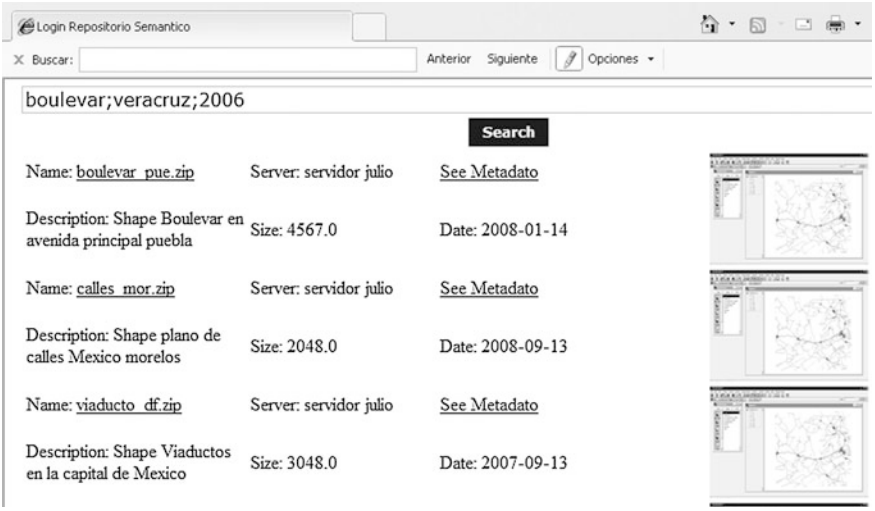


Fig. 14 Semantic search: "boulevard," "Veracruz," "2006" with $k = 2$

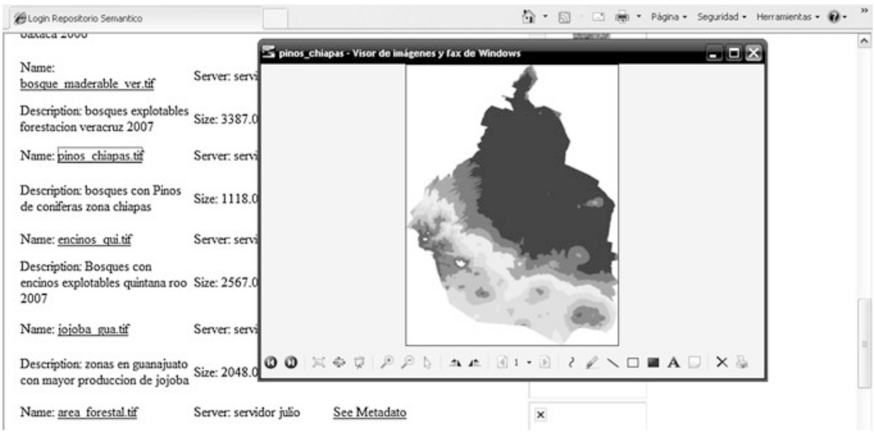


Fig. 15 Snapshot of a geographic object

already located in one server in the repository. The results that accomplish the request query by means of semantic similarity are depicted in Figs. 13 and 14 (in this case, for "boulevard" in "Veracruz" with $k = 1$ and $k = 2$, respectively).

In the retrieval process, the geographic objects can be downloaded directly or viewed by means of a snapshot, as depicted in Fig. 15

5 Conclusions

In this chapter, a methodology focused on the tasks of semantic integration and retrieval of heterogeneous data sources, which may be located on different servers, is proposed. Similarly, the main goal of this chapter is present a method for sharing geospatial information by means of semantic retrieval mechanism using a conceptual representation. In this approach, a mechanism to measure the conceptual distance (the DIS-C algorithm) is proposed. This algorithm measures how conceptually close two concepts are. It allows the system to resolve queries according to k values, which represent a conceptual distance. This method avoids returning empty requests to the users.

The present work leads to an easier and faster way of searching for geographic objects; it also aids in solving multiple conceptualizations that integrate different domains. In this proposal, it is not important to know the user's degree of expertise because the approach integrates and conceptualizes the domains by describing the metadata. As a case study, we implemented the *SemGsearch* system for testing the proposed approach. This application provides a ranked list of the geographic objects that were semantically retrieved.

Future work is oriented toward applying this methodology in particular cases of the semantic web. The conceptual distance is an important issue that covers semantic information integration and semantic interoperability among applications and information in the web.

Acknowledgments This work was partially sponsored by the National Polytechnic Institute (IPN), the National Council for Science and Technology (CONACyT) under grant 106692, and Research and Postgraduate Secretary (SIP) under grants 20120563 and 20121661.

References

Abu-Hanna A, Jansweijer WNH (1994) Modeling domain knowledge using explicit conceptualization. IEEE Expert 9(2):53–64
Amor D (2000) The e-business (r) evolution: living and working in an interconnected world. Prentice Hall PTR Upper Saddle River, NJ
Bishr Y (1998) Overcoming the semantic and other barriers to GIS interoperability. Int J Geogr Inf Sci 12(4):299–314
Boley H et al (2010) Integrating positional and slotted knowledge on the semantic web. J Emerg Technol Web Intell 2(4):343–353
Buccella A et al (2007) Integration of geographic information systems. In: IX Workshop of Researchers in Computer Science, pp 397
CFE (Mexican Federal Electricity Commission). http://www.cfe.gob.mx/. Accessed 20 June 2012
CONABIO (Mexican National Commission for Knowledge and Use of Biodiversity). http://www.conabio.gob.mx/. Accessed 20 June 2012
FGDC (Federal Geographic Data Committee), http://www.fgdc.gov/. Accessed 20 June 2012
Floyd RW (1962) Algorithm 97: shortest path. Commun ACM 5(6):345
Heywood I et al (2006) Geographical information systems. Pearson Education Limited

INEGI (Mexican National Institute of Statistics, Geography and Informatics) (2012). http://www.inegi.org.mx/. Accessed 20 June 2012
Janowicz K (2007) Similarity-based retrieval for geospatial semantic web services specified using the web service modeling language (wsml-core). In: Scharl A, Tochtermann K (eds). The geospatial web-how geo-browsers, social software and the web, vol 2. Springer, London, pp 235–246
Kiryakov A et al (2004) Semantic annotation, indexing, and retrieval. Web Semant Sci Serv Agents World Wide Web 2(1):49–79
Liang Y et al (2007) Hybrid ontology integration for distributed system. In Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing. SNPD 2007, 1(1):309–314
MacEachren AM (2004) How maps work: representation, visualization, and design. The Guilford Press, NY
Malik SK et al (2010) Semantic annotation framework for intelligent information retrieval using KIM architecture. Int J Web Seman Technol 1(4):12–26
McGuinness DL et al (2004) OWL web ontology language overview. W3C recommendation, 2004–2003 (10). http://www.w3.org/TR/owl-features/. Accessed 20 June 2012
Sabbouh M et al (2007) Web mashup scripting language. In: Proceedings of the 16th International Conference on World Wide Web, ACM, pp 1305–1306
Santos JM et al (2003) Fuentes, tratamiento y representación de la información geográfica, "Sources, treatment and representation of the geographic information". Universidad de educación a distancia UNED, Madrid, p 421
Torres M et al (2011) GEONTO-MET: an approach to conceptualizing the geographic domain. Int J Geogr Inf Sci 25(10):1633–1657
Vilches LM et al (2009) Towards a semantic harmonization of geographic information. Treballs de la Societat Catalana de Geografía, pp 727–736
Zhan Q, Zhang X, Lic D (2008) Ontology-based semantic description model for discovery and retrieval of geospatial information. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol 32

A Semantic Model to Query Spatial–Temporal Data

Benjamin Harbelot, Helbert Arenas, and Christophe Cruz

Abstract There is a growing need for the study of spatial–temporal objects and their relationships. A common approach for this task is the use of relational databases, which unfortunately do not allow inference. In this research, we introduce a new approach that uses the concept of a "continuum" together with ontologies and semantic Web technologies. The continuum allows us to define parent–child relationships between representations of objects. It also allows us to compare the evolution of two different objects and establish the relationships between them along time. Our approach is based on the four-dimensional fluent, which is extended to obtain spatial–temporal qualitative information from the analysis of objects and their relationships. The results of our analysis are later added to our knowledge base, thus enhancing it. Our preliminary results are promising and we plan to further develop the model in the near future.

Keywords GIS · Semantics · Spatio–temporal

1 Introduction

There is a growing demand for tools to handle spatial–temporal information; however, currently there is a limited number of available options. This research introduces a novel approach that adds semantics to spatial–temporal data, allowing

B. Harbelot · H. Arenas · C. Cruz (✉)
Laboratoire Le2i, UMR-6302 CNRS, Departement Informatique Dijon, University of Burgundy, 7 Boulevard Docteur Petitjean, Dijon CEDEX, BP 17867 21078, France
e-mail: christophe.cruz@u-bourgogne.fr
URL: http://checksem.u-bourgogne.fr

B. Harbelot
e-mail: benjamin.haberlot@checksem.fr

H. Arenas
e-mail: helbert.arenas@checksem.fr

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_5,

reasoning and inference. When designing a spatio–temporal information system, it is necessary to deal with three aspects of data: (1) the spatial aspect, which consists of geometry; (2) the temporal aspect, which defines the interval of existence of the geometries; and finally (3) the semantic aspect of an object, which aims to provide a meaning beyond the purely geographic (Yuan 1999). There is a fourth optional feature, which consists of the representation of the semantics derived from the evolution of the various attributes of the objects. For instance, in the study of land parcels, we found that each one has identification, geometry, and possibly other attributes, such as land use, which might evolve over time. By considering the evolution of the geometry and attribute values, we would be able to handle the semantics resulting from changes in the objects and in the relationships among them over time.

Currently, among geographic information system (GIS) tools, there is a lack of suitable options when there is need to perform spatial–temporal analysis. There are previous examples in the field of RDBMS for storage of dynamic spatial objects. However, RDBMS do not implement inference or reasoning mechanisms. Ontologies, on the other hand, allow these mechanisms by creating formal representations of concepts, properties, and relationships between concepts. Traditionally, ontologies have been applied to static domains, in the sense that entities represented in these ontologies do not change over time or space. However, spatial object's are dynamic; it is possible that an object changes attributes and spatial representation over time. Objects with a spatial representation might grow, shrink, change their shape, split, disappear, or merge into a new object. To overcome this limitation, in this paper we propose a model that develops the concept of continuum within an ontology. The continuum is an abstract concept that represents the evolution of an object over time from spatial and semantic points of view. The continuum allows the representation of objects with known semantics and spatial representations with a defined lifetime. Using our model, we can register the object's evolution and link an object with its spatial representation that corresponds to any given point of time. The continuum model allows the handling of spatial–temporal dynamics and stores the process in the form of spatial graphs (Fig. 1).

We can distinguish two approaches to handle the semantics of the spatial–temporal objects: (1) adding semantic capabilities to GIS tools, which are not necessarily ontologies and (2) inserting spatial data into an ontology. The first approach provides only an extension of the attributes, whereas the second allows for an enhanced description of the knowledge that can be used to perform reasoning on spatial data. Our work is based on the second approach.

Several methodologies for the dynamic representation of objects and their properties have already been proposed. Among the most well known are the temporal description logic, temporal RDF, versioning, reification, N-ary relationships, and the four-dimensional (4D)-fluent approach.

The 4D-fluent approach can effectively represent temporal information in OWL. Other works such as SOWL are based on this approach, extending OWL and allowing it to handle not only quantitative temporal information but also qualitative temporal information. Qualitative temporal information, such as

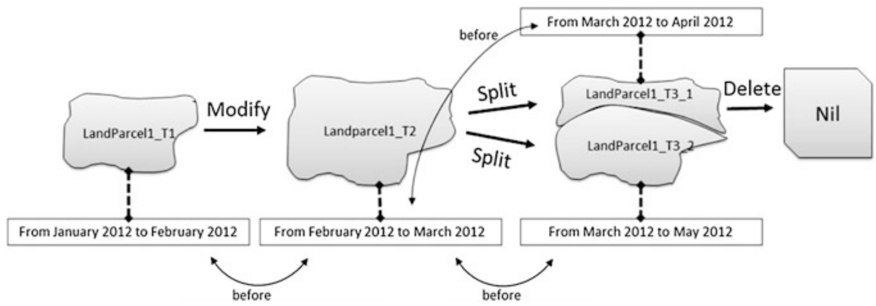


Fig. 1 Evolution of a land parcel over time

"before" or "after," has the advantage of representing relationships between time intervals with unknown starting or ending points. Qualitative information can be inferred from quantitative data. Qualitative information can also be used as an alternative if there is missing quantitative information.

The inference of qualitative information from quantitative information, both spatial and temporal, is the core of this work. To accomplish this goal, a set of SWRL rules are defined and used to enrich the ontology with inferred facts. Additionally, SWRL provides a powerful extension that allows the definition of its own methods. These methods are called built-ins. In our work, we have developed built-ins that perform calculations between space and temporal representations. The result of the operations is included in our ontology to enrich it. Besides inference capabilities, a system of this kind must also provide mechanisms to query the knowledge base. Traditionally, the query language used for ontologies has been SPARQL. However, because it can only be used with RDF-based ontologies, it has limited effectiveness. An alternative to SPARQL is SQWRL, which retains the semantics of OWL. SQWRL syntax is similar to SQL, providing an easy and explicit way to query the system.

Work on knowledge representation is presented in Sect. 2. In Sect. 3, we present the proposed model and the inference system implemented. Finally, in Sect. 4 we present the query language chosen for this work.

2 Background

The most common way to represent spatially dynamic objects is through the use of a relational DBMS. A very interesting example is DOMINO which uses a combination of ArcView and Informix as an application server to store spatial data and ArcIMS as a render tool. Another interesting example is SECONDO, a prototype of a DBMS that is able to handle moving objects. SECONDO is able to store the history of movements of a given object and respond to spatio-temporal queries

(Secondo 2012, Innerebner 2007). However, one limitation of these examples is that although they are able to represent dynamic objects, they are not able to perform any inference or reasoning with the stored information. On the other hand, we have the semantic Web technologies that have been designed expressly to represent knowledge in a form that can allow the use of reasoning and inference mechanisms. Several representation languages have been defined in the semantic Web. In this section, we discuss the existing data management technologies in a scalable ontology.

2.1 Representing Spatial Dynamics in an Ontology

The two main philosophical theories concerning object persistence over time are *endurantism* and *perdurantism*. The first one, *endurantism*, considers objects as three-dimensional entities that exist wholly at any given point of their life. On the other hand, *perdurantism*, also known as the four-dimensional view, considers that objects have temporal parts, ("timeslices") that compose their temporal dimension (Welty 2006). This approach represents the different properties of an object over time as fluent. A fluent property is valid only during certain intervals or moments in time. In this research, we use a *perdurantism* approach to record the changes and movement of spatial objects over time. In order to apply this approach within an ontology, it is necessary to convert static properties into dynamic ones. In the field of conceptual modeling, this is expressed in the concept-class association defined between two entities. OWL has limitations for the representation of dynamic entities; it only allows binary relationships between individuals. In order to solve this limitation, previous research has proposed several alternatives, such as temporal RDF, reification, versioning, or 4D-fluent.

Temporal RDF proposes an extension of the standard RDF for naming properties with the corresponding time interval, allowing explicit management of time (Gutierrez et al. 2007). Reification is a technique used to represent n-ary relations with languages such as OWL, which allow only binary relations (Hayes 2004). Versioning is described as the ability to handle changes in ontologies by creating and managing multiple variants of them (Klein and Fensel 2001). However, these methods have some disadvantages. Temporal RDF relies solely on RDF triples; therefore, it does not have all the expressiveness of OWL. For example, when using only RDF, it is not possible to express qualitative relationships. Reification allows the use of a triplet as object or subject of a property. But this method has also its limitations; for example, the transformation from a static property into a dynamic one increases substantially the complexity of the ontology, reducing the querying and inference capabilities. Additionally, reification is prone to redundant objects, which reduces its effectiveness. The major drawback of versioning is the redundancy generated by the slightest change of an attribute. Any information requests must be performed on multiple versions of the ontology, reducing its effectiveness.

The 4D-fluent approach is based on the *perdurantism* philosophical approach. It considers that objects have a temporal dimension composed by several temporal representations, each corresponding to a defined interval of time. Together, all of these representations compose the temporal dimension of an object. In the literature, 4D-fluent is the most well-known method to handle dynamic properties in an ontology. It has a simple structure, allowing it to easily transform a static ontology into a dynamic one. However, using this approach, it is not possible to handle explicit semantics. This fact causes two problems: (1) it is difficult to maintain a close relationship between geometry and semantics; and (2) it increases the complexity for querying the temporal dynamics and understanding the modeled knowledge. Furthermore, this approach does not define qualitative relationships to describe the type of change that has occurred or to describe the temporal relationships between objects. We cannot then know which objects have undergone a change and what objects might be the result of that change.

The spatial evolution of an object involves movement or a change of shape (Brisaboa et al. 1998). In the case of a movement, it is easy to identify and locate the object before and after the event. However, when an object suffers a succession of changes, a key question arises: how much can an object change before its semantics are modified? And if there is a semantic change, then how do we know that this is the same object at different times? The 4D-fluent approach does not allow an object to change its nature; it only allows the change of the value of some of its properties. However, the semantics associated with a geometry may change. For example, a land parcel may change from being forest into being an urban parcel. In this example, the geometry has not changed; however, there is a semantic change. Conversely, the semantics might not change as the geometry evolves. For instance, a given urban land parcel might expand by purchasing neighboring parcels. In the first example (see Fig. 2a), there are two different semantic objects associated with the same geometry at different times. In the second example (see Fig. 2b), we have two related geometries for the same semantic object at different times.

2.2 Representation of Semantic Relations

To study the evolution of a concept over time, we require tools to define temporal concepts in an ontology. The OWL-Time ontology is used for this purpose to describe the temporal aspects of the content of web pages and properties of web services. Moreover, this ontology provides good support for expressing topological relationships between times and time intervals, as well as times and dates (Hobbs and Pan 2012). Another tool designed to handle temporal concepts in an ontology is SWRL Temporal Ontology. This tool has all the capabilities of OWL-Time ontology and additionally has the advantage of built-ins, which can be used to perform calculations between intervals and time instants to infer relationships from qualitative and quantitative information (O’Connor and Das 2010). The qualitative

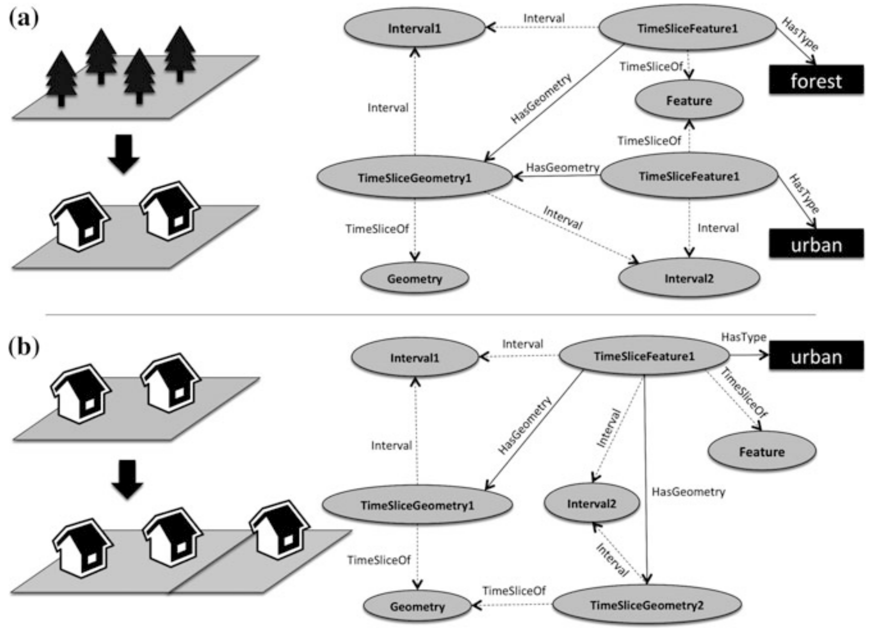


Fig. 2 Examples of the evolution. **a** Two different semantic objects for the same geometry. **b** Two related geometries for the same semantic object

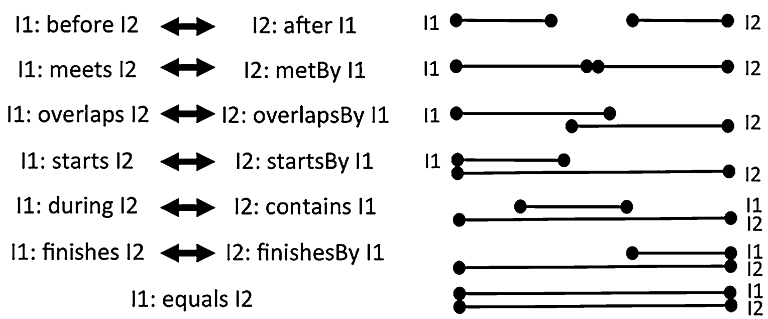


Fig. 3 Allen temporal relations

relations in the time domain are based on binary and mutually exclusive relations as proposed by Allen (1983) (Fig. 3).

The addition of Allen relations can increase the expressive power of the system by adding qualitative information in addition to the quantitative data. Allen relationships allow one to go even further when intervals are semi-closed (just a defined start date or just a defined end date) (Batsakis and Petrakis 2010). For example, let us suppose we have three intervals: I1, I2, and I3. We know that I1 meets I2 and that I2 contains I3, but we do not know the ending point of I2 or the

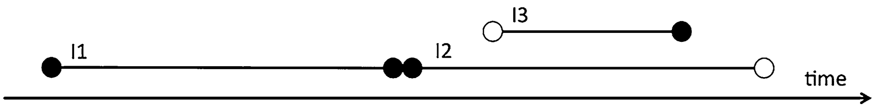


Fig. 4 Using Allen temporal relations to infer new knowledge

starting point of I3. However, then we can infer that because I2 contains I3, I3 must be after I1, even if the information about starting and ending points is incomplete (see Fig. 4). Lack of knowledge caused by semi-closed intervals is largely filled by the integration of Allen relations to the model.

In GIS, objects are represented by points, lines, polygons, or other more complex figures based on these geometries. All these geometries are defined using the coordinates of points, which are quantitative information. The use of an ontology is of interest when one wants to study the relationships between these objects. There are mainly three types of relationships between geometries: directional, metric, and topological relationships. The relationships based on quantitative information can be translated into qualitative data (Brisaboa et al. 1998), in a similar fashion as we have described for the temporal aspect. By analyzing a set of moments and time intervals, it is possible to deduce qualitative topological relationships between objects. The topological analysis between two objects is done using the models: dimensionally extended nine-intersection model (DE-9IM) or RCC8 (Strobl 2008). In both cases, we obtain an equivalent set of topological relationships for specific regions. To calculate the spatial relationships between two geometries, the DE-9IM model takes into account the inside and outside of the contour of the geometries leading to the analysis of nine intersections, as depicted in Table 1.

Table 2 depicts the eight possible spatial relationships resulting from DE-9IM.

2.3 *Reasoning with Spatial–Temporal Information in an Ontology*

Only a reasoning system allows the switch from quantitative to qualitative data. In (Ryu and Ahn 2001), the authors introduce a model in which spatial–temporal information contained in a database and a spatial–temporal inference system work together. However, no information is given on the semantic Web technologies and only the Java language is quoted as a component of the inference engine; therefore, the universality and effectiveness of the inference system can be questioned. In Karmacharya et al. (2011), the authors propose a reasoning system that combines the topological calculus capabilities of a GIS and the inference capabilities of the semantic Web field. However, the notion of time is not incorporated into this model. Research conducted by Batsakis and Petrakis (2010) shows the model-based approach that uses SOWL 4D-fluent and ontology OWL-Time. A set of SWRL rules is established to allow the system to infer the topological, relational,

Table 1 DE-9IM Intersection Matrix

	dim(I(A) ∩ I(B))	dim(I(A) ∩ B(B))	dim(I(A) ∩ E(B))
DE—9IM(A, B) =	dim(B(A) ∩ I(B))	dim(B(A) ∩ B(B))	dim(B(A) ∩ E(B))
	dim(E(A) ∩ I(B))	dim(E(A) ∩ B(B))	dim(E(A) ∩ E(B))
where:	I = interior	B = Boundary	E = Exterior

Table 2 Topological predicates and their corresponding meanings

Topological	Predicate meaning
Equals	The geometries are topologically equal
Disjoint	Geometries have no point in common
Intersects	Geometries have at least one point in common (inverse of Disjoint)
Touches	Geometries have at least one boundary common point (no interior points)
Crosses	Geometries share some but not all interior points, and the dimension of the intersection is less than that of at least one of the geometries
Overlaps	Geometries share some but not all points in common, and the intersection has the same dimension as the geometries themselves
Within	Geometry A lies in the interior of geometry B
Contains	Geometry B lies in the interior of geometry A (the inverse of Within)

and metric relations. A customized language was developed by the authors for this model, which is similar in structure to SQL. The SOWL language allows simple spatial–temporal querying for both static and dynamic data.

2.4 Querying Information on a Spatial–Temporal Ontology

Traditionally, SPARQL has been the query language in ontologies. It is a World Wide Web Consortium recommendation that operates at the level of RDF graphs. However, using SPARQL, the queries become relatively complex for spatio-temporal information. An extension of this language, st-SPARQL, defines new functions that allow it to handle geometries but not temporal data (Koubarakis and Kyzirakos 2010). St-SPARQL is based on an extension of RDF called st-RDF, which integrates contact geometries and incorporates time in RDF. St-SPARQL and SPARQL are both based on RDF graphs; therefore, their inference capabilities are limited. Other research led to the development of SOWL, a language designed for querying models based on 4D time-fluent. SOWL is easy to use because its structure is similar to SQL. It provides specific spatio-temporal topology operators in addition to orientation and metric ones.

3 The Continuum Model

The continuum model represents objects with three distinct aspects as depicted in Fig. 5a:

Semantics: To identify an object and describe its associated knowledge

Space: The graphical representation of the object

Time: The interval or time instants that describe the temporal existence of the object

The continuum model aims to follow the evolution of dynamic objects. In the model, the spatial, temporal, and semantic aspects of the object are handled independently. Each change automatically creates a new object. If the change occurs only on the spatial part of the object, the newly created object will retain the same semantics as the original one, and vice versa if the change occurs only on the semantic part.

Each change adds to a genealogy of spatial–temporal objects. Some objects can then be defined as "parent" and/or "child" of other spatial–temporal objects. This genealogy enforces a coherency between the time intervals of each spatial–temporal object. Figure 5b depicts an example of objects genealogy. In this example, "object_2" is the result of a spatial change on "object_1," then "object_2" is child of "object_1"; it is also true that the time interval of "object_1" *meets* the time interval of "object_2." In addition, "object_3" is the child of "object_2," so we can infer that the time interval of "object_3" is *after* the time interval of "object_1." Coherence between time intervals and genealogy can be verified by the system. In Fig. 5b, the arrows between the objects represent the established offspring relationships generated by the evolution of a real-world object. It is possible to characterize the evolution of each object in the model according to the conceptual hierarchy depicted in Fig. 6.

A *continuum* is a set of representations of an entity along time. Each representation has a valid, finite time interval. The model links each representation to its context. A representation can belong to more than one continuum, then continuums can intersect. Figure 5b depicts the evolution of an entity and how the continuum concept is used to study it.

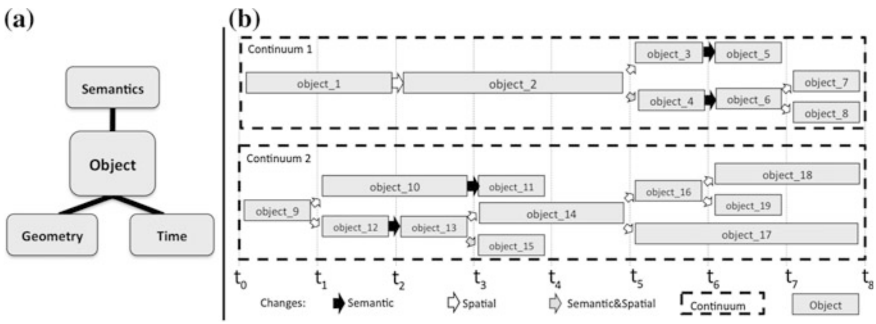


Fig. 5 **a** The three components of an object within the continuum model. **b** Using the continuum model to represent the evolution of an entity

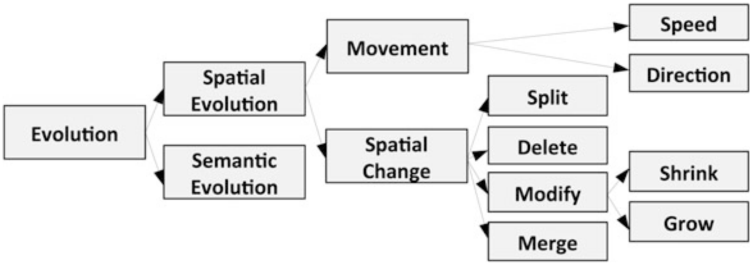


Fig. 6 Qualification of transition in the spatial graph

However, the relevance of the continuum model is not limited to this aspect. The objects represented in a continuum have a temporal part defined by a time interval and a spatial part defined by a geometry. It is possible to analyze the qualitative relationships between representations of objects of two different continuums.

3.1 Rules for an Appropriate Use of the Model

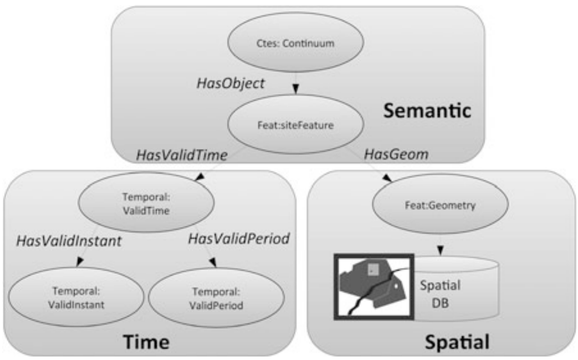
The continuum model aims to observe and store the spatial–temporal evolution of entities over time. A dynamic entity can undergo two types of evolutions: a change or a movement. To allow the observation and recording of the evolution of an entity, the model creates a new representation of the entity, each time it suffers a change. We represent dynamics in a shopping center to illustrate our point.

For this example, we have created a representation of a real shopping center in our spatial database. Later semantic and temporal information have been added and organized in the ontology. Figure 7 depicts the ontology class hierarchy. The main classes are "Continuum" and "siteFeature." The last one represents objects (semantically organized in class and subclass). The "Geometry" and "validTime" classes represent, respectively, the spatial and temporal parts of an object. The "spatialAnalysisResult" class is a special class which stores the results of "Union," "Difference," "Buffer," or "Intersection" spatial operators. More details about the implementation are depicted in Fig. 8.

A shopping center is composed of stores. Each store has an owner. Within the mall, some stores appear or disappear. A store might buy a neighboring property then merge with it and grow; others might split and give rise to new stores, or some shops might change their owners. In the mall, there are moving customers who enter and exit from the stores. The mall, and the stores that compose it, are considered as entities undergoing changes, while customers are seen as moving entities. Spatial–temporal moving objects, such as customers walking through the mall, are manageable in the continuum model. The movement of a client is represented by its recorded position at regular time intervals. Each recording has a spatial part corresponding to its representation in the geographic information system and a temporal part corresponding to the instant in which the position was

Fig. 7 Class hierarchy of the ontology (using protege plug-in OWL viz-Tab)

Fig. 8 Implementation of the Continuum Model in the ontology



recorded. The continuum then aggregates all recordings of a client hierarchically in time. The continuum model offers several options for managing the evolution of the mall. The user can decide to follow the evolution of the entity "mall," in which case it will be able to identify the emergence, disappearance, or renovations that involve mall sections. Alternatively, the user can follow the evolution of individual stores. It is also possible to combine both options and have a complete view of the mall evolution. Ultimately, for changing objects, the use of continuum depends on the goal the user has in mind. The continuum model offers different views of the same geographical region. The continuum model is very flexible and allows the handling of fixed and moving entities.

3.2 Reasoning on the Continuum Model

Reasoning capabilities are essential components of the application. They allow one to both check the consistency of information and also enrich the ontology automatically from data inferred through concepts, such as transitivity, symmetry, or inverse. The inference is done by introducing a set of SWRL rules. The use of SWRL is crucial because it provides a powerful extension that allows the definition of customized methods, called built-ins.

Spatial Built-ins:

In space, there are eight possible spatial relationships between two geometries. As explained above, the geometries are stored in a GIS system, and spatial analysis of these can be done in the GIS system. But the calculation of topological spatial relationships should be done in the ontology, more precisely with the SWRL rules. Spatial built-ins have been implemented to compute the topology between two geometries. The definition of these built-ins requires an interconnection between the ontology and GIS, which is done through a JAVA program. Spatial built-ins can be used with all concepts represented in the ontology that have a spatial component. The topological calculation requires access to the geometry store in the GIS. When a spatial built-in is used in SWRL rules, the JAVA program detects it and runs the calculation in the GIS.

Example: Give all the people within a given shop

$$\text{feat} : \text{shop}(?x) \wedge \text{feat} : \text{people}(?y) \wedge \text{spatialswrl} : \text{Within}(?x, ?y) \\ \rightarrow \text{sqwrl} : \text{select}(?x, ?y)$$

Temporal Built-ins:

On the time domain, the Allen relationships are defined between time intervals. Temporal built-ins do not require any calculation but only provide a qualitative result from a comparison between intervals or instants.

Example: Give all shops existing on the 1st January of 2012

$$\begin{aligned} \text{feat} : \text{shop}(?x) \wedge & \text{ temporal} : \text{hasValidTime}(?x, ?\text{time1}) \\ & \wedge \text{temporal} : \text{contains}(?\text{time}, {''}2012-01-01{''}) \\ & \rightarrow \text{sqwrl} : \text{select}(?x) \end{aligned}$$

Spatial–temporal Built-ins:

There are no specific spatial–temporal built-ins; however, the dynamic combination of spatial and temporal built-ins in a SWRL rule allows a spatial–temporal analysis between two objects. It may, for example, launch a topological analysis between spatial objects with only a common or disjoint time interval.

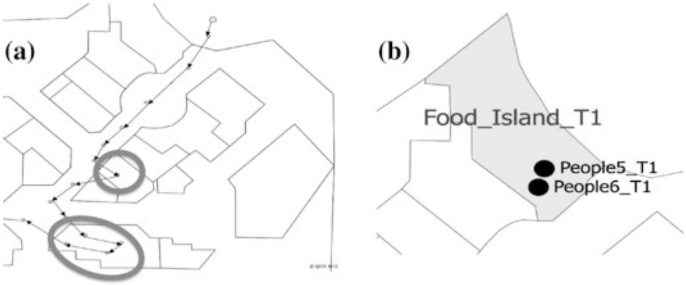


Fig. 9 Examples **a** and **b**

4 Querying on the Continuum Model

Apart from the SWRL built-ins explained above, there is a query language in SWRL, based on built-ins, called SQWRL (O'Connor and Das 2009). This is a language specially adapted to OWL that retains all the expressiveness and semantics of OWL, contrary to SPARQL. SQWRL is a concise language that is easily understood and semantically robust, thus making it the ideal candidate to query a system whose inferences capabilities are a major issue. Its explicit SQL-like structure allows a good understanding for users unfamiliar with the semantic Web domain.

Example A: Let us assume we have the positions of the customers for a certain mall. In this example, we desire to identify all people that have entered into a shop that existed in December 2011. We would also like to know the number of recorded positions for each person in each store (see Fig. 9a):

$$\begin{aligned}\text{feat : shop(?shop)} &\wedge \text{temporal : hasValidTime(?shop, ?time1)}\\ &\wedge \text{temporal : contains(?time1,}''\,2011-12'')\\ &\wedge \text{feat : people(?people1)}\\ &\wedge \text{temporal : hasValidTime(?people1, ?time2)}\\ &\wedge \text{temporal : contains(?time1, ?time2)}\\ &\wedge \text{sa : hasWithin(?shop, ?people1)}\\ &\wedge \text{isObject(?people1, ?continuum)}\\ &\circ \text{sqwrl : makeSet(?total, ?people1)}\\ &\wedge \text{sqwrl : groupBy(?total, ?shop, ?continuum)}\\ &\circ \text{sqwrl : size(?size, ?total)}\\ &\rightarrow \text{sqwrl : select(?shop, ?continuum, ?people1, ?size)}\end{aligned}$$

Example B: Let us assume we require to identify people who have met in the restaurant (see Fig. 9b):

$$
\begin{aligned}
\text{feat : people}(?\text{people1}) &\wedge \text{temporal : hasValidTime}(?\text{people1}, ?\text{time1}) \\
&\wedge \text{temporal : hasValidTime}(?\text{people2}, ?\text{time2}) \\
&\wedge \text{temporal : equals}(?\text{time1}, ?\text{time2}, \text{temporal : Minutes}) \\
&\wedge \text{abox : hasClass}(?\text{people2}, \text{feat : people}) \\
&\wedge \text{sa : isWithin}(?\text{people}, ?\text{restaurant}) \\
&\wedge \text{sa : isWithin}(?\text{people2}, ?\text{restaurant}) \\
&\wedge \text{abox : hasClass}(?z, \text{fea}t : \text{food}) \\
&\wedge \text{temporal : hasValidTime}(?\text{restaurant}, ?\text{time3}) \\
&\wedge \text{temporal : contains}(?\text{time3}, ?\text{time1}) \\
&\wedge \text{temporal : contains}(?\text{time3}, ?\text{time2}) \\
&\wedge \text{tbox : notEqualTo}(?\text{people1}, ?\text{people2}) \\
&\rightarrow \text{sqwrl : select}(?\text{people1}, ?\text{people2}, ?\text{restaurant})
\end{aligned}
$$

5 Conclusions

We introduced a model capable of handling temporal, spatial, and spatial–temporal information in an ontology. The continuum model is based on 4D-fluent and develops the continuum concept in the context of a spatial–temporal GIS. The model preserves understandable semantics for the dynamic objects represented.

The continuum model includes a set of rules and built-ins for inferring qualitative relations from quantitative data. It handles time and space independently for each object, allowing the inclusion (or not) of time and space in queries of spatial, temporal, or spatial–temporal nature.

Currently, the semantics introduced in the system allow the identification of related objects along time.

This model introduces a novel approach for handling properties and attributes for each object. The semantic management of properties and attributes for each object will be part of further research in order to develop a complete system for the semantics of spatial–temporal information.

References

Allen J (1983) Maintaining knowledge about temporal intervals. Commun ACM 26(11):832–843

Batsakis S, Petrakis E (2010) SOWL: spatio-temporal representation reasoning and querying over the semantic web. In: Proceedings of the 6th. International conference on semantic systems I–SEMANTICS 2010, vol 15. (September 2010), pp 1–15:9

Brisaboa N, Mirbel I, Pernici B (1998) Constraints in spatio-temporal databases: a proposal for classification. In: Proceedings of the 3rd. International workshop on evaluation of modeling methods in system analysis and design

Gutierrez C, Hurtado A, Vaisman A (2007) Introducing time into RDF. IEEE Trans Knowl Data Eng 19:207–218

Hayes P (2004) RDF semantics, W3C recomendation, 10 february 2004. (Online) http://www.w3.org/TR/rdf-mt/. Accessed Nov 2012

Hobbs J, Pan F (2012) Time ontology in OWL. (Online) http://www.w3.org/TR/owl-time/. Accessed Nov 2012

Innerebner, M, Bohlen M, Timko I (2007) A web-enabled extension of a spatio-temporal DBMS. In: GIS'07 Proceedings of the 15th annual ACM international symposium on advances in geographic information systems, vol 34, pp 1–34:8

Karmacharya A, Cruz C, Boochs F, Marzani F (2011) Integration of spatial processing and knowledge processing through the semantic web stack. In: Claramunt C, Levashkin S, Bertolotto M, (eds) GeoSpatial semantic. Lecture notes in computer science, vol 6631. Springer Berlin Heidelberg, pp 200–216

Klein M, Fensel D (2001) Ontology versioning on the semantic web. In: Proceedings of the first international semantic web working symposium SWWS'01. (July 2001), pp 75–91

Koubarakis M, Kyzirakos K (2010) Modeling and querying metadata in the semantic sensor web: the model stRDF and the query language stSPARQL. In: Aroyo L, Antoniou G, Hyvonen, E., Teije, A., Stuckenschmidt, H., Cabral, L., Tudorache, T., eds.: The semantic web: research and applications. Lecture notes in computer science, vol 6088. Springer Berlin Heidelberg , pp 425–439

O'Connor M, Das A (2009) SQWRL: a query language for OWL. In: Proceedings of OWL: Experiences and Directions (OWLED) 6th. International Workshop

O'Connor M, Das A (2010) A method for representation and querying temporal information in OWL. In: Proceedings of biomedical engineering systems and technologies BIOSTEC 2010, pp 97–110

Ryu K, Ahn Y (2001) Application of moving objects and spatiotemporal reasoning (2001) A TIMECENTER Technical Report

Secondo Team: Secondo (2012) An extensible database system. (Online) http://dna.fernuni-hagen.de/Secondo.html/. Accessed Nov 2012

Strobl C (2008) Dimensionality extended nine–intersection Model (DE–9IM). In: Encyclopedia of GIS Springer. Springer, Berlin, pp 240–245

Welty, C, Fikes R (2006) A reusable ontology for fluents in OWL. In: Proceedings of 2006 conference on formal ontology in Information systems (FOIS 2006), pp 226–236

Yuan M (1999) Use of a three-domain representation to enhance GIS support for complex spatial-temporal queries. Transactions in GIS 3:137–159

Modeling Concepts for Consistency Analysis of Multiple Representations and Heterogeneous 3D Geodata

Susanne Becker, Volker Walter, and Dieter Fritsch

Abstract In a dynamic world where the steadily increasing demand for up-to-date geodata drives the continuous acquisition of three-dimensional (3D) data, appropriate systems for managing and analyzing the resulting data become more and more important. Efficient solutions for handling multiple representations and data heterogeneity are of special significance. Existing geoinformation systems are still not able to cope with the huge diversity of geodata. Available approaches and systems that apply merging processes in order to generate one single representation for each real-world object are not practicable any more. Thus, our goal is a hybrid 3D geoinformation system that allows for integrated management of heterogeneous and multiply-represented geodata. Our concept is hybrid with respect to data given in different data models, dimensions, and quality levels. Multiple representations and data inconsistency can be handled through the explicit modeling of geometric correspondences.

Keywords GIS · Integration · Modeling · Three-dimensional · Consistency

1 Introduction

Dealing with multiple representations and heterogeneous data are important issues on the way toward full interoperability in geoinformation systems. Multiple representations result from the fast increasing availability of geodata. On the one hand, this flood of geoinformation implies immense potential for solving various problems. On the other hand, due to the multitude of different sensors, algorithms, and modeling concepts used for data acquisition and processing, such geodata is highly complex and heterogeneous—posing a big challenge when the data has to

S. Becker (✉) · V. Walter · D. Fritsch
Institute for Photogrammetry, University of Stuttgart, 70174 Stuttgart, Germany
e-mail: susanne.becker@ifp.uni-stuttgart.de

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_6,

be evaluated together. Data heterogeneity generally goes back to structural aspects concerning conceptual data modeling, as well as geometric and topological issues.

In terms of structural heterogeneity in conceptual modeling, Bishr (1998) distinguishes between semantic, schema, and syntactic heterogeneity (Gröger and Kolbe 2003). Semantic heterogeneity arises when dissimilar ways of understanding real-world phenomena lead to different object abstractions. Schema heterogeneity, however, denotes structural differences in the modeling concept. For instance, the same object property could be modeled as a class in concept A but as an attribute in concept B. Syntactic heterogeneity is related to different geometric data models: The two-dimensional (2D) world is mainly based on raster and vector representations; typical data models for 2.5D surfaces are grids or TINs; three-dimensional (3D) solids can be described by voxels, boundary representations (BRep), mathematical definitions such as parametric instancing, half-space modeling, or constructive solid geometry (CSG) and cell decomposition.

Geometric and topological heterogeneity is related to various aspects, particularly different reference systems and data quality. Due to the applied sensors and their configuration during measurement, data sets can differ significantly in accuracy, resolution, density, and completeness. Inconsistencies may occur when the integration of various data sets lead to spatial intersections or interpenetrations of different geoobjects, or when there remain gaps between geometries that in fact are adjacent.

Considering all these aspects, merging multiple representations in order to achieve a consistent view on the data is obviously impracticable. Solutions considering today's challenges should aim at an integrated management of multiple representations. This, however, requires geoinformation systems that allow for the explicit modeling of multiply represented data given in various geometric data models, dimensions, and quality levels. Moreover, an integrated management of multiple representations inevitably leads to inconsistencies, which have to be handled in an appropriate way. In this respect, traditional geoinformation systems that try to ensure full data consistency are not powerful and flexible enough.

In order to overcome the lack of appropriate concepts and methods for handling heterogeneous and multiply represented geodata, we propose an all-encompassing modeling concept that extends an existing International Organization for Standardization (ISO) standard in such a way that it is hybrid in the sense of data model, dimension, and quality. Our data model is designed to be an appropriate basis for a powerful and flexible 3D geoinformation system: powerful because it provides the basis for efficient consistency analyses and updating processes, and flexible because it supports multiple representations and is able to cope with structural as well as geometric and topological data heterogeneity. Visualization aspects and the modeling of semantics are not taken into account here.

The chapter is organized as follows: After an overview of related work in Sect. 2, our hybrid data model will be presented in Sect. 3. The explicit modeling of multiple representations will be introduced in Sect. 4. Section 5 will demonstrate how our modeling concepts can be used for efficient consistency analyses. Finally, Sect. 6 will conclude the paper.

2 Related Work

Data heterogeneity is a complex and multifaceted topic. Thus, approaches dealing with heterogeneous geodata usually focus on certain subproblems. They try to overcome either structural heterogeneity covering semantic, schema, or syntactic issues or geometric and topological heterogeneity. An overview of approaches that address structural heterogeneity will be given in Sect. 2.1. Section 2.2 will present current work focusing on data inconsistency caused by geometric and topological heterogeneity. Resulting consequences for our work will be discussed in Sect. 2.3.

2.1 Data Heterogeneity

Concerning structural aspects, the first investigations on hybrid data models and analysis methods go back to the 1980s; however, they only cover the 2D world (Fritsch 1988). An integrated view of hybrid 3D data has only been a topic of research for a few years. A step in this direction was taken by Dakowicz and Gold (2010), who went beyond pure 2D representations by suggesting a unified spatial model for 2D and 2.5D data. Existing approaches that are also able to handle 3D data are generally tailored to specific applications; thus, they just address subproblems, such as the combination of 2D and 3D building data (Inhye et al. 2007), the merging of TINs and grids for the representation of digital elevation models (Proctor and Gerber 2004), or the handling of CSG- and BRep-models in computer-aided design (Stekolschik 2007). Lee and Zlatanova (2008) propose a 3D data model especially suited for emergency response. Here, neighborhood relations are explicitly modeled through graph models, allowing for efficient routing algorithms; the geometric part of the data model is limited to BRep-representations, though. The same restriction holds for the slice representation introduced by Chen and Schneider (2009) as a general data representation method for 3D spatial data.

The approaches mentioned so far address structural data heterogeneity with respect to very specific application scenarios and, thus, are not suitable for general use. An application-independent conceptual framework to integrate discrete objects and continuous field-based objects on a logical level is given by Voudouris (2010); however, the study does not consider how different implementations of such object- and field-based models (e.g., vector and raster) can be managed to overcome the inherent syntactic data heterogeneity.

In principle, standards are indispensable when interoperability problems have to be avoided. The Open Geospatial Consortium (OGC) is one of the driving forces for the development of standards. OpenGIS is the brand name for standardization processes under the umbrella of OGC. A lot of OpenGIS specifications have already become an ISO standard, such as the OGC Topic 1 “Feature geometry,” whose specifications and concepts can also be found in the ISO 19107 Spatial

Scheme standard. Focusing on the description of vector data only, ISO 19107 comprises geometric and topological modeling concepts for 3D objects (Herring 2001). Because only BRep is supported, syntactic data heterogeneity remains a problem (Gröger and Kolbe 2003). ISO 19107 is based on set, theory. Each geometry is interpreted as a topological point set, which is the basis for specifying spatial comparison operators.

The Special Interest Group 3D (SIG 3D) proposed a specification for city models, CityGML, which is based on ISO 19107 (Kolbe et al. 2005). CityGML means a considerable advance to the interoperability of 3D city models. Nevertheless, a lossless integration of data that follows the CSG modeling approach still is not possible because CSG concepts are not supported. It is also not possible to integrate parametric instancing as it is often used for modeling frequently occurring similar objects.

Seen from a conceptual point of view, data integration is feasible without interoperability problems when object representations follow the same modeling standard. However, this only holds true if, additionally, the data are consistent in terms of geometric and topological aspects. Generally, multiple representations cannot be expected to be consistent as to accuracy, completeness, level of detail, etc. Thus, standardizations are only of limited use for multiply represented geodata.

2.2 Consistency and Multiple Representations

Geometric and topological data heterogeneity inevitably leads to inconsistencies in merged data sets. Gröger and Plümer (2011) support consistency analyses in 3D city models by specifying axioms for topological components and their aggregations. For this purpose, the city model is topologically interpreted as a complete and unique 3D tessellation where each geometric object is represented exactly once. Multiple representations are not supported. However, detecting and managing multiply represented objects plays an important role in geoinformation systems, especially when different data sets are to be combined. In the 2D world, a number of approaches have been developed; each of these approaches focuses on specific data types. For instance, Walter (1997) proposed a method for the matching of street data from different sources. Based on this, Volz and Walter (2004) realized the integration of multiply represented 2D vector data on the schema level. Although the range of approaches for identifying and processing multiply represented 2D geodata is wide, the situation is different for 3D data. The first ideas for analyzing the consistency of selected 3D geometries have been presented in recent years. For example, Peter (2009) compares geometric properties of planar 3D faces to estimate the consistency of different building representations. However, there is still a considerable need for research in this area.

2.3 Consequences for Our Work

The review of existing approaches dealing with heterogeneous geodata reveals a number of yet unsolved challenges on the way toward full data interoperability. The vast majority of approaches present application-specific solutions for a rather narrow range of different data types, data models, or quality levels; an overall modeling concept for arbitrary geodata is still missing. Additional problems and limitations result from the separate consideration of structural heterogeneity on the one side and geometric and topological heterogeneity on the other side. For example, although CityGML aims at interoperability on a semantic and syntactic level, the explicit treatment of geometric and topological heterogeneity is neglected; consistency analyses are not supported. However, a full interoperability and, no less important, a sustainable management of geodata, which is a basic requirement for efficient analyses and updating processes, necessarily demands for an integrated view on all heterogeneity aspects—structural as well as geometric and topological ones. Present and future challenges in the field of geoinformation demonstrate the urgent need for systems that are able to manage heterogeneous and multiply-represented geodata and, furthermore, support consistency analyses.

3 Hybrid Data Model

We introduce an application-independent modeling concept that is hybrid in the sense of structural and geometric plus topological aspects. Our data model basically builds on two modeling decisions. The first one, which will be described in Sect. 3.1, is related to the idea of expressing arbitrary object representations through geometric elements that are part of all existing data models. The second modeling decision refers to the aim of creating as much interoperability as possible and, thus, results in applying the standard ISO 19107. The geometric primitives specified in ISO 19107 are particularly appropriate to model discrete objects. However, they can also be used to define the geometric basis (e.g., sampling points) for continuous, field-based objects (Andrae 2009) and, thus, pave the path to a unified, concept for modeling both discrete object-based phenomena and continuous field-based phenomena. The basic modeling principles of ISO 19107 as well as our extensions to the standard will be explained in Sect. 3.2.

3.1 Hybrid Core

In order to overcome syntactical heterogeneity, we base our data model on fundamental modeling elements that are part of the most relevant existing geometric data models. We introduce the term *hybrid core* to denote such common modeling elements. It appears that a *general* hybrid core that is valid for all data models does not exist. To prove this, it is sufficient to compare the 2D vector format with the 2D raster representation. 2D vector data is modeled through points, lines, and faces. Because lines and faces, in turn, are described by sequences of points, the point turns out to be the basic modeling element of 2D vector data. The existence of a hybrid core for vector and raster data would implicate the point to be a basic modeling element of raster data, too. However, points cannot be expressed in the raster format in purely geometric terms. The explicit semantic modeling as point object is additionally required because—as a consequence from the approximating character of raster data—a raster cell could also represent a short line or a small surface.

Because a *general* hybrid core is not available, we create an *artificial* one based on the working hypothesis, which states that all modeling types considered so far (e.g., vector, raster, TIN, grid, voxel, cell decomposition, CSG, etc.) can be transferred to BRep. By internally creating BRep for all data sets, even syntactically inhomogeneous geodata can be reduced to a hybrid core comprising points, lines, surfaces, and solids. In the case of raster and voxel data, where each 2D or 3D cell is then described by its bounding lines or surfaces, respectively, this modeling concept is of course not efficient. However, according to fast advances in the development of high-speed processors and parallel computing, it seems reasonable to ignore performance issues for now. Efficient access structures can be added to the model at a later stage.

3.2 Extension of the Standard ISO 19107

In order to ensure as much interoperability as possible, we build our modeling concept on the ISO 19107 standard. ISO 19107 is a widely accepted standard for the modeling of geometric and topological aspects of a real-world phenomenon (Andrae 2009). Based on BRep, it is appropriate to describe 2D and 3D vector data as well as TINs and grids. We propose several standard compatible extensions that open the standard to further geometric representations. Here, the focus is on approximating data models such as raster and voxel, and on the CSG modeling approach.

Figure 1 shows our data model in UML notation; explanations will be given in the following sections: Essential modeling principles of ISO 19107, the basis of our data model, will be described in Sect. 3.2.1 (Fig. 1 presents corresponding object classes in light gray). The standard compatible extensions for raster and voxel data will be given in Sect. 3.2.2 (highlighted in orange, red frames (horizontally hatched)), while Sect. 3.2.3 will show how the CSG concept (highlighted in orange, blue bold frames (horizontally hatched)) can be integrated in ISO 19107. By means of the object classes and corresponding associations colored in green (diagonally hatched), Fig. 1 illustrates how various geometric data representations can be expressed by our data model.

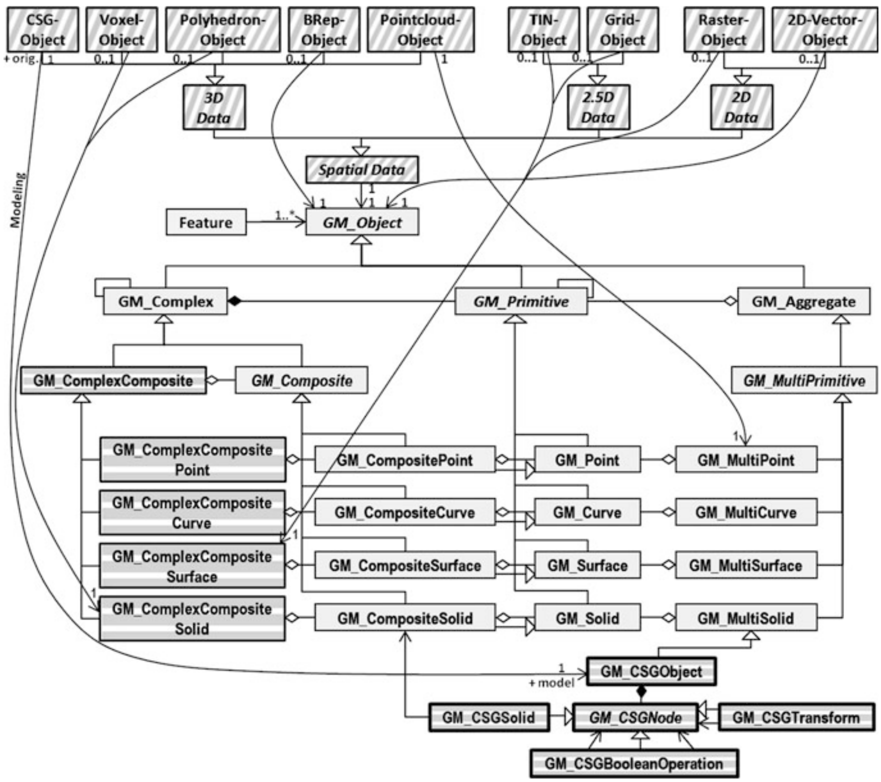


Fig. 1 Hybrid data model

3.2.1 Basic Modeling Principles

ISO 19107 defines GM_Object as a base class for the geometric properties of all geoobjects. An instance of GM_Object is either a GM_Primitive, a GM_Aggregate, or a GM_Complex. Specializations of GM_Primitive are the classes GM_Point, GM_Curve, GM_Surface, and GM_Solid. These geometric primitives cannot be divided into further primitives and, thus, represent basic elements. Instances of the class GM_Aggregate are unstructured collections of geometries free of any topological restrictions. Aggregates whose components all belong to the same primitive type are elements of the class GM_MultiPrimitive.

In contrast to GM_Aggregate, GM_Complex offers an opportunity to combine geometric elements in a structured way. Topological constraints ensure these elements to be disjoint and not self-intersecting; they are allowed to touch each other, though. A complex belongs to the class GM_Composite if the following additional conditions are fulfilled: (1) all components of the complex are of the same primitive type; (2) the complex is isomorphic to a primitive. Important specializations of GM_Composite are GM_CompositeCurve, GM_CompositeSurface, and GM_CompositeSolid.

As mentioned above, ISO 19107 additionally allows for the explicit modeling of a geoobject's topological properties by separate classes. To simplify matters for now, we do without an explicit topological modeling. Topological properties can be derived anyway when geoobjects are modeled as instances of the class GM_Complex. A geometric complex describes topology implicitly because ISO 19107 defines that—in contrast to primitives and aggregates that represent open sets—a complex contains its components plus the boundary of each component.

3.2.2 Extensions for Raster and Voxel Data

According to our working hypothesis, a raster representation of a geoobject can be interpreted as a composition of single surface elements, in which each surface element corresponds to one pixel and is described by its bounding lines. Figure 2 shows an exemplary 2D geoobject in both raster (Fig. 2a) and boundary representation (Fig. 2b). In order to emphasize the different characteristics of these two concepts, pixels are illustrated in black, surface elements in gray with black boundaries. Due to the properties and topological relations of raster cells (not self-intersecting, disjoint), such a composition of surface elements meets the requirements of a GM_Complex. But, modeling a raster object as a general complex means losing knowledge about important geometric properties because a complex does not know about its components' primitive types: ISO 19107 does not specify or restrict which primitive types may occur in a complex; even a mixture of dissimilar types is allowed.

Modeling a raster object instead as an instance of GM_CompositeSurface, which is a specialization of GM_Composite and, thus, also of GM_Complex, would preserve the knowledge about occurring primitive types. However, as will be shown by the examples in Fig. 2c, GM_CompositeSurface cannot express raster objects of arbitrary shape. The reason is that a composite is defined to be isomorphic to a primitive; consequently, a composite surface—here, the union of various raster cells—has to be isomorphic to a single-surface primitive. As ISO 19107 requires a surface primitive to be simple, i.e., free of self-intersections and self-touches; only those raster objects can be modeled as a valid composite surface whose raster cells each have at least one edge in common with another raster cell. Although this is true for Fig. 2c1, raster objects similar to the example in Fig. 2c2 cannot be modeled as composite because the outer boundary of the merged cells touches itself.

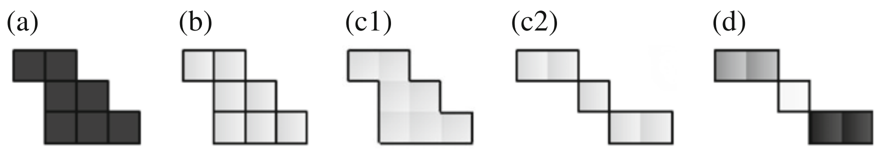


Fig. 2 **a** Raster object, **b** Raster object interpreted as BRep, **c1** and **c2** Raster objects modeled as GM_CompositeSurface, **d** Raster object modeled as GM_ComplexCompositeSurface

To overcome this problem, the new class GM_ComplexComposite is introduced as a specialization of GM_Complex. An instance of this class is a complex of several composites, which can be of different composite types. Restrictions forcing these composites to be of identical primitive type are realized through the specializations GM_ComplexCompositePoint, GM_ComplexCompositeCurve, GM_ComplexCompositeSurface, and GM_ComplexCompositeSolid. Based on this extension to the data model, it is now possible to model raster objects of arbitrary shape. The class appropriate for this purpose, GM_ComplexCompositeSurface, even allows for the modeling of completely unconnected raster cells or raster configurations in which cells are connected through just a corner, as is the case in Fig. 2c2. As illustrated in Fig. 2d, parts of the object that are isomorphic to a single surface are modeled as instances of GM_CompositeSurface; all together, they can then be interpreted as a complex of three composite surfaces, i.e., as an instance of the class GM_ComplexCompositeSurface. Extensions for the modeling of voxel representations follow analogous considerations. The new object class introduced for this purpose is called GM_ComplexCompositeSolid.

3.2.3 Extensions for CSG Data

In principle, CSG data can be converted into BRep by determining the visible bounding faces. Doing so, however, implies the loss of information on the construction process and geometric conditions of the CSG object (Gröger et al. 2005). Such information can be relevant for updating purposes.

We integrate the CSG concept in the data model through the new object class GM_CSGObject. Derived from the aggregate GM_MultiSolid, this class allows its components to overlap and penetrate each other, which is a characteristic property of CSG objects. By means of the so-called CSG node, realized through the class GM_CSGNode, the hierarchical structure of the CSG construction process can be modeled. GM_CSGNode serves as a base class to define transformations, Boolean operations, and CSG solids, the constructive elements of a CSG object. A Boolean operation, for example, refers to two nodes to which it is applied. Transformations are modeled accordingly. A CSG solid refers to an instance of GM_CompositeSolid, which ensures that the solid's boundary is a part of the object.

Our object-oriented way of modeling CSG objects makes it possible to completely hide their constructive design from the rest of the standard. Special analysis methods for CSG objects can be introduced without changing the standard.

4 Management of Multiple Representations

The hybrid data model proposed in Sect. 3 can cope with structural heterogeneity; data of different dimensions and geometric representations can be handled, analyzed, and visualized together. In order to further increase the flexibility of our data model, we introduce concepts for the explicit modeling of multiply represented data in consideration of geometric and topological data heterogeneity (Sects. 4.1 and 4.2).

4.1 Modeling Concept for Multiple Representations

An arbitrary geoobject, which is called a *feature* in our data model, can be realized through one or more representations, each of them modeled as an instance of GM_Object. These instances actually do not need to cover the geoobject completely, but instead can also describe only parts of the object. Thus, on the one hand, our modeling concept provides the possibility to manage multiply represented geoobjects. On the other hand, it is also feasible to combine various object parts to one geoobject, even if these object parts stem from very different geometric representations (e.g., from a TIN mesh and a voxel representation).

However, an efficient usage, analysis, and interpretation of the data is only possible if geometric equivalences between different object representations are known, i.e., if it is known which geometry of one representation corresponds to which geometry of another representation of the same geoobject. In the following, we will denote such geometric correspondences between different object representations as *hybrid identities*.

Assuming an ideal world, in which coordinates of corresponding object representations coincide exactly, hybrid identities are given implicitly through incident geometries. As an example, Fig. 3 (left) depicts several representations of a simple building: a 3D vector representation of the building's solid, the 2D vector outline, a raster representation of the building's footprint, and a 3D point cloud observed at one building face. Because the boundaries of these representations exactly match with each other, corresponding geometries can automatically be derived by means of geometric comparisons.

Such an ideal situation illustrated in Fig. 3 (left) is a special case that can only occur as result of specific conversions or when one representation has been created based on another (e.g., a 3D solid through extruding a 2D outline). In practice, we usually face geodata that are geometrically and topologically heterogeneous due to

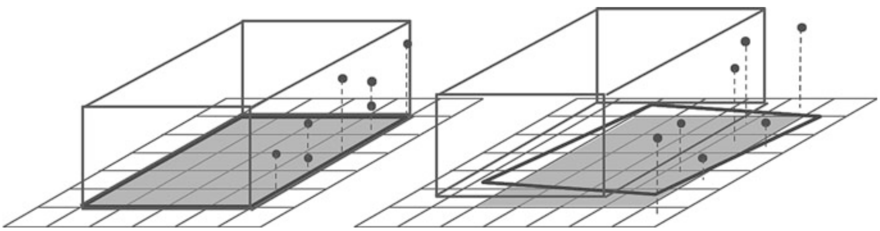


Fig. 3 Multiple representations of a building in an ideal, consistent, and error-free world (*left*) and in the real-world (*right*)

inaccuracies, generalization processes, or incomplete data acquisition. As a consequence, multiple object representations derived thereof show significant discrepancies between corresponding geometries (Fig. 3 (right)). Thus, knowledge about hybrid identities is not given implicitly any more but has to be added explicitly instead. Details on modeling aspects and the possible usage of hybrid identities are described in Sect. 4.2.

4.2 Modeling Concept for Hybrid Identities

Figure 4a shows the concept we developed for the explicit modeling of hybrid identities. The concept goes beyond the modeling of purely geometric aspects because knowledge about correspondences and relations between different object representations is introduced. The class HybridIdentity is used for managing hybrid identities. Each hybrid identity refers to at least two mutually corresponding structures modeled as instances of the class HybridElement. Depending on whether such a *hybrid element* stands for a single primitive or is a collection of several primitives, it can be a *hybrid primitive*, a *hybrid complex,* or a *hybrid aggregate*. The way in which several hybrid primitives are combined to a hybrid complex or aggregate follows the basic modeling principles as proposed in Sect. 3.2.1. In order to avoid redundancy, a hybrid primitive does not contain an explicit geometric description but refers to an existing instance of the class GM_Primitive. Conversely, an instance of GM_Object refers to all hybrid identities in which it is involved.

The data model for hybrid identities is designed to offer as much flexibility as possible. Being modeled independently of each other, hybrid identities can be defined for either a whole object or components of it. Additionally, one and the same object or object part can belong to several hybrid identities. Based on the example of a multiply represented 2D line object, Fig. 4b–e demonstrates a small selection of the many possibilities to define hybrid identities. Figure 4b shows the

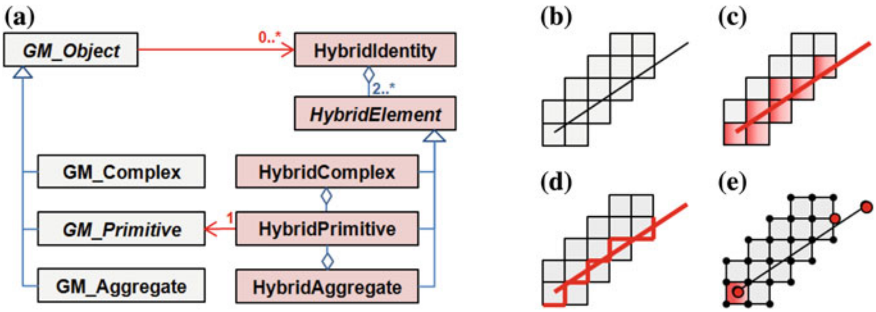


Fig. 4 **a** Data model for hybrid identities, **b** Raster and vector representation of a line object, **c**–**e** Exemplary definitions of geometric correspondences (*red*, *bold*)

two representations available for the 2D line object. The linear one (rep_A) stems from a 2D vector representation and is modeled as an instance of GM_Complex (here, consisting of a single line and its boundary). The areal one (rep_B) originates from raster data and is given as an instance of GM_ComplexComposite-Surface. Possible hybrid identities can be defined for the following geometries: the line of rep_A and a subset of the surface patches of rep_B (Fig. 4c); the line of rep_A and a sequence of lines bounding the surface patches of rep_B (Fig. 4d); the points bounding the line of rep_A and a single surface patch, as well as a boundary point of rep_B (Fig. 4e).

5 Hybrid Consistency

The concepts proposed in Sect. 4 provide the basis for efficient consistency analyses between arbitrary data sets. Through the combination of various GM_Objects to one feature (Sect. 4.1) and the definition of hybrid identities (Sect. 4.2), the pure geometric modeling is enriched by information on semantic entities, (i.e., knowledge about the relations between different object representations is introduced). This explicitly modeled knowledge about multiple representations and geometric correspondences provides the basis for consistency analyses. Integrated in our hybrid data model, consistency can now be evaluated and quantified even for highly heterogeneous object representations that stem from different data models and have different dimensions and quality levels. The traditional understanding of consistency as the lack of contradiction within a single data set or between two structurally homogeneous data sets consequently has to be extended to a so-called *hybrid consistency*. The definition of the term hybrid consistency is closely related to our hybrid data model and directly refers to the modeling concepts proposed for multiple representations and hybrid identities. Thus, it is possible to determine the degree of hybrid consistency between different object representations that describe the same real-world object either entirely or partially. Based on two exemplary scenarios, the following two sections will demonstrate how our modeling concepts can be used to evaluate hybrid consistency for entirely overlapping object representations on the one hand (Sect. 5.1), and partially overlapping or adjacent object representations on the other hand (Sect. 5.2).

5.1 Hybrid Consistency for Entirely Overlapping Object Representations

By “entirely overlapping object representations,” we mean data sets that solely describe geometries of the same real-world object. Although these geometric descriptions do not have to cover the real-world object completely, they do not contain geometries from other neighboring real-world objects. Various aspects of our modeling concepts support the evaluation of hybrid consistency between multiple representations of one and the same real-world object.

First, the assignment of various GM_Objects to one feature object indicates all representations that are available for a specific real-world object. These multiple representations provide the geometric input for the consistency analysis.

Second, following the idea of expressing all data models through the geometric elements of a hybrid core, it is ensured that the object representations that are to be analyzed show structural homogeneity. According to our artificial hybrid core, which is based on BRep, the difficulty of comparing data sets given in different data models and dimensions is consequently reduced to the problem of comparing 2D or 3D points, lines, surfaces, and solids.

Third, applying the basic modeling principles of the ISO 19107 standard includes the interpretation of geometries as sets of points. For solids and polygonal objects, such point sets consist of all points that cover the objects' surfaces; for a line object, the point set comprises the line points; for a point object, the set is defined by a single point. Understanding arbitrary geometries as point sets is a further simplification because now it is not necessary to distinguish between points, lines, surfaces, and solids any more.

To give an example for entirely overlapping object representations, we refer to the situation illustrated in Fig. 3 (right). Here, a building is represented by several data sets showing significant discrepancies. The usage of the hybrid modeling concepts described above transfers the complex consistency analysis between heterogeneous data sets—originally given in different data models—to the much simpler problem of comparing structurally homogeneous point sets. These point sets, however, may appear in different dimensions. When 2D and 3D data has to be compared, the z-coordinates of the 3D point sets are neglected. Doing so, we create dimensionally adapted sets of points, as can be seen in Fig. 5 (left). For the comparison and analysis of such point sets, we can fall back on a number of approaches and metrics that have already been developed (Alt and Guibas 1996).

5.2 Hybrid Consistency for Partially Overlapping or Adjacent Object Representations

We use the term "partially overlapping or adjacent object representations" to denote data sets that mainly describe different regions of the world but, at the same time, are connected to each other due to geometric correspondences. These can be data sets showing real-world objects of the same type, such as two partially overlapping street networks from different providers. Beyond this, the object representations can also show different object types, such as an indoor model of a building on the one hand whose entrance is connected to a street network on the other hand. In both cases, the determination of hybrid consistency is restricted to those entities that are represented in both data descriptions.

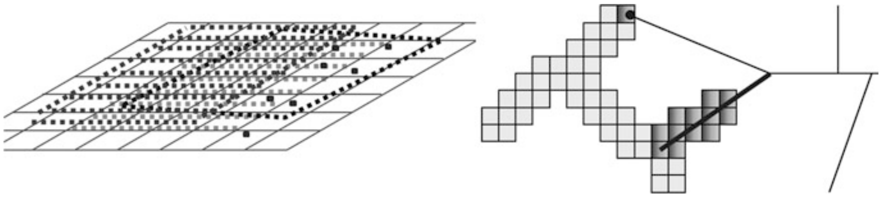


Fig. 5 Consistency analysis for entirely overlapping object representations (*left*) and consistency analysis for partially overlapping object representations (*right*)

The basis for evaluating hybrid consistency for partially overlapping or adjacent object representations is given by the possibility to explicitly model geometric correspondences as hybrid identities. The geometries to which a hybrid identity refers can be interpreted as multiple representations of local entities showing an entire overlap. Thus, the comparison of the respective geometries can be treated in the same way as discussed in Sect. 5.1.

As an example for partially overlapping data sets, we extend the geometric configurations illustrated in Fig. 4b–e to a scenario of two different network representations. As indicated in Fig. 5 (right) by the geometries highlighted in red (bold), data sets may be connected through more than one hybrid identity. One possibility to get an overall consistency value is to compute a weighted mean out of the consistency values determined for all individual hybrid identities. The weight of a hybrid identity either can be estimated from the accuracy of the geometries involved or may result from the ratio of the spatial extension of its geometries compared to the spatial extension of the geometries of all other hybrid identities. Detailed investigations on these and further possibilities for aggregating consistency values of several hybrid identities will be part of our future work.

6 Conclusions and Outlook

We proposed a data model that is meant to provide an application-independent conceptual basis for smart geoinformation systems. The data model is hybrid in the sense of structural and geometric aspects. Through targeted extensions of an existing ISO standard, our concept is able to bridge the gap between 2D, 2.5D, and 3D data and break down barriers between various modeling strategies. The consideration of geometric and topological heterogeneity is realized on the conceptual level: Hybrid identities can be defined for various objects or object parts whether they are geometrically and topologically consistent to each other or not. The explicit modeling of such geometric correspondences allows not only for the connection of objects or object parts given in different types, geometric data models, dimensions, and quality levels, but it also supports consistency analyses and updating measures, which is an important aspect considering the frequently occurring changes in geodata. The system supports multiple representations that can be based on either the same or differing data models. Additionally, it is also possible to model parts of a single object using different modeling concepts. Although, for example, the main body of a building can efficiently be represented by cell decomposition, decorative elements such as 2.5D reliefs could be added as fine surface meshes.

In future work, we will evaluate the efficiency and the potential of our hybrid modeling concept based on application scenarios. One application might be mapping and integrating multiply represented inconsistent building data into our hybrid data concept, and modeling hybrid identities for corresponding geometries. Another scenario could be the connection of disjoint or only partially overlapping data sets—for example, vector representations of street data and raster images of evacuation plans representing the interior of buildings—as basis for an outdoor–indoor navigation.

Through the integrated evaluation of geodata from different sources covering different aspects of real-world objects, we expect a deeper insight in geometric but also semantic relations. Explicitly defined hybrid identities constitute links between various data sets, and, thus, provide a basis for the inference and comprehension of higher context.

References

Alt H, Guibas LJ (1996) Discrete geometric shapes: matching, interpolation, and approximation: a survey. In: Technical report B 96-11, EVL-1996-142, Institute of Computer Science, Freie Universität Berlin

Andrae C (2009) Spatial Schema—ISO 19107 und ISO 19137 vorgestellt und erklärt. From the series. In: Andrae C, Fitzke J, Zipf A (eds) OpenGIS essentials—Die Geo-Standards von OGC und ISO im Überblick. Wichmann, Heidelberg

Bishr Y (1998) Overcoming the semantic and other barriers to GIS interoperability. Int J Geograph Inf Sci 12(4):299–314

Chen T, Schneider M (2009) Data structures and intersection algorithms for 3D spatial data types. In: Proceedings of the 17 ACM SIGSPATIAL, pp 148–157

Dakowicz M, Gold C (2010) A unified spatial model for GIS. In: IAPRSSIS, Hong Kong, vol XXVIII, part 2, pp 22–27

Fritsch D (1988) Hybride graphische Systeme—eine neue Generation von raumbezogenen Informations systemen. Geo-Informations-Systeme 1(1):12–19

Gröger G, Kolbe TH (2003) Interoperabilität in einer 3D-Geodateninfrastruktur. In: Proceedings of Münsteraner GI-Tage 2003, pp 325–343

Gröger G, Plümer L (2011) How to achieve consistency for 3D city models. GeoInformatica 15(1):137–165

Gröger G, Benner J, Dörschlag D, Dress R, Gruber U, Leinemann K, Löwner MO (2005) Das interoperable 3D-Stadtmodell der SIG 3D. ZfV 6:343–353

Herring J (2001) The OpenGIS abstract specification, topic 1: feature geometry (ISO 19107 spatial schema), version 5. OGC Doc. No. 01-101

Inhye P, Hyeyoung K, Chulmin J (2007) 2D-3D Hybrid data modeling. In: Proceedings of the 27 annual ESRI user conference, San Diego (on CD-ROM)

Kolbe TH, Gröger G, Plümer L (2005) CityGML—interoperable access to 3D city models. In: Oosterom P, Zlatanova S, Fendel EM (eds) Proceedings of the 1st international symposium on geoinformation for disaster management. Springer, pp 883–899

Lee J, Zlatanova S (2008) A 3D data model and topological analyses for emergency response in urban areas. In: Geospatial information technology for emergency response, Taylor & Francis Group, London, UK, pp 143–168

Peter M (2009) Presentation and evaluation of inconsistencies in multiply represented 3D building models. In: Rothermel K, Fritsch D, Blochinger W, Dürr F (eds) Quality of context: first international workshop, QuaCon 2009, Stuttgart, pp 156–163

Proctor MD, Gerber WJ (2004) Line-of-sight attributes for a generalized application program interface. J Defense Model Simul: Appl Methodol Technol 1(1):43–57

Stekolschik A (2007) Beitrag zum ganzheitlichen Qualitätsmanagement von CAD-Modellen in der Produktentstehung. Dissertation, Uni Bochum

Volz S, Walter V (2004) Linking different geospatial databases by explicit relations. In: Proceedings of the 20 ISPRS congress, Com IV, Istanbul, pp 152–157

Voudouris V (2010) Towards a unifying formalisation of geographic representation: the object-field model with uncertainty and semantics. Int J Geograph Inf Sci 24(12):1811–1828

Walter V (1997) Zuordnung von raumbezogenen Daten—am Beispiel von ATKIS und GDF. Dissertation, DGK, Reihe C, Nr. 480, München

Modeling of Search Actions Under the Conditions of Variable Environment Properties

Victor Ermolaev

Abstract As a rule, modeling and evaluation of search action efficiency is made using the theory of search. Constraints and assumptions of the specified theory provide for an application of analytical models that assume an invariance of properties for modeling objects. However, application of such models to single situations of search causes essential errors due to insufficient adequacy of the mathematical model for real-life search situation. This is typical for an object search under a variability of environment properties affecting observer possibilities as well as possibilities for the observation object. Such problem solving is connected with applications of geospatial data that reflect the environment properties. In this chapter, the problem of search efficiency evaluation under variable water environment properties is addressed the necessity for geographic information system (GIS) application to modeling is substantiated, the role and importance of GIS are considered, an ontology of search operation is specified, and modeling results are described.

Keywords Search operation · 2D heterogeneous environment · Geoinformation system · Efficiency

1 Introduction

A successful search for underwater objects (shoals, trespassers of the guarded marine economic subjects, mines) depends significantly on accounting for the environmental features in a search area. Therefore, when modeling search actions, choosing an environment model that adequately reflects the external conditions is critical.

V. Ermolaev (✉)
St. Petersburg Institute for Informatics and Automation of RAS (SPIIRAS), 39, 14 Line, St. Petersburg 199178, Russia
e-mail: ermolaev@oogis.ru

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_7,

Environment models are divided into two main groups—deterministic and stochastic. The first group is characterized by deterministic values of environmental features at each space point. In stochastic models, at least one parameter at each point of search space is a stochastic variable. Within the specified groups, two types of models are used: models of layered-heterogeneous and two-dimensional (2D) heterogeneous environments. The main feature of the 2D heterogeneous environment model accounts for the properties' variability over vertical (by depth) and horizontal (by distance) planes. Currently, the stochastic model theory is at the formation stage and the software is complex; therefore, the practical realization of such models is limited. For this reason, only deterministic models are considered thoroughly in this chapter.

In turn, the search theory (Koopman 1956a, c; Popovich 2000; Hellman 1985) is the basis for the modeling process of maritime object searches. Up-to-date ontology and mathematical tools of search problems have been developed in detail. Koopman (1956a, b, c) described the problem's setting and presented the detailed development of a mathematical apparatus for search efficiency evaluation and optimization. Other works (Popovich 2000; Prokaev 2005) refine the problem's abstract setting and define the theory's generalized abstraction—search operation. Further studies (Prokaev 2005; Hellman 1985; Gorbunov 1980; Diner 1969; Volgin 1999) classify search operations and develop analytical models that provide quantitative evaluation of indices and their efficiency. A search operation by Popovich (2000) elucidated a certain vector that (depending on decomposition level) incorporates the following components: observer, observation system, search object, environment, boundary conditions (assumptions and constraints), and factors of direct interaction (a priori situation data, counteraction, control, data exchange).

Generally, the search efficiency within a zone depends on its area, density of distribution, search object location, observer and observation object search capabilities, search strategy, search period, and a number of constraints such as advance existence at search object, leaving the search zone by the object prior to its detection, etc. At that, mathematical models of the search theory do not handle directly the environment models. Instead, these models indirectly determine the mathematical model parameters related to evaluation of observation capabilities.

The geographic information system (GIS) development and modern computer technology capabilities create the premises for the environment model application for mathematical models of the search efficiency evaluation. Let us consider the practicability and necessity of such integration for search situations in layered and 2D heterogeneous environments.

2 Search Modeling in Layered-Heterogeneous Environments

The model of the layered-heterogeneous environment adequately describes areas of water with flat bottoms and homogeneous hydrological-acoustical and geo-acoustic conditions through the entire search area. Use of such a model allows evaluation of the observation capability invariantly to their coordinates.

To clarify the layered-heterogeneous environment model role in the search operation, modeling will use (Entiti relationship)-(ER) diagrams in graphical notation close to the Barker's notation (Barker 1990; Chen 1995). Such diagrams can help to describe the entities of different search models, as well as the interconnections between them. Note some peculiarities of the represented model:

(a) The dashed lines connecting the environment model with the observation object models and the observer models denote the possibility but not the necessity of integration of the environment model into the search efficiency evaluation model. That is, the environmental parameters are taken into consideration only at the stage of modeling in which the source data are prepared; they are not directly incorporated into the mathematical model for search efficiency evaluation.

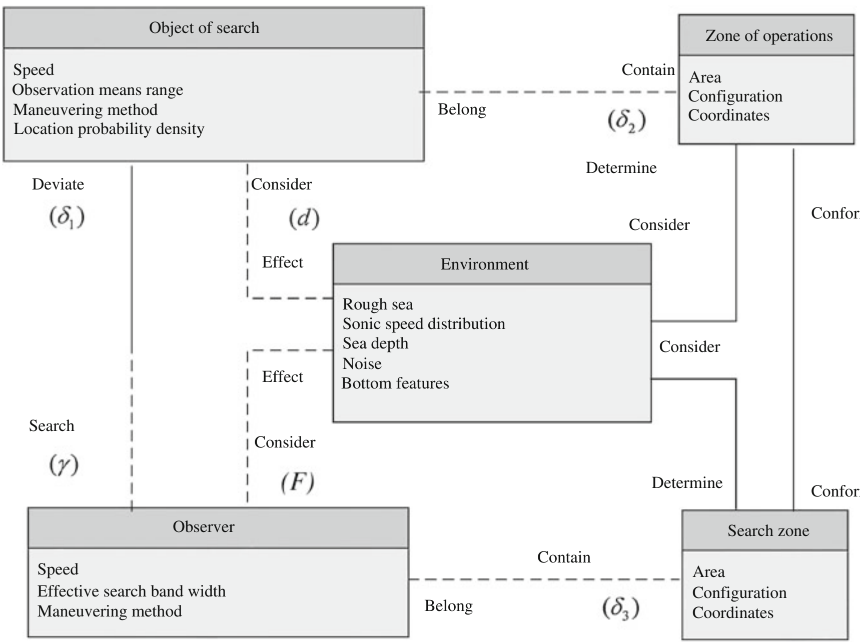


Fig. 1 Search model Entiti-relationship diagram in the layered-heterogeneous environment

(b) The dashed line connecting the search object and the zone of its operation denotes the object's possibility of leaving that zone prior to completion of the search (it is a restrictive factor).
(c) The dashed line characterizing the search by the observer denotes the possibility of terminating the search prior to the search time completion (it is a restrictive factor).

Figure 1 represents the Entiti-relationship diagram describing the search model (1).

In this case, the process of efficiency evaluation for search actions $W(t)$ consists of the following steps:

- search zone selection (configuration specification, coordinates setting, area S calculation);
- quantitative evaluation of the action range d and the effective search band width F of observation means for objects by means of calculation models considering layered-heterogeneous environment features;
- quantitative evaluation of search intensity $\gamma = f(F, v, S)$ at the observer's speed v;
- quantification k and evaluation of impact intensity for search restrictive factors δ_i;
- search efficiency evaluation $W(t)$.

Equation (1) (Diner 1969; Volgin 1999) may serve as an example of the analytical model that accounts for layered-heterogeneous environment features:

$$W(t) = \frac{\gamma}{\gamma + \sum_{i=1}^{k} \delta_i} \prod_{i=1}^{k} (1 - r_i) \left(1 - e^{-\left(\gamma + \sum_{i=1}^{k} \delta_i\right) t} \right) \quad (1)$$

Application of GIS to search modeling in the layered-heterogeneous environment is advisory and not mandatory. As a rule, a GIS is used to access the geospatial data characterizing configuration, coordinates, and area of the search zone. Examples of the software products providing for search efficiency evaluation in the layered-heterogeneous environment are shown in Figs. 2 and 3.

Figure 2 depicts the user interface that supports the source data input as well as output of the search modeling results received for a layered-heterogeneous environment in absence of search restrictive factors. Figure 3 gives an example of using GIS capacities for solving a similar task aimed at the determination of search zone position, zone configuration, and area.

In more complex versions, the environmental model of the search region may have several zones with different layered-heterogeneous environment parameters. In this case, a methodological approach can be used (Prokaev 2005) to calculate the observer's mathematical expectation of the search band effective width. This allows the use of analytical models similar to (1) and the evaluation methodology of the search efficiency given above.

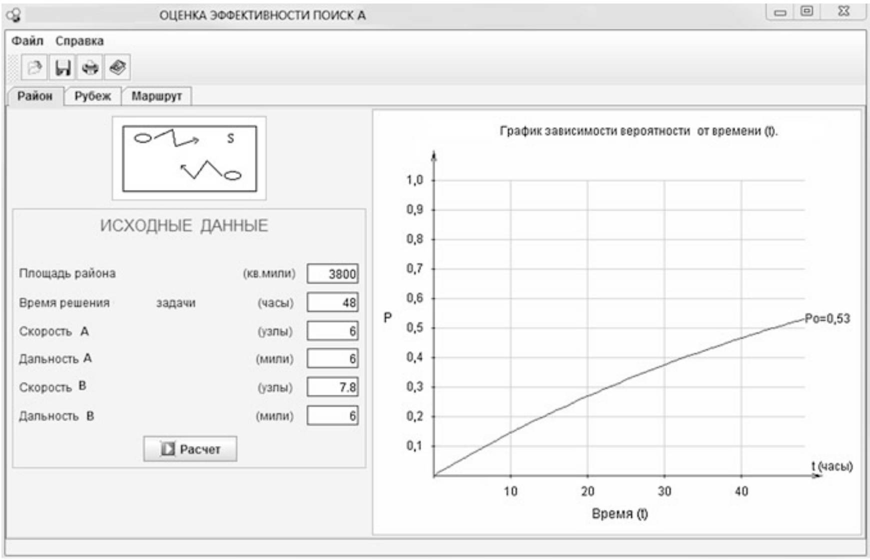


Fig. 2 Software realization of the search efficiency evaluation model without use of GIS

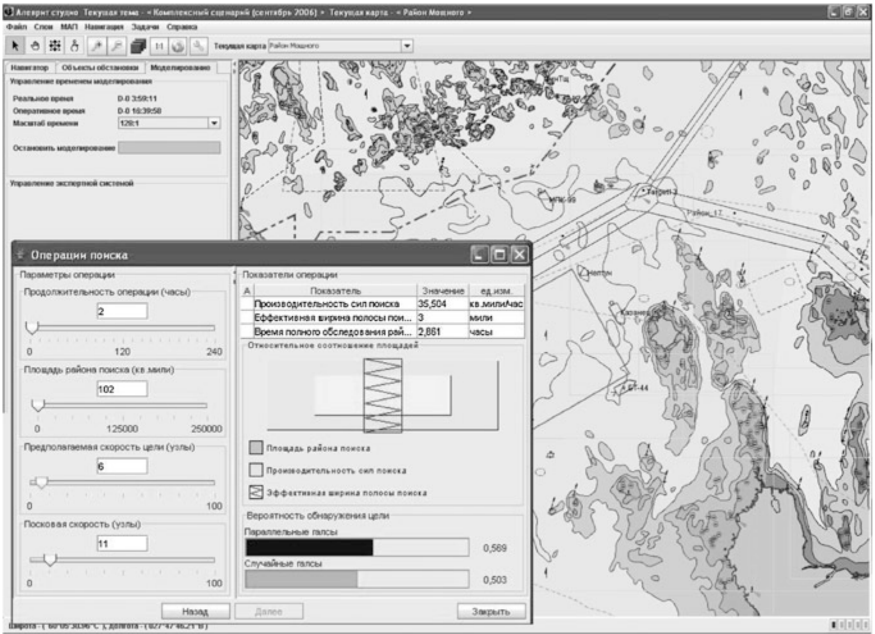


Fig. 3 Software realization of the search efficiency evaluation model based on GIS

3 Search Modeling in 2D Heterogeneous Environments

The 2D heterogeneous environment model provides more accurate evaluation of possibilities for modeling object observation because it considers a greater number of essential environment factors. The model well describes the properties of a large water area with abrupt depth differences, as well as variability of geo-acoustic and hydrological-acoustical conditions. In this case, the search capabilities for underwater observation are different at each point in the region. The differences in the impact of the environmental model on observation capabilities are graphically illustrated in Figs. 4 and 5.

In the search situation shown in Fig. 5, calculation of the effective search width is somewhat complicated. In addition, value F and, consequently, search intensity γ depend on current coordinates of the observer:

$$\gamma = \gamma(F(\varphi, \lambda), S, v)$$

In this case, the use of geospatial data in modeling search actions is mandatory. Modeling should provide binding for the observer and the observation object to the current coordinates, access to databases that characterize the environment in the point with these coordinates, and further evaluation of modeling objects' capabilities; only then should search and detection logic be realized. Obviously, this

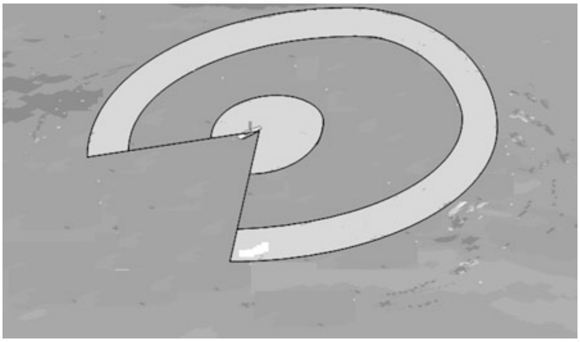


Fig. 4 Example of an evaluation of observation capabilities in a layered-heterogeneous environment

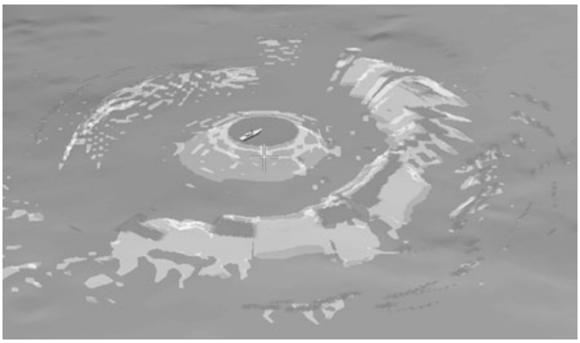


Fig. 5 Example of an evaluation of observation capabilities in a 2D heterogeneous environment

sequence of operations can only be realized by using the simulation model (Buslenko 1978; Sovetov and Yakovlev 1985).

Figure 6 shows the ER-diagram that describes the object search model in a 2D heterogeneous underwater environment. As shown in this diagram, the environment model is the mandatory feature of the search efficiency evaluation model. Additionally, the following features of the search operation model in 2D heterogeneous environment should be noted:

- Capabilities of the observation object and the observer are characterized by the zone of operations that, in turn, characterizes a probability of specified class object detection depending on its coordinates relative to the observer, rather than by the observation range or effective search width.
- In general, the zones of operations for observer and observation object cannot fully coincide and will have common sectors.
- The search zone area is not required, as is typical for analytical models, to correlate with the area instantly observed by the observer.
- The search object can be areal or volumetric rather than a point (e.g., oil spill, shoals, etc.).

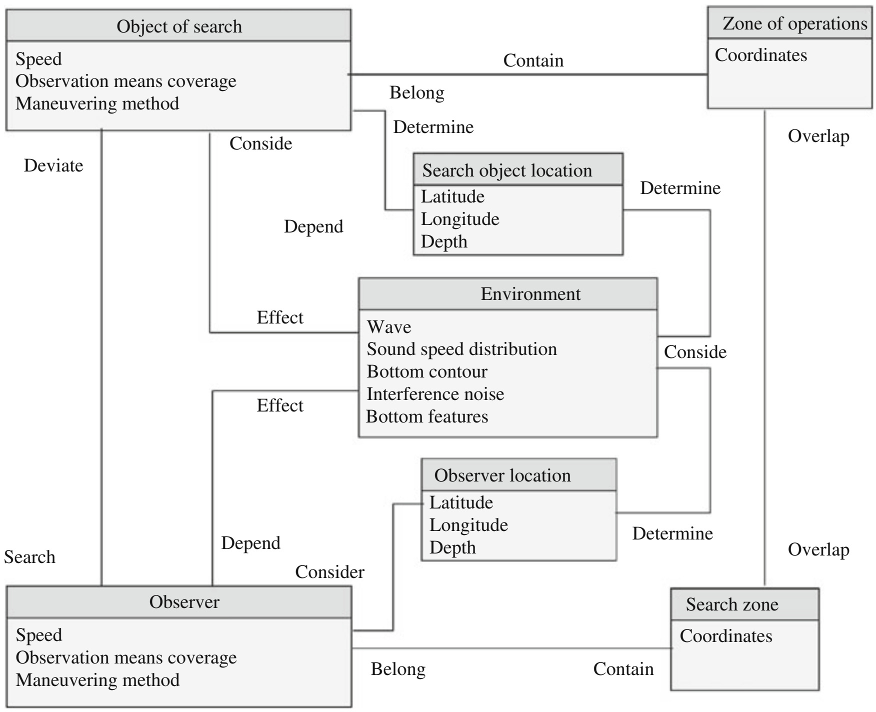


Fig. 6 ER-diagram of the search model in the 2D heterogeneous environment

Modeling with geospatial data objectively defines a necessity for integration of the search simulation model and GIS. For this, GIS has the following characteristics:

(a) It functions as the control platform of simulation. It provides realization of the modeling logic and sequence, distribution of computational resources, modeling results interpretation, errors messages generation, and development of recommendations on search efforts optimization as per the modeling results.
(b) It ensures the generation of source data for access to databases that characterize parameters of the 2D heterogeneous environment.
(c) It visualizes the modeling process. The GIS interface executes cartography-based generation of search operation elements and their binding to geographical coordinates. Additionally, the GIS interface provides visualization of observer's and search object's action dynamics. This allows defining search features in different sections of the region and selecting search strategy as intended for achieving the extreme efficiency values.
(d) It functions as a model verification tool. Capabilities of up-to-date GIS permit the model to be checked for different time scales for integrity and correctness of the source modeling data input, intermediate calculations, and consistency of modeling objects' action logic.

Figure 7 shows an example of GIS use for simulation modeling of mobile object search in a 2D heterogeneous underwater environment. The operation zones of mobile objects A and B are characterized by high heterogeneity of the bottom relief and hydrological-acoustical conditions, which is illustrated by inherent differences in operation zones and their detecting means in different points of routes. The use of a 2D heterogeneous environment model enhances the accuracy of search efficiency evaluation.

Figure 8 presents the results of detection probability calculations for the deviating mobile object within a 3,800 mi^2 zone made by a mobile observer over a 48-h time span. The search area is characterized by high environmental variability. In the first version (curve 1), search efficiency evaluation was made via an analytical model. To evaluate capacities of the observation object and the observer, the layered-heterogeneous model was used. In the second version (curve 2), the evaluation was made via a simulation model. Here, averaged data specifying the modeling object capacities calculated for the first version were used in the process. In the third version (curve 3), evaluation was made via the simulation model and geospatial data characterizing the 2D heterogeneous environment model. The simulation modeling (versions 2 and 3) was carried out and processed 3,800 tests that provided the evaluation of detection probability, with accuracy equal to 0.01 at validity equal to 0.95.

Pursuant to the diagrams in Fig. 8, the detailed recording of environment properties for version 3 allows increasing accuracy of search object detection probability evaluation by 18–20%. This, in turn, allows a accurate evaluation of search efforts to be made for achieving the preset search result.

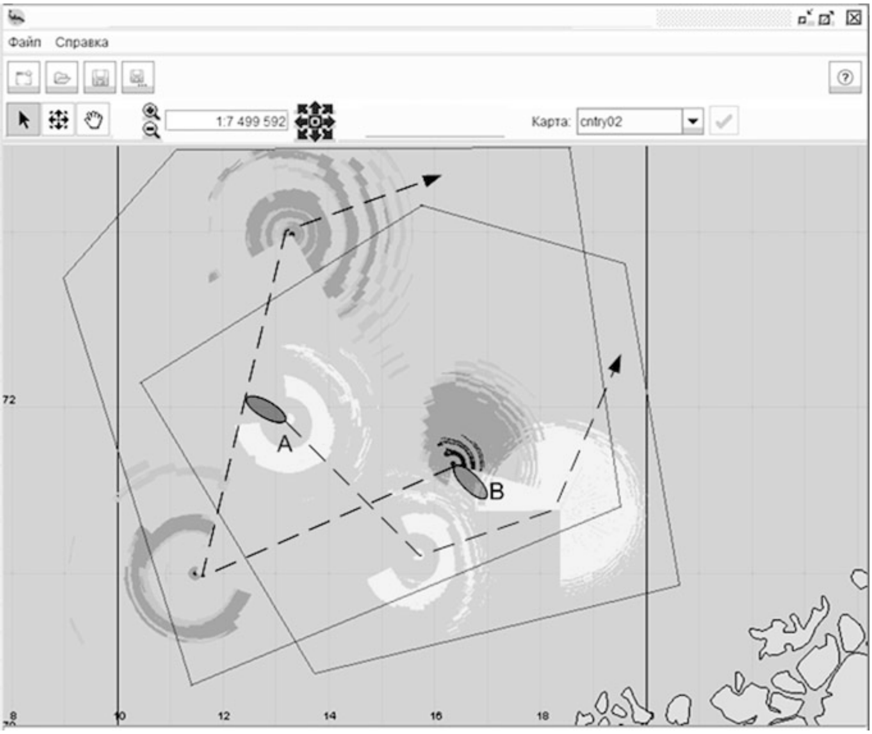


Fig. 7 Visual representation of search simulation via GIS

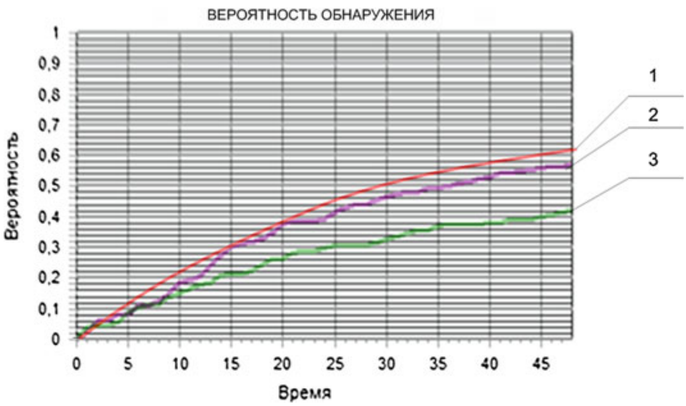


Fig. 8 Modeling results

4 Conclusion

The use of 2D heterogeneous environment models creates ample opportunities for evaluating the search efficiency and optimization in zones with high variability of the bottom relief, geo-acoustic conditions, and hydrological-acoustical conditions. Here, the presented research shows that, for the considered search situations, modeling systems with the following components may be recommended as tools to perform the desired calculations:

- geoinformational interface;
- environment database (bottom relief, geo-acoustic and hydrological-acoustical conditions);
- calculation models for evaluation of simulation object possibilities in a 2D heterogeneous environment;
- simulation model for search efficiency evaluation;
- expert system that solves a number of problems related to calculations' management (search optimization, identification of search regularities, development of recommendations intended for optimizing calculations, and search as a whole) and problems related to distribution of computational resources.

The use of intelligent GIS (Popovich 2009; Popovich et al. 2009, 2011; Ermolayev and Makshanov 2011) could be a suitable platform for realizing this structure of a modeling system. However, improved accuracy evaluation of search operations efficiency may considerably complicate the mathematical models used in evaluation of modeling object capacities. It may also increase the necessary computational and time resources, the need for developing and maintaining the environment database topicality, and the need to develop complex simulation models to realize the search logic.

References

Barker R (1990) CASE* method: entity—relationship modeling. Addison-Wesley, MA

Buslenko NP (1978) Simulation of complex systems, Moscow

Chen P (1995) ER model—toward a unified view of data//database control systems, p N3

Diner I (1969) Operational research: Naval Academy

Ermolayev VI, Makshanov AV (2011) In: IGIS capabilities application to controlling polystatic detection systems ensuring security of marine economic activity/works of 13th all-Russian scientific-practical conference "guard and security"

Gorbunov VA (1980) Targets detection efficiency Moscow

Hellman O (1985) An introduction to the theory of optimal search Moscow

Koopman BO (1956a) Operations Research. 4(3):324–346

Koopman BO (1956b) Operations Research. 4(5):503–531

Koopman BO (1956c) Operations Research. 5(5):613–626

Popovich VV (2000) Modeling, evaluation of efficiency and optimization of naval surveillance systems (theory of mobile objects research). St Petersburg

Popovich V (2009) Concept for corporate information systems based on GIS technologies development. In: Proceedings of IF&GIS-09, May 17–20, 2009, lecture notes in geoinformation and cartography, St. Petersburg, Springer, 371pp

Popovich VV, Ermolayerv VI, Leontiev YB, Smirnova OV (2009) Hydroacoustic fields modeling based on intellectual geographic information system. Artif intell Decisions Making, N 4, 2009

Popovich V, Vanurin S, Kokh S, Kuzyonny V (2011) Intellectual geographic information system for navigation safety/systems. 25(8)

Prokaev AN (2005) Information fusion in objects search problems with the intellectual geographic information systems. Collected papers of IF&GIS seminar. St. Petersburg

Sovetov BY, Yakovlev SA (1985) System modeling, Moscow

Volgin NS (1999) Operational research. Naval Academy, St Petersburg

OpenStreetMap-Based Dynamic Ridesharing Service

Alexander Smirnov, Nikolay Shilov, Alexey Kashevnik, and Nikolay Teslya

Abstract This chapter describes a service-based approach to dynamic ridesharing based on the smart space concept. Ridesharing is the shared use of a car by a driver and one or more passengers. Dynamic ridesharing supposes dynamic formation of carpools depending on the current situation. The approach uses a set of open-source applications for routing and visualization of joint trips based on the OpenStreetMap platform. An open-source Smart-M3 information platform is utilized as an information fusion infrastructure, which acquires information about the current situation from ridesharing participants and other services. One possible use of the presented service is supporting tourists in a certain region.

Keywords GIS · Information fusion · Ridesharing · Ontology · Smart space

1 Introduction

Ridesharing (also known as carpooling, lift-sharing, and *covoiturage*) is the shared use of a car by a driver and one or more passengers, usually for commuting (Abrahamse and Keall 2012). Dynamic ridesharing (also known as instant ridesharing, ad hoc

A. Smirnov (✉) · N. Shilov · A. Kashevnik · N. Teslya
St. Petersburg Institute for Informatics and Automation RAS (SPIIRAS), Saint Petersburg, Russia
e-mail: smir@iias.spb.su

N. Shilov
e-mail: nick@iias.spb.su

A. Kashevnik
e-mail: alexey@iias.spb.su

N. Teslya
e-mail: teslya@iias.spb.su

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_8,

ridesharing, real-time ridesharing, or dynamic carpooling) denotes a special implementation of a ridesharing service that enables the dynamic formation of carpools depending on the current situation. Typical features of this type of ridesharing are: arrangement of one-time trips instead of recurrent appointments for commuters; the use of mobile phones for placing ridesharing requests and offers; and automatic and instant matching of rides through a network service.

The following schemes are used by people in different countries to search for fellow travelers:

- Search via public forums and other communities (eRideShare.com 2012; PickupPal 2012; Zimride 2012; RideshareOnline 2012; rideshare.511.org 2012; CarJungle 2012). The advertisements about trips are posted on a website by users. This advertisement includes the start and end points, some information about the person who posted the ad, trip cost, time of the trip, etc.;
- Search via private web services. People can get an account in a private service only if they have an invitation (Zimride has an interface for universities and companies);
- Search via special applications on mobile devices. With these applications, users can edit their profiles, routes, and search for fellow travelers (PickupPal 2012; Avego 2012);
- Search via agents (e.g., taxi companies);
- Pick-up points (not pre-arranged).

This chapter describes a service-based approach to dynamic ridesharing based on the smart space concept. For routing and visualization of joint trips in a ridesharing system, a geographic information system (GIS) that satisfies the following requirements is needed: routing support, no limitations on amount of routes, address search support, maximal coverage area, and data accuracy.

In this chapter, the following two most powerful GIS are considered and analyzed for comparison: Google Maps and OpenStreetMap (2012) (Table 1). According to the information presented in Table 1, the GIS based on the OpenStreetMap data better satisfies the defined requirements. The use of Google Maps in the project is not possible because Google has a daily limit on the number of calculated routes; in addition, in some countries (such as Russia), detailed maps are available only for large cities.

The following scenario can be considered for a tourist who is going to visit several remote places of interest in a certain region. The tourist arrives to St. Petersburg. He/she is going to attend the Hermitage museum. His/her mobile device shares current location and destination point (the Hermitage). A special broker finds a driver who transports the tourist to the Hermitage using the algorithm for finding matching driver and passenger paths (see Sect. 5).

The approach presented in this chapter is based on a decentralized information fusion infrastructure. This approach allows increased service stability and speed and reduced network load. The rest of the chapter is structured as follows. Section 2 presents the Smart-M3 platform. Section 3 introduces the logistic service ontology used for enabling interoperability between different devices in the smart

Table 1 Google Maps and OpenStreetMap comparison

	Google Maps	OpenStreetMap
Coverage area and accuracy	– Whole world, but the level of detail varies by country	– Whole world, but accuracy depends on the community activity. Maps are updated every week
Address searching	– Supported	– Supported
Routing	– Provided by Google	– Online and offline applications, developed by the OpenStreetMap community
Limitations	– No more than 2,500 route calculations per day (paid license allows up to 100,000 route calculations) – Up to 8 key points are allowed – A limitation to the query length (up to 2,048 symbols)	– The main OpenStreetMap server has a limitation on the amount of queries for map rendering. However, developers can use their own servers to render the map and routing. In this way, all limitations depend on the developers' hardware and software

space. An OpenStreetMap-based GIS for a ridesharing service is given in Sect. 4. The algorithm for finding matching driver and passenger paths with two heuristics is presented in Sect. 4. The system's working scenario can be found in Sect. 6. Main results are summarized in the conclusion.

2 Information Fusion with Smart-M3 Platform

The open-source Smart-M3 platform (2012) has been used for the implementation of the presented ridesharing system. The use of this platform makes it possible to significantly simplify further development of the system; it includes new information sources and services and makes the system highly scalable. The key idea of this platform is that the formed smart space is device, domain, and vendor independent. Smart-M3 assumes that devices and software entities can publish their embedded information for other devices and software entities through simple, shared information brokers. Information exchange in the smart space is implemented via HTTP using a uniform resource identifier (URI) (Berners-Lee et al. 2012). Semantic Web technologies have been applied for decentralization purposes. In particular, ontologies are used to provide for semantic interoperability.

The Smart-M3 platform consists of two main parts: information agents and kernel (Fig. 1) (Smart-M3 at Wikipedia 2012). The kernel consists of two elements: a semantic information broker (SIB) and data storage. Information agents are software entities installed on the mobile device of the smart space user. These agents interact with the SIB through the Smart Space Access Protocol (SSAP) (Honkola et al. 2010). The SIB is the access point for receiving the information to be stored or retrieving the stored information. All this information is stored in the

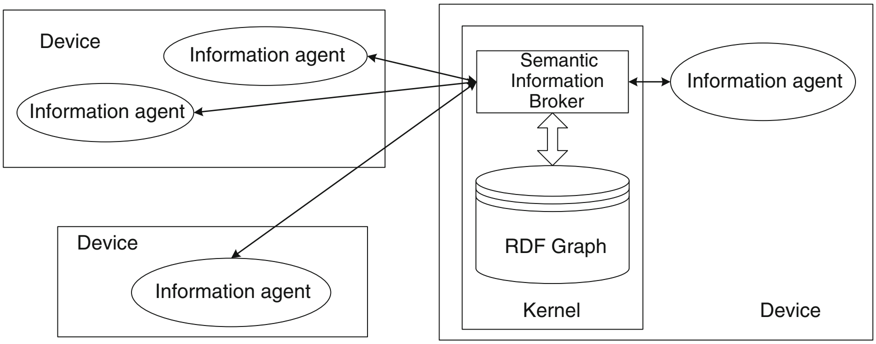


Fig. 1 Smart-M3 platform. RDF. resource description framework

data storage as a graph that conforms to the rules of the resource description framework (RDF) (2012). In accordance with these rules, all information is described as “Subject-Predicate-Object”. More details about Smart-M3 can be found in Honkola et al. (2010).

3 The Ridesharing Service Ontology

The logistic service ontology describes the domain area of ridesharing at the macro level (Fig. 2).The macro-level ontology is based on integration of parts of the mobile device ontologies. The logistics service ontology consists of three main parts: vehicles, actors, and paths. More details about the logistics service ontology can be found in Smirnov et al. (2010).

3.1 Actors

The actors are drivers, passengers, and cargo items. All of them are associated with vehicles and have paths. For example, the driver has his/her own car and several

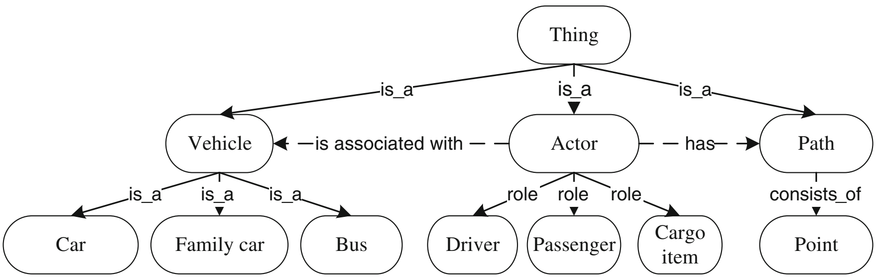


Fig. 2 Logistics service ontology on the macro level

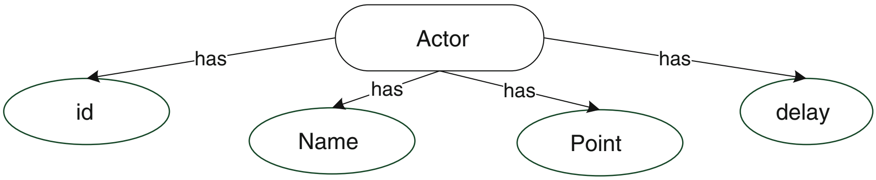


Fig. 3 Class "actor"

points defining his/her home, work, and other locations. The passenger may prefer some vehicle type and has points of home, work, and other locations. Cargo items have size and vehicle type needed for their transportation.

The class actor consists of (Fig. 3):

- ID. Unique identification for each user;
- Name. First and last name of the user;
- Point. Path point belonging to the user's path (two minimum: the start and the end);
- Delay. Maximal possible time of waiting in the meeting point.

The class "Driver" is a subclass of the class Actor and inherits all its properties with two own of its properties:

- Vehicle. Vehicle type;
- Detour. Maximal detour from the shortest path.

The class "Passenger" is a subclass of the class Actor and inherits all its properties with own property "Detour" (the same as in the class "Driver").

The class "Cargo item" is a subclass of the class Actor and inherits all its properties with own property "Size," defining the physical size of the cargo item.

3.2 Paths

For the path definition, a set of points is used. This set is an ordered list of key points obtained as result of the shortest path searching algorithm (e.g., Dijkstra or A*). The class "Point" has the following structure (Fig. 4):

- previousPoint. Contains the previous path point. For the start point, its value is "FALSE";
- Latitude;
- Longitude;
- driveByVehicle. If the point belongs to the passenger, it contains the driver who gives a ride to this passenger. If the passenger walks, then its value is "FALSE";
- vacantseats. The number of vacant seats in vehicle at this point;

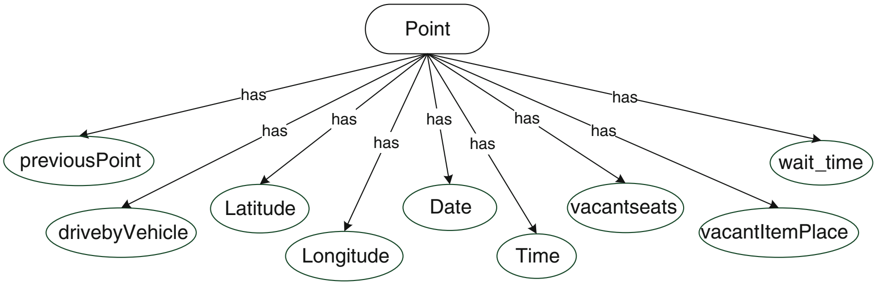


Fig. 4 Class "Point"

- vacantItemPlace. The number of vacant places for cargo items;
- Date. The date, when the user will be at this point;
- Time. The time, when user will be at this point;
- Wait_time. How long the user will be waiting at this point.

Because the ontology in the smart space is represented in the RDF standard, it looks like the following:

('user1',	'name',	'Name Surname')	- name of user1
('user1',	'is_a',	'Driver')	- user1 is a driver
('user1',	'vehicle',	'vehicle_type')	- user1 has this type of vehicle etc.

In Smirnov et al. (2010), the logistics service ontology is described in detail.

4 OpenStreetMap-Based GIS for Ridesharing Service

OpenStreetMap is a Wikipedia-like project to create a free map of the world. The data about the roads are usually obtained from the tracks recorded by GPS-recievers and imported by special programs, which further to convert it into GIS data. Usage of existing maps from other services is allowed only if it does not conflict with the interests of their owners.

For map rendering, the users can use the servers of OpenStreetMap. In addition, the developers can use their own servers with the map data and special software. Using the OpenStreetMap servers like Google Maps imposes a number of restrictions on the developers. Therefore, in this work, another server has been developed. This server is based on the Python modules for HTTP support to render a map on request from a client. The request syntax is as follows:

http://map_server_name/?method=routing&x1=latitude&y1=longitude&x2=latitude&y2=longitude,

where x1, y1 are coordinates of the start point and x2, y2 are coordinates of the end point.

The map data are distributed in special XML files. There are applications to convert OpenStreetMap XML to images for the map rendering. In this work, the Mapnik module (2012) is used for this purpose. Mapnik can use the OpenStreetMap XML file or the PostrgeSQL (2012) database, with the information imported from the OpenStreetMap XML as the data source. For this purpose, PostgreSQL must have the PostGIS (2012) extension, which provides support for geo-information processing.

An example of an OpenStreetMap XML file is presented below:

```
<bounds minlat="54.0889580" minlon="12.2487570"
  maxlat="54.0913900" maxlon="12.2524800"/>
<node id="1" version="1" changeset="1" lat="54.0900666"
  lon="12.2539381" user="user1" uid="10" visible="true"
  timestamp="">
 &lt;tag k="name" v="Neu Broderstorf"/&gt;
 &lt;tag k="traffic_sign" v="city_limit"/&gt;
&lt;/node&gt;
...
<way id="110" user="user2" uid="11" visible="true" ver-
sion="5" changeset="4" timestamp="">
 &lt;nd ref="292403538"/&gt;
 &lt;nd ref="298884289"/&gt;
 ...
 &lt;nd ref="261728686"/&gt;
 &lt;tag k="highway" v="unclassified"/&gt;
 &lt;tag k="name" v="Pastower Straße"/&gt;
&lt;/way&gt;
<relation id="5" user="user3" uid="12" visible="true"
  version="28" changeset="6" timestamp="">
 &lt;member type="node" ref="294942404" role=""/&gt;
 ...
 &lt;member type="node" ref="364933006" role=""/&gt;
 &lt;member type="way" ref="4579143" role=""/&gt;
 ...
 &lt;member type="node" ref="249673494" role=""/&gt;
 &lt;tag k="name" v="Küstenbus Linie 123"/&gt;
 &lt;tag k="network" v="VVW"/&gt;
 &lt;tag k="operator" v="Regionalverkehr Küste"/&gt;
 &lt;tag k="ref" v="123"/&gt;
 &lt;tag k="route" v="bus"/&gt;
 &lt;tag k="type" v="route"/&gt;
&lt;/relation&gt;
```

This example consists of the following main elements of the XML structure: nodes, ways, and relations. A node consists of a single geospatial point using a latitude and longitude and can be used to define point features with tags. A way is an ordered list of nodes, which normally also has at least one tag or is included within a relation. A way can have between 2 and 2,000 nodes. A relation consists of one or more tags and also an ordered list of one or more nodes and/or ways as members, which is used to define logical or geographic relationships between other elements.

PostGIS does not support routing in most recent stable version. For this purpose, pgRouting (2012) module is used. This module works with the PostGIS/PostgreSQL database and supports three algorithms for searching shortest paths: Dijkstra's algorithm, A* (A-star), and shooting-star. A* is the Dijkstra's algorithm extension with heuristic function. This algorithm provides some performance gains, especially with a large number of vertices. Shooting-star searches for the shortest path from edge to edge (not from vertex to vertex). Performance of the shooting-star is not higher than A* and the Dijkstra algorithms. Therefore, in this work A* is used for the shortest path searching.

5 Algorithm for Finding Matching Driver and Passenger Paths

The problem of finding a matching path between the driver and the passenger in the ridesharing service is formulated here. It is necessary to determine the possibility of ridesharing between users, based on the information about their routes and restrictions set by users' services. The following algorithm describes the procedure of finding a matching path that is acceptable for the driver and the passenger in the presented ridesharing service.

Let A be the start point and B be the end point of the pedestrian's path. C is the start point and D is the end point of the driver's path. The shortest driver's path, which is found with the help of GIS, is indicated by the solid line (in generally, CD is not a straight line, but it depends on the map of the region). Figure 5 shows that the driver and pedestrian move almost in the same direction; in some parts of the route, the driver can give the pedestrian a ride. This situation is indicated in the figure by the dotted line (the CABD path) and it is the simplest situation, because the meeting points match with the start and end points of the pedestrian's path. A more difficult situation is searching for a meeting point when it belongs neither to the driver's shortest path nor to the pedestrian's one, but satisfies both the driver and the passenger. One of the possible situations is indicated in the figure by the dash-dot line with the meeting points E and F (the CEFD path). These points have to meet the following restrictions:

1. The distance between the start point of the passenger and his/her meeting point should be less than the maximum allowed detour of the passenger. This area is indicated in the figure by the dotted circle around point A.

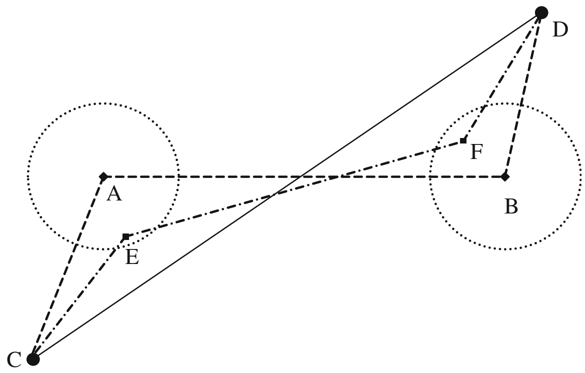


Fig. 5 A general search for matching driver and passenger paths

2. The distance between the end point of the passenger and his/her drop-off point should be less than the maximum allowed detour of the passenger. This area is indicated in the figure by the dotted circle around point B.
3. The driver's detour should be less than the maximum allowed detour.

The general scheme of the matching route searching algorithm will be as follows:

```
FOR EACH driver DO
  FOR EACH passenger Do
    Find_mathing_path(driver.path,passenger.path); //
according to the above scheme
    constraint_checking();
    IF ALL constraints IS performed THEN
      set_passenger_for_driver();
  ENDFOR;
ENDFOR;
```

The goal functions for finding the meeting points are:

- Shortest total path (interesting for the driver);
- Minimal waiting time (interesting for the driver and passenger);
- Shortest distance between the passenger's start and end points and meeting points (interesting for the passenger).

As a result, the general task of matching paths has exponential complexity; therefore, it is necessary to apply heuristics to reduce the task dimension.

5.1 *Heuristics 1*

Assumption There is no need to calculate matching paths for all pairs of drivers and passengers. It is enough to build a set of candidate passengers for every driver:

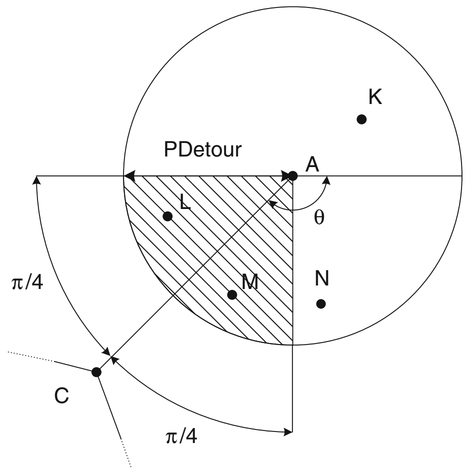


Fig. 6 The first subheuristics

$$(pp_1^x - \mathrm{d}p_i^x)^2 + (pp_1^y - \mathrm{d}p_i^y)^2 \leq (\mathrm{PDetour} + \mathrm{DDetour})^2, \quad (1)$$

$$(pp_2^x - \mathrm{d}p_i^x)^2 + (pp_2^y - \mathrm{d}p_i^y)^2 \leq (\mathrm{PDetour} + \mathrm{DDetour})^2, \quad (2)$$

where pp_1, pp_2 are the start and the end points of the passenger's path, $\mathrm{d}p_i$ is the driver's path point i, and PDetour, DDetour are the detours of the driver and the passenger.

5.2 *Heuristics 2*

Assumption There is no need to search through all possible combinations of meeting points. The following alternative subheuristics help to reduce the number of possible combinations.

The first subheuristic selects points of the sector from which the driver starts. Figure 6 shows the situation when there is only one point ("C" point) meeting constraints (1) and (2). To determine the potential meeting points, it is necessary to calculate the angle (3) and select points in the area $\left[\theta - \frac{\pi}{4},\ \theta + \frac{\pi}{4}\right]$ (points L and M in Fig. 6).

$$\theta = \mathrm{arctg}\left(\frac{C^y - A^y}{C^x - A^x}\right) \quad (3)$$

Point A will always be within the list of the possible points as the passenger's start or end point. If there is more than one point meeting constraints (1) and (2), then the search area expands. This situation is shown in Fig. 7 with two points C and F meeting the constraints (1) and (2), and point N is also included in the expanded area.

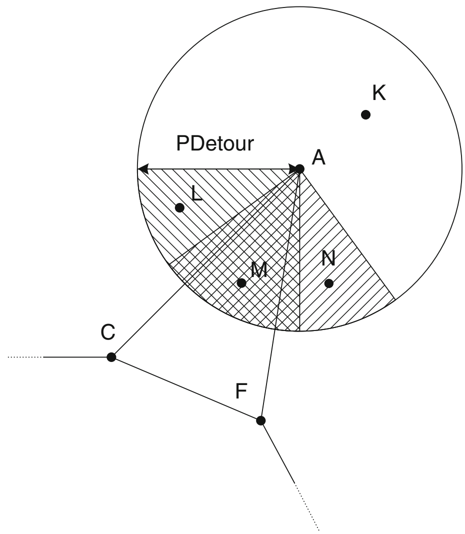


Fig. 7 The first sub-heuristics with two driver's points

The negative sides of this sub-heuristic are:

- selected points can be further than the driver's maximal detour;
- some of potential meeting points can be lost if an incorrect angle is chosen.

The second sub-heuristic (Fig. 8) selects meeting points in the intersections of the circles of radius PDetour around the passenger's start and end points, with the circles of radius DDetour around the points of the driver's path. In this case, all of the selected points are potentially reachable for both the driver and the passenger, with no need to determine the angle that restricts the selection area. The selection area can be expanded by increasing the number of the driver's path points meeting constraints (1) and (2).

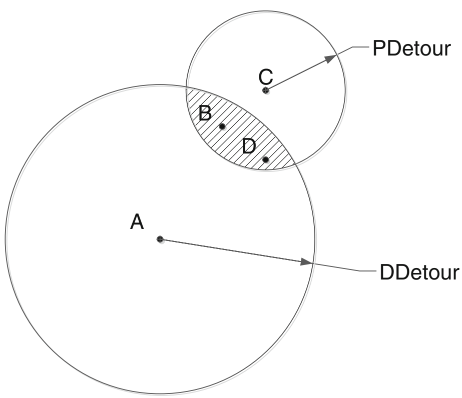


Fig. 8 The second sub-heuristics

Both subheuristics require the following constraints to work effectively:

- A large number of drivers. Heuristics have strong limitations and filter out a lot of points. If there are not enough drivers, then the use of the heuristics will rarely produce a positive result.
- A small value of DDetour. Heuristics will not be helpful with a large value of DDetour.
- Uniform distribution of roads on the map. The uneven distribution of roads (rivers, lakes, etc.) leads to a lack of roads in some sectors, which could lead to the loss of possible meeting points due to the need to detour around the obstacles and pick up the pedestrian on the other side.

Both heuristics 1 and heuristics 2 are used in the logistics service prototype. Without using the heuristics, the system finds from 10 to 12 meeting points for each pair of driver and passenger, and it needs to check all 100–144 combinations to find the best one. By using the heuristics, the number of points is reduced to 8–9 points with 64–81 combinations for each pair of driver and passenger.

6 System Working Scenario

A common system working scenario is shown in Fig. 9. The mobile application is installed by all users of the service. This application collects the information about the user's agenda, preferences (Fig. 10a), most frequent routes (Fig. 10b), etc., with the agreement of the user. Also, the user can set additional constraints, such as maximum delay, maximum detour, social interests, etc. (Fig. 10b). This information is transferred into the smart space after the internal processing and depersonalization (only signs of information are transferred, not the raw information).

During the execution of the above-described algorithm, groups of fellow travelers are formed. Then, users interactively get possible fellow travelers with

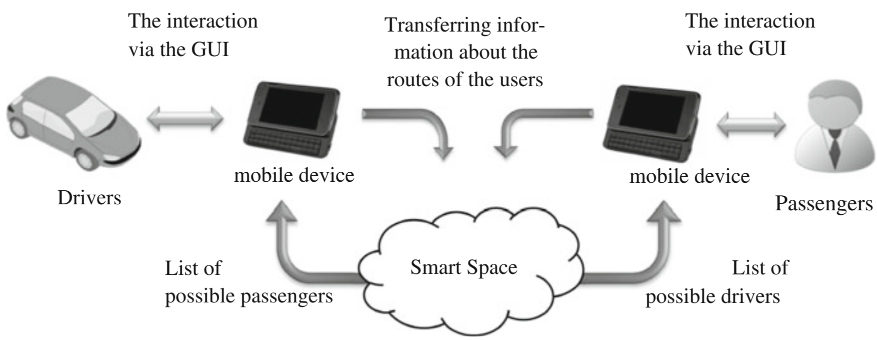


Fig. 9 System working scenario. GUI, graphical user interface

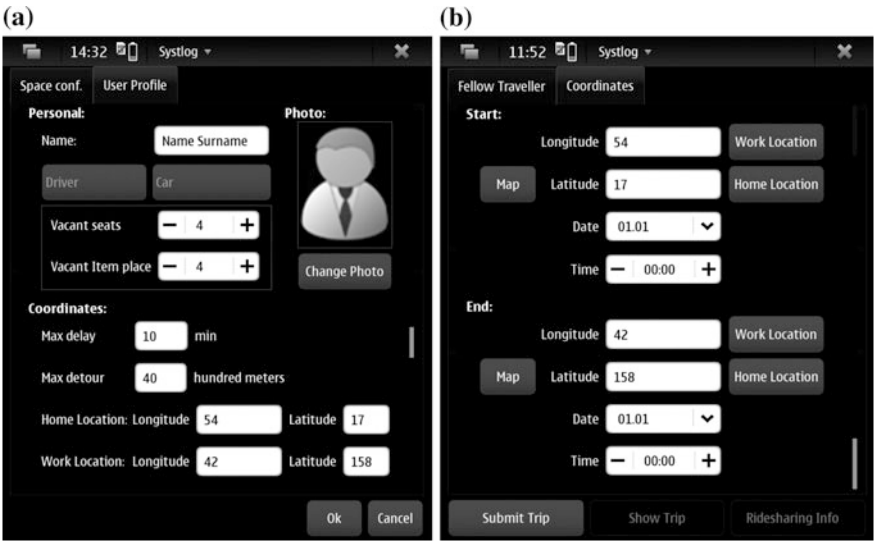


Fig. 10 Prototype screenshots (user's routes and preferences). **a** User's profile configuration, **b** User's path configuration

their profiles, meeting points, meeting time, and full recommendations about the route (Fig. 11a–d); if they have permission, users can get the link to the external resources, such as a social network page, which helps the user to make a decision. Sometimes, the suggested driver may be a friend of the friend of the given passenger (this information can be useful for the user in the decision-making stage).

Also, dynamic search is supported. Users can login and logout from the smart space, change restrictions, and then receive the list of fellow travelers within a short interval of time (almost in real time). All this work is done by the smart space, so users do not need to perform any actions to find fellow travelers.

For example, when the tourist is going to St. Petersburg, an acceptable itinerary based on his/her preferences (how many days the visitor is going to spend in St. Petersburg, his/her cultural preferences, etc.) is formed. Figure 12 presents the acceptable plan for a visitor. It consists of five museums: the Hermitage, Kunstkamera, the Museum of Karl May Gymnasium History, St. Isaac Cathedral, and Dostoevsky museum. The system automatically finds drivers for the visitor to reach these museums.

From the Hermitage (label A in Fig. 12) to the Kunstkamera (label B in Fig. 12), the visitor can walk (the distance is about 500 m). However, from Kunstkamera to the Museum of Karl May Gymnasium History, it is preferable to travel by car. In this case, the system finds an appropriate driver, who goes in this direction in this time and picks up the visitor (Fig. 13).

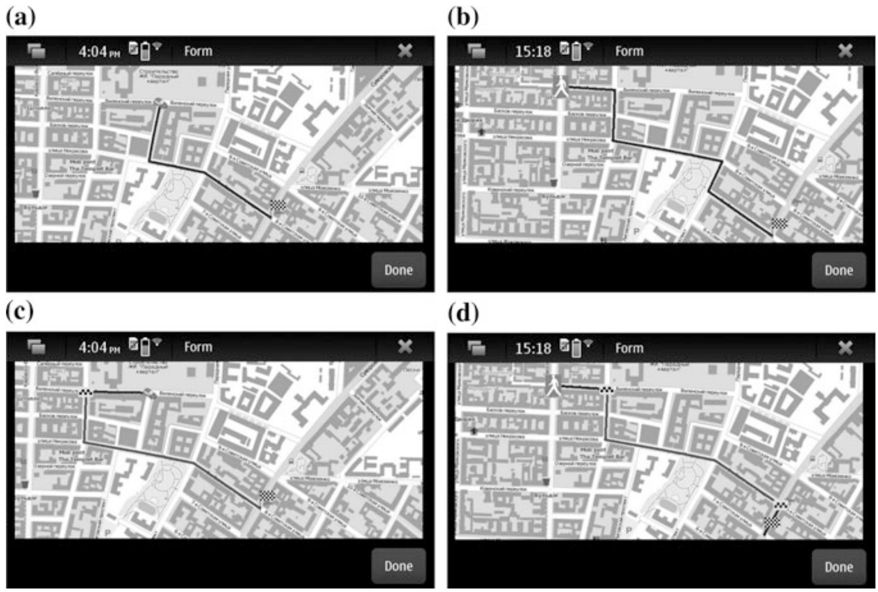


Fig. 11 Prototype screenshots (routes). **a** Driver's path without ridesharing. **b** Passenger's path without ridesharing. **c** Driver's path with ridesharing. **d** Passenger's path with ridesharing

Fig. 12 A sample museum itinerary on a visitor's mobile device

Fig. 13 A driver picks up a tourist near Kunstkamera and drives to the next museum

7 Conclusion

The chapter proposes a service-based approach to dynamic ridesharing based on the smart space concept and describes the system prototyping the approach, which provides an open-source solution for tourists in a certain region. The Smart-M3 information platform is used as a smart space infrastructure for the presented approach. Use of this platform makes it possible to significantly increase the scalability and extensibility of the system. The algorithm for finding appropriate fellow travelers for drivers as well as defining acceptable pick-up and drop-off points for them is presented The presented heuristics help to reduce the time of search by more than 1.5 times.

Projects that have been used in the proposed service are in the stage of active development. Particularly, the last updating of the pgRouting library provides for a 25% increase in overall system performance, such that the time for finding the shortest path between two points (at a workstation with an Intel Core i7-2700K 3.5-GHz processor, and 8 Gb of RAM DDR3) has been reduced from 160 ms to 120 ms.

Acknowledgments The chapter is a part of the research carried out within the ENPI project KA322 development of cross-border e-tourism framework for the program region (Smart e-Tourism); projects funded by grants # 12-07-00298-a, 11-07-00045-a, and 10-07-00368-a of the Russian Foundation for Basic Research; project # 213 of the research program "Intelligent information technologies, mathematical modeling, system analysis and automation" of the Russian Academy of Sciences; project 2.2 of the Nano- & Information Technologies Branch of the Russian Academy of Sciences; and contract # 14.740.11.0357 of the Russian program "Research and Research-Human Resources for Innovating Russia in 2009–2013".

References

Abrahamse W, Keall M (2012) Effectiveness of a web-based intervention to encourage carpooling to work: a case study of Wellington, New Zealand. In: Transport policy, vol 21, pp. 45–51

eRideShare.com (2012) http://erideshare.com. Last accessed on 05 June 2012

PickupPal (2012) http://www.pickuppal.com. Last accessed on 05 June 2012

Zimride (2012) http://www.zimride.com. Last accessed on 05 June 2012

RideshareOnline (2012) http://www.rideshareonline.com. Last accessed on 05 June 2012

Rideshare 511 (2012) http://rideshare.511.org. Last accessed on 05 June 2012

CarJungle (2012) http://www.carjungle.ru. Last accessed on 05 June 2012

Avego (2012) http://www.avego.com. Last accessed on 05 June 2012

OpenStreetMap (2012) http://www.openstreetmap.org/. Last access date 05 June 2012

Smart-M3 at Sourceforge (2012) http://sourceforge.net/projects/smart-m3, last access date 05.06.2012

Berners-Lee T, Fielding R, Masinter L (2012) RFC 3986—uniform resource identifier, Generic Syntax. http://tools.ietf.org/html/rfc3986. Last accessed on 05 June 2012

Smart-M3 at Wikipedia (2012) http://en.wikipedia.org/wiki/Smart-M3. Last accessed on 05 June 2012

Honkola J, Laine H, Brown R, Tyrkko O (2010) Smart-M3 information sharing platform. In: Proceedings of IEEE symposium computers and communications (ISCC'10). IEEE Comp Soc, June 2010, pp 1041–1046

Resource Description Framework (RDF) (2012) W3C standard. http://www.w3.org/RDF/. Last accessed on 05 June 2012

Smirnov A, Kashevnik A, Shilov N, Paloheimo H, Waris H, Balandin S (2010) Smart space-driven sustainable logistics: ontology and major components. In: Sergey B, Andrei O (eds) Proceedings of the 8th conference of open innovations framework program FRUCT, Lappeenranta, Finland, 9–12 Nov 2010, pp 184–194

Mapnik (2012) http://mapnik.org/. Last access on 05 June 2012

PostgeSQL (2012) http://www.postgresql.org/. Last access on 05 June 2012

PostGIS (2012) http://postgis.refractions.net/. Last access on 05 June 2012

pgRouting (2012) http://pgrouting.org/. Last access on 05 June 2012

Part IV
Algorithms and Computational Issues for GIS

Application of Sleator-Tarjan Dynamic Trees in a Monitoring System for the Arctic Region Based on Remote Sensing Data

Philipp Galiano, Mikhail Kharinov, and Sergey Vanurin

Abstract In this chapter, the monitoring system for the Arctic region, as used for automation of an ice situation assessment, is described. The main type of input data for the system is remote sensing of Earth observations. System improvement using Sleator-Tarjan dynamic trees is discussed. Elementary algorithms of considered tree-structured data are presented. A comparison of computing results with the results of interactive recognition executed by experts from the Arctic and Antarctic Research Institute is carried out. Further implementation development and the development of the system applications are analyzed.

Keywords Remote sensing · Image processing · Dynamic trees

1 Introduction

The Arctic is vulnerable region that is highly affected by the ongoing climate change. Current models predict that Arctic sea ice will disappear in the summer within 20 or 30 years, yielding new opportunities and risks for human activities in the Arctic. To meet the scope of challenges, which are constantly arising as a result of changing climate conditions, it is important to develop a geographic information system (GIS)-based monitoring system that allows various groups of users to store,

---

P. Galiano (✉) · M. Kharinov · S. Vanurin
St.Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS), St.Petersburg, Russia
e-mail: galiano@oogis.ru

M. Kharinov
e-mail: khar@iias.spb.su

S. Vanurin
e-mail: vanurin@gmail.com

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_9,

display, and process different data about the region according to their activities and needs. The list of these activities is increasing. In particular, it includes navigation, fishery, and mineral resources exploration/extraction. This newly emerged issue is well understood by policymakers/stakeholders around the world. An increasing number of research projects are focused on economical and social consequences of ongoing changes in the Arctic environment. Because of the changes, expected in the next 5–10 years in the Arctic environment, researchers are motivated by the necessity to develop effective high-tech tools that allow fast access to information on the state of key components of the Arctic environment: atmosphere (meteorological data), sea ice, and the ocean.

2 System Description

Each type of data may be used directly or as an input for further calculations:

- Meteorological data may be used to describe meteorological conditions and as input data for calculations of derivative parameters (e.g., energy exchange at the surface, wind current, ice drift, etc.).
- Oceanographic data are used to describe oceanographic conditions or as source data for calculations of derivative parameters (e.g., water density, sound velocity, geostrophic currents, etc.)
- Sea ice data may be used, researchers are motivated by ice conditions or as a discriminating parameter for distinction of ice-covered and ice-free areas.

The initial data to be acquired by the developed system from various sources are characterized by:

- Heterogeneity
- Incompleteness
- Errors of measurements
- Coherence

For each data type, there exist three stages of analysis:

- Initial (raw) data
- Processed data
- Results of data analysis

Heterogeneity of the data does not cancel a necessity of the general approach to task of their analysis, based on the results of oceanographic research from heterogeneous sources and concepts of harmonization, integration, and information fusion.

During the transformation from raw to aggregated (processed) data, the initial measurements obtained during various times and from various coordinates should be transformed for representation in a regular grid. For solving the specified task, one can interpolate and extrapolate raw data.

The distortions must also be rejected during raw data processing (error checking according to prescribed criteria). For rejection of bursts, methods of statistical processing should be used.

After obtaining the aggregated data, the analysis begins. The specific content of the analysis depends on specificity of the initial data and objects in view. In a simple case, the analysis may include:

- The calculation of statistics (mean field approximation and sample standard deviations)
- The calculation of implicitly defined characteristics (the sound velocity, the density of water, etc.)

In more complex cases (e.g., classification of ice types by value of compaction), if the obtained data permit, the recognition/classification and the forecast tasks may be executed. The classification process consists of two stages:

- Formations of feature space, often including segmentation for raster images.
- The subsequent classification of object patterns in the feature space. Thus, at the stage of creation of feature space, the set of used algorithms essentially depends on specificity of the data in hand.

Thus, for purposes of classification and recognition, it is necessary to:

- Develop and implement mathematical and simulation models for research and environmental monitoring of the Arctic region formations of feature space, often including segmentation for raster images.
- Develop and implement the models, methods, and algorithms for observation processing from various sources in the Arctic region.
- Develop models of data representation on the state of the ocean, atmosphere, and ice conditions. Development of algorithms for data integration within the framework of the developed decision support system is also needed.
- Develop methods and algorithms for statistical processing of raw data from each source.
- Develop models and algorithms for the analysis of the environment of the Arctic region based on the assimilation of oceanographic data and data on the ice state.

One of main data sources used in the system is Earth remote sensing data—satellite imagery, which are represented as monochrome bitmaps. The analysis process has several features that distinguish it from the analysis of point data from meteorological and oceanographic observations.

Initial processing of raster images is intended to reduce the impact of noise. It uses a wide range of image processing techniques (processing in frequency and spatial domain methods; methods, based on wavelet transform, and so on). Pre-processing depending on the peculiarities of the problem at hand. The characteristics of the source data are needed.

The second stage of the analysis is calculation of a set of properties—numerical characteristics of the image or the area suitable for further analysis. One of the methods used in the formation of feature space is partitioning the image into regions (segments) that satisfy the specified criteria. Thereafter for each segment, a computed set of properties (brightness, geometric, etc.) will be used to solve the problems of classification, prediction, and calculation of derived properties. To solve the problem of segmentation, there are extensive sets of different algorithms, but important characteristics of remote sensing data are a significant amount of the processed data, which limits the applicability of the common algorithms.

3 Application of Dynamic Trees

To calculate the features of image segments that are used for the detection of objects, the computed sequence of image partitions, which were obtained in the segmentation process, must be stored in random access memory (RAM). At the same time, storing the sequence of image partitions enables selection of the best one for current purpose; it also provides approximating of the objects with the segments from different partitions. However, storing multiple partitions in an ordinary matrix form requires too much memory. To overcome this challenge, so-called Sleator-Tarjan dynamic trees (Sleator and Tarjan 1983, 1985; Tarjan 1975) are irreplaceable. In modern notation, these trees are often referred to as “Disjoint sets,” “Persistent data structures,” or “Splay trees.” In terms of image processing, the first designation is justified by the fact that the dynamic trees support operations with disconnected pixels (Kharinov 2012a). The second designation emphasizes the ability to restore the data structure at any processing stage (Sarnak and Tarjan 1986; Kharinov 1993). The third designation reflects the restructuring of dynamic trees for faster computations (Tarjan 1985; Kharinov and Nesterov 1997).

In the field of image processing, Sleator-Tarjan dynamic trees (Sleator and Tarjan 1983, 1985; Tarjan 1975) first were implemented in SPIIRAS (Kharinov 1993; Kharinov and Nesterov 1997). At that time, they were treated as part of a complicated data structure for multilevel segmentation intended for an old-fashioned computer with extreme limited resources of memory, which were not enough even to store the original image. For modern computers, an image is processed as an active component of stored data, which allows us to reduce and simplify our version of tree-structured data (Kharinov 2012b). Image processing based on the Sleator-Tarjan dynamic tree has recently begun to gain momentum (Nock and Nielsen 2004).

3.1 Overview on the Implementation of Sleator-Tarjan Dynamic Trees

The data structure based on Sleator-Tarjan dynamic trees, hereafter simply called dynamic trees, was developed long ago (Sleator and Tarjan 1983, 1985; Tarjan 1975). However, only in recent years have they become interesting for complete implementation in image processing (Kharinov 2012b; Nock and Nielsen 2004). The structure of dynamic trees is an alternative to the four-connected lattice, which is used in the popular segmentation method of Graph Cuts (Boykov and Kolmogorov 2004). To laconically describe the segment, we use the formal tree-wise uniting of pixels instead of a lattice. Unlike conventional trees, additional nodes describing the union of pixels are not added in the structure of dynamic trees. Therefore, the required memory is reduced.

In this case, the connectivity of pixels in the segments persists as long as it is not affected by the operations with the segments. Therefore, if merging of segments is only used to generate the desired sequence from some initial partition of an image into connected segments, e.g., into separate pixels, then a connectivity of segments is supported automatically and the resulting partition sequence is hierarchical. Thus, for utilization of dynamic trees, it is preferable to program transactions with nonconnected pixels in the first place, considering, if necessary, the connectivity of pixels as a separate condition.

According to our present experience, the abilities of dynamic trees allow for the following options:

- Generation of a hierarchical (Kharinov 1993, 2012b; Kharinov and Nesterov 1997) or nonhierarchical (Kharinov 2012) sequence of partitions of an image into the sets of disconnected pixels, particularly into connected segments (hereinafter simply called segments).
- Storing of a hierarchy of multiple partitions in a fixed part of RAM (Kharinov 1993, 2012b; Kharinov and Nesterov 1997) and approximating objects with the segments from different partitions of that hierarchy (Kharinov and Nesterov 1997).
- Redistribution and filtering of segments in a hierarchical sequence of image partitions (Kharinov 2012b).

In the case of a nonhierarchical sequence of partitions, we have only started to develop the algorithms for generating of a sequence of overlapping partitions in terms of dynamic trees. However, we have already succeeded in utilization of dynamic trees for generation, storage, and also conversions of the hierarchy of digital image partitions. So, the implementation of dynamic trees in the real monitoring system based on memory-consuming remote sensing data seems timely.

To clarify the advantages of the implementation of the discussed tree-structured data, it may be sufficient to note that, owing to some complete set of algorithms (exemplified below by merging and concomitant procedures), the dynamic trees provide all of the capabilities of traditional trees but without the related excessive expenses of computer resources.

3.2 Elementary Algorithms

The idea of using dynamic trees for hierarchical image representation is quite simple. In terms of this tree-structured data, each segment is represented as a tree of pixels, and segment merging is described as merging of trees. The initial pixels are regarded as an elementary segment associated with an elementary tree, containing a single root node. To merge the trees, the root nodes are linked with each other by means of the arc. To restore the trees at some processing stage, an appropriate value of some index is assigned to each arc.

The structure of dynamic trees consists of three matrices of equal size: the original image; the matrix of arcs, which defines the formal structure of the tree; and the matrix of indices assigned to the arcs, which ensure the recovery of any partition from a hierarchy.

In the C programming language, the merging of trees is presented as follows:

```
BOOL TreeMerging
(int index,   int * Arcs, int * Indices, __int64* Additive, int a, int b)
{int ii, iii;
  for(ii = Arcs[a]; ii != Arcs[ii]; ii = Arcs[ii]);
  for(iii = Arcs[b]; iii != Arcs[iii]; iii = Arcs[iii]);
  if(ii < iii){ Arcs[iii] = ii ; Indices[iii] = index ; Additive[ii]+ = Additive[iii];
                return TRUE ;
               }
  else if(ii > iii){Arcs[ii] = iii ; Indices[ii] = index ; Additive[iii]+ = Additive[ii];
                return TRUE ;
               }
 return FALSE ;
}                                                                          ,   (1)
```

where *TreeMerging* is the Boolean function for merging of dynamic trees, specified by the nodes *a* and *b*, which return *FALSE* if *a* and *b* turn out the nodes of the same tree; *index* is the consecutive number of the processing stage or simply is the clock parameter; *Arcs* and *Indices* are the line-wise ordered matrix of arcs and corresponding matrix of indices; *Additive* is the line-wise ordered matrix of additive values assigned to pixels; and *ii* and *iii* are the variables, calculated as root nodes for given nodes *a* and *b*.

Arcs is initiated as $Arcs[i] \equiv i$, whereas *Indices* initially is zeroed, i.e., $Indices[i] \equiv 0$ for each pixel coordinate *i*. If *Additive* is initiated as a copy of the image, then for each segment the sum of pixel values will be accumulated in the root element of the array corresponding to the root node of appropriate tree. If *Additive* is initially filled with the one, i.e., $Additive\,[i] \equiv 1$, then in each root elements of the array and the number of pixels in the segments will be calculated. The values of other additive quantities are calculated in the same way, and the extreme values of segment attributes are calculated slightly differently. Online calculations of segment attributes are necessary to immediate account for their modifications during generation of a hierarchy. The recovery of any partition for the corresponding value of *index* is provided by the algorithm of "direct acess":

$$for(ii = 0;\ ii! = items;\ ii = Arcs[ii])\ if(Indeces[ii] < \,= index)curArcs[ii] = ii; \quad (2)$$

where *items* is the total pixel number in the image, and $curArcs[i]$ is the required matrix of arcs, initiated for each pixel coordinate i as the copy of source matrix of arcs $curArcs[i] \equiv Arcs[i]$.

For a given image partition, the dynamic trees are defined ambiguously and computed depending on the established linear-wise order of the pixel coordinates. This is not a challenge for the elementary algorithms (Kharinov 1993; Kharinov and Nesterov 1997). However, for further development of implementation software based on dynamic trees (Kharinov 2012a, b), it is useful to convert them into so-called "Directed Trees," in which all the pixels in each segment refer to the first one. This conversion is performed by the following algorithm:

$$\begin{aligned} for(ii = 0;\ ii! = items;\ ii = DirectArcs[ii])\ if(DirectArcs[ii] == ii)DirectArcs[ii] = ii; \\ else\ DirectArcs[ii] = DirectArcs[DirectArcs[ii]]; \end{aligned} \quad (3)$$

where *DirectArcs* is a required matrix of arcs for Directed Trees, initiated for each pixel coordinate i as a copy of current dynamic trees $DirectArcs[i] \equiv curArcs[i]$.

The latter algorithm provides filling segments with the values of previously calculated attributes or certain functions of attributes, computed for each segment. The calculation of additive attributes for each segment via pre-computed dynamic trees is performed by a simple procedure, which for vivid illustration we explain by two equivalent algorithms of calculation of the number of pixels for each segment.

The first algorithm for arbitrary dynamic trees is described by:

$$\begin{aligned} for(ii = items - 1;\ ii! = -1;\ ii - -)\ if(curArcs[ii]\ ! = ii)\{Members[curArcs[ii]] + = Members[ii]; \\ Members[ii] = 0; \\ \}, \end{aligned} \quad (4)$$

where *Members* is a required array of pixel number for each segment, initially filled with the one, i.e., $Members[i] \equiv 1$ for each pixel coordinate i.

The second algorithm, using Directed Trees, is described by:

$$for(ii = 0;\ ii! = items;\ ii + +)\ Members[DirectArcs[ii]] + +; \quad (4')$$

where *Members* initially is zeroed, i.e., $Members[i] \equiv 0$ for each pixel coordinate i.

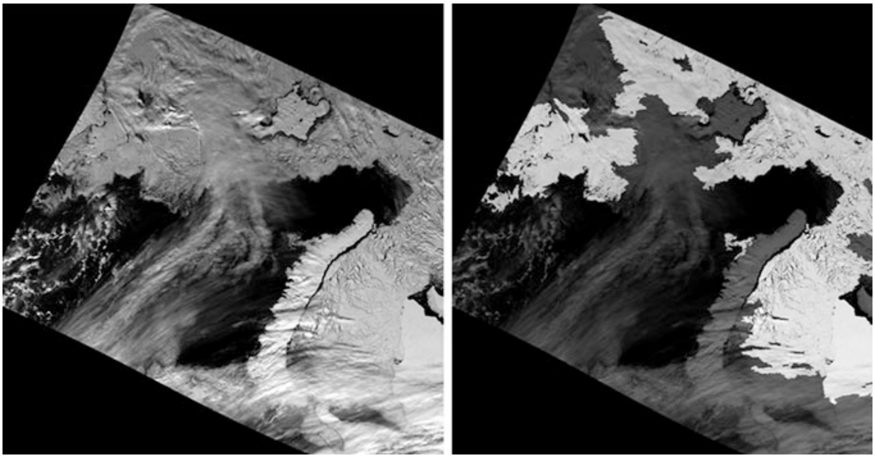


Fig. 1 Example of an ice situation estimation according to remote sensing data: input data—at the (*left*) and result of ice allocation (*right*). Areas of automatically allocated ice are marked by *white color*. *Image source*: Arctic and Antarctic Research Institute (http://www.aari.ru)

3.3 Case Study

To estimate the possibility of using dynamic trees for full-scale remote images, we use the SRM segmentation algorithm (Nock and Nielsen 2004), which is modified to provide the necessary speed. This algorithm allows one to carry out an image segmentation (one level of hierarchy) of the size 6,917 on 7,346 pixels (Fig. 1 at the left) in 10–20 s using the described data presentation structure. Different levels of hierarchy for storing in a dynamic tree data structure can be obtained by changing the value of SRM Q parameter (Nock and Nielsen 2004). The example of ice selection based on the MODIS satellite data for the Barents Sea region is given in Fig. 1. Spatial image resolution is 250 m.[1]

In feature analysis, the following three types of segment properties were used:

- Geometrical properties, e.g., square, perimeter
- Bright properties, e.g. average segment bright, variance of bright, minimal, and maximal bright of the segment
- Properties of neighborhood segments

Geometrical properties are generally useful but do not play key role for the ice recognition task. Bright properties are the main types of properties used for ice selection; however, for this type of property, it is difficult to obtain stability of classification results when survey conditions are changed. Normalization of the values scale, (i.e., conversion to relative properties values helps to solve a problem

[1] Due to memory restrictions for this test we use image, scaled to one quarter on its initial size.

Table 1 Recognition results for different property sets

Set of segment properties	Missed ice, %	Wrongly detected ice, %	Total correctly classified pixels, %	Total time of classification, seconds
Average brightness	02, 17	14, 87	82, 96	5
Added spectral properties	03, 13	09, 84	87, 04	8
Added variance of pixel brightness	03, 96	08, 06	87, 96	10
Added square, min and max bright, and average difference between color components in RGB color space	05, 33	05, 86	88, 81	11

partially); however, this leads to dependence of single-segment classification results from the properties of the whole image.

The final type of property is the frequency (i.e., a set of textural features). These features are based on spectrum shape. It could be some kind of wavelet transform, but the most obvious and common is the Fourier spectrum. The basic approach uses a Fourier transform of the square part of the image, moving this square window over the image. This method could be updated for segmented images using the additive characteristic of Fourier transform. Each segment is complemented to a square image by zero brightness pixels and a typical transformation is made. The arbitrary window size imposes some restrictions on the ability to compare two segments frequency by frequency, but using integral characteristics such as spectrum shape or its approximation by parametrized curves allows the use of this textural property as a segment property. For ice analysis, we use average spectrum value, variance of spectrum value, and ratio of spectrum square deviation to spectrum average.

In our study, a PC with Intel Core i7 CPU and 8 Gb of RAM was used. All software was written in the Java program language. For calculation of Fourier transform, we use the Parallel Colt programming library (https://sites.google.com/site/piotrwendykier/software/parallelcolt). For all property sets, we use the same training set, which consists of 19 points for ice segments; this explains why the missed ice percent monotonically increases when adding new segment properties (but total number of right classified pixels monotonically increases too) (Table 1).

4 Conclusions

In this chapter, we describe the results of the implementation of Sleator-Tarjan dynamic trees in applications for remote sensing data consisting of a large number of pixels compared to conventional images.

In the introductory section, we briefly described our software system designed for processing of remote sensing data, as well as the available meaningful recognition tasks that are performed in an ice situation assessment.

In the main section, we focused on the formal algorithms of generation and transformations of Sleator-Tarjan dynamic trees that improve analysis and recognition of full-scale remote sensing data. Compared with ordinary trees, they have a decisive advantage because they support the storing of any number of nested partitions of the image in a fixed part of RAM in accordance with algorithm (1). It should be noted that, in our opinion, a stumbling block for wide application of dynamic trees in image processing is the lack of a minimum set of basic algorithms that support all the features of conventional trees. A slight modification of the basic algorithms, particularly algorithm (1), leads to reprogramming of the entire image processing system. This chapter is likely just the beginning of this work. However, the strict conditions of full-scale image processing may facilitate an optimal solution.

The further development of remote data processing in terms of Sleator-Tarjan dynamic trees should consist of expanding the toolkit of presented elementary algorithms (1)–(4). Another way of developing a monitoring system for the Arctic region in applications for remote sensing data may use the results of solving of the so-called "segmentation problem," which we developed (Kharinov 2012a) based on the analytical generalization of the famous Mumford-Shah model (Mumford and Shah 1985, 1989; Bugaev and Khelvas 2001).

References

Boykov Y, Kolmogorov V (2004) An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. IEEE Trans Pattern Anal Mach Intell 26(9):1124–1137

Bugaev AS, Khelvas AV (2001) Pilot studies and development of methods and tools for analysis and automatic recognition of media flows in global information systems, Scientific and technical report, Moscow Institute of Physics and Technology (MIPT), Moscow, vol 1, p 140 (in Russian)

Kharinov MV (1993) Data structures of learning system for automatic image recognition, PhD thesis, SPb: SPIIRAS, p 20 (in Russian). See also http://www.dslib.net/vychislit-mashiny/azrabotka-dinamicheskih-struktur-dannyh-sistemy-avtomatizirovannogo-raspoznavanija.html

Kharinov MV (2012) Stable segmentation of digital image, computer graphics and vision (Graphicon'2012). In: Proceedings of the 22th international conference on computer graphics and vision, Moscow, MSU, 1–5 Oct 2012, pp 208–213 (in Russian). See also http://arxiv.org/abs/1208.2655v1

Kharinov MV (2012b) Adaptive dichotomous image segmentation toolkit. Pattern Recogn Image Anal: Adv Math Theor Appl 22(1):228–235

Kharinov MV, Nesterov MM (1997) Intelligent program for automatic image recognition based on compact object-fitting hierarchical image representation in terms of dynamic irregular ramified trees. International Academy, Vestnik. Special issue of MAISU bulletin no12-C, SPb, pp 1–35. (LC Control Number: 98646239-98646240)

Mumford D, Shah J (1985) Boundary detection by minimizing functionals. In: Proceedings of the IEEE computer vision pattern recognition conference, San Francisco, pp 22–26

Mumford D, Shah J (1989) Optimal approximations by piecewise smooth functions and associated variational problems. Commun Pure Appl Math XLII(4):577–685

Nock R, Nielsen F (2004) Statistical region merging. IEEE Trans Pattern Anal Mach Intell 26(11):1452–1458

Sarnak N, Tarjan RE (1986) Planar point location using persistent search trees. Commun ACM 29(7):669–679

Sleator DD, Tarjan RE (1983) A data structure for dynamic trees. J Comput Syst Sci 26(3):362–391

Sleator DD, Tarjan RE (1985) Self-adjusting binary search trees. J ACM 32(3):652–686

Tarjan RE (1975) Efficiency of a good but not linear set union algorithm. J ACM 22(2):215–225

Tarjan RE (1985) Sequential access in splay trees takes linear time: sequential access in splay trees takes linear time. Combinatorica 5(4):367–378

A Two-Phase Multiobjective Local Search for GIS Information Fusion: Spatial Homogeneity and Semantic Information Tradeoff

Enguerran Grandchamp and Evelin Fonseca-Cruz

Abstract This chapter deals with information simplification after the union of several information layers within a geographic information system (GIS). The challenge is to have better visualization (spatial homogeneity) while keeping as much information as possible (semantic information). These two objectives are opposing. Each layer has a set of concepts attached to an ontology, allowing the computation of a semantic distance used to select the information. Each object (called an instance) in a layer is annotated with the concept of the layer and also with its spatial information (shape, localization, etc.). The proposed approach uses a two-phase multiobjective local search in an ascendant way starting from the most complete set of concepts or in a descendant way starting from the most simplified set of concepts. In this chapter, we use this technique for environmental applications in order to determine ecological units based on environmental and topological layers. These units are used to identify isolated or threatened ecosystems in tropical forests. We compare the quality of the results and the computation time with other approaches, such as genetic algorithms.

Keywords GIS · Optimization · Multiobjective · Semantic information · Ontology · Map visualization

E. Grandchamp (✉) · E. Fonseca-Cruz
LAMIA, Université Antilles Guyane, Campus de Fouillole, Pointe-à-Pitre 97157 Guadeloupe, France
e-mail: egrandch@univ-ag.frenguerran.grandchamp@univ-ag.fr

E. Fonseca-Cruz
e-mail: evelin@uh.cu

E. Fonseca-Cruz
Havana University, Havana, Cuba

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_10,

1 Introduction

This study takes place in the general geographic knowledge discovery (GKD) field (Ded et al. 2002), which is an extension of knowledge discovery from database (KDD) to geographic information systems (GIS) (Tomlin 1990). In our context, the knowledge is both semantic information, structured with ontology of concepts, and geographic information, including the position and shape of the objects. The use of ontologies in GIS is nowadays wide spread since the achievement of an ontology-driven GIS [Fonseca and Egenhofer 1999].

In this chapter, we examine interesting information not within a unique layer that represents a thematic point of view of a territory, but rather within a layer obtained after the union of several layers (describing the same space with different points of view). In this context, the territory is partitioned as a lot of small objects with properties inherited from each layer. However, this information is too rich and too scattered to be understandable. The resulting layer must be simplified by keeping the essential information. So, the question is: How can we have interpretable information without losing too much information?

This kind of technique is applicable to environmental problems, such as localizing threatened ecosystems or identifying ecological units in order to map a territory to determine biodiversity. For this purpose, we can use information layers such as humidity, temperature, slope, elevation, ground, vegetation, etc.

This chapter is organized as follows: In Sect. 2, we provide a background overview to define the terminology, the notion of multiobjective optimization, and the objectives to be optimized. We explain in detail the resolution of the problem in Sect. 3, with a complexity analysis of the problem that justifies the two-phase local search. Section 4 presents the experimental results and a comparison to solutions obtained using a genetic algorithm. Finally, Sect. 5 gives the conclusion and perspectives of the study.

2 Background and Definitions

2.1 Terminology and Modeling

Inspired by Tomlin's map algebra (Jadaan et al. 2008) as applied to semantic and vector information, the problem is expressed as follows: let $E = \{(L_1, O_1), (L_2, O_2) \ldots, (L_n, O_n)\}$ be a set of n pairs where L_i is an information layer and O_i the associated ontology. In our case, an ontology is a set of concepts structured in a tree $Oi = \left(\left\{C_{i,1}, \ldots, C_{i,n_i}\right\} T_i\right)$ where n_i is the number of concepts of O_i, $C_{i,j}$ the jth concept of O_i and T_i the corresponding tree. The $C_{i,j}$ are the leaf concepts of the tree (the more specific ones). Each node of the tree (which could also be considered as a concept) is a generalization of its son nodes. The roots node represents the more abstract concept.

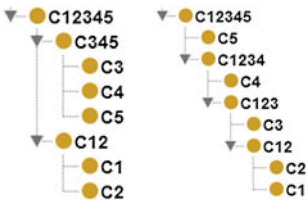


Fig. 1 Ontology examples

We can have different ontologies for the same layer (different concepts and different trees). Figure 1 presents two example trees for the same concepts $\{C_1, C_2, C_3, C_4, C_5\}$: the first one has different possible generalizations for the same level (left) and the other one authorizes only one generalization per level (right). The choice of the ontology will have an impact on the results of the information selection process.

In a layer, each object is annotated with one of the concepts. The object is called an instance of the concept. Each instance is also annotated with its geographical information (position, shape, surface, shape length). We denote $I_{i,j} = \left\{I_{i,j}^1, \ldots, I_{i,j}^{n_{i,j}}\right\}$ the instances of the concept $C_{i,j}$. Figure 2a illustrates the instances of a concept C_1. In the scope of this study, we only consider layers representing a partition of the territory.

Two instances are adjacent if they share the same border. Figure 2b shows an example of an adjacency graph.

The features attached to each concept are the mean surface of the instances, the number of instances, and the instances themselves.

After the union of several layers, and the union of the instances leads to smaller instances annotated with one concept per used layer. Each instance I_k of the resulting layer is then annotated with a vector of concepts $\left\{C_{1,k_1}, \ldots, C_{n,k_n}\right\}$, $k_i \in [1, n_i]$ (Segretier et al. 2012).

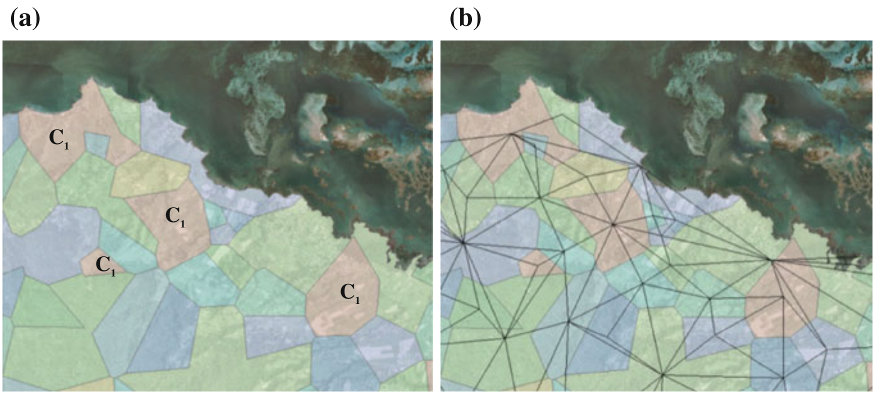


Fig. 2 Layer definition. **a** Instances of a concept C_1. **b** Adjacency graph

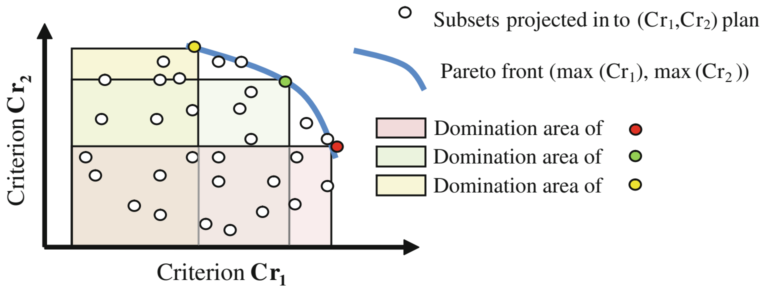


Fig. 3 Multiobjective optimization and Pareto front

2.2 Multiobjective Optimization

Research on the optimal aggregation of concepts is based on multiobjective optimization. We review here some basic principles of multiobjective optimization; more details can be found in Jadaan et al. (Liefooghe et al. 2011).

A multiobjective optimization algorithm aims to simultaneously improve several criteria during the search. Without loss of generality, we consider in this example the maximization of two criteria Cr_1 and Cr_2 (Fig. 3). Most of the time, it is not possible to reach a solution that optimizes all criteria (and often the solution does not exist). In these cases, we make a trade off between the criteria and have to choose among different kinds of solutions that each optimizing a different criterion. The set of all incomparable solutions is called the non-dominated set or Pareto set.

The Pareto set is formally defined for two criteria as follows:

- Each solution is evaluated for two criteria Cr_1 and Cr_2
- A solution S_1 dominates a solution S_2 according to Cr_i if $Cr_i(\mathrm{S}_1) > Cr_i(\mathrm{S}_2)$. $i = 1$ or 2
- A subset S_1 dominates a subset S_2 if $\forall i Cr_i(\mathrm{S}_1) \geq Cr_i(\mathrm{S}_2)$ and $\exists i | Cr_i(\mathrm{S}_1) > Cr_i(S_2)$.
- A subset S_1 is not dominated if $\nexists S_2 | S_2$dominates $\mathrm{S}_1 (\nexists S_2 | \forall i Cr_i(S_2) \geq Cr_i(\mathrm{S}_1)$, and $\exists i Cr_i(S_2) > Cr_i(\mathrm{S}_1))$.
- The set of all nondominated subsets is called the Pareto set.

According to this definition, Fig. 3 illustrates the templates of the Pareto front. All the solutions in the Pareto front optimize the criteria in a different way, which allows the research algorithm to return a set of solutions instead of a unique one. This technique will be twice used as the two-phase multiobjective local search.

2.3 Objectives

2.3.1 Visualization Improvement

To satisfy the main objective, which is to simplify the representation of the information after the union of several layers in GIS, we have to maximize the mean surface of the instances of the aggregate concepts. This first criterion is expressed as follows:

$$S(E) = \frac{1}{NI} \sum_{k=1}^{NI} I_k \cdot Surf,$$

where NI is the number of instances and $I_k \cdot Surf$ is the surface of the instance I_k

This criterion expresses the complexity of the information: the lower is the mean surface, whereas the higher is the complexity of the data (in a spatial point of view). According to this criterion, the best solution is the one that aggregates all the concepts in each layer, leading to a unique object (instance of the unique aggregated concept) for which the surface is maximal (it covers the whole territory).

2.3.2 Information Keeping

The map produced after the union of the layers also must contain as much information as possible. This leads to the maximization of a second criterion, which is the number of concepts after the aggregation. This criterion is expressed as follow:

$$N(E) = \sum_{i=1}^{n} n'_i,$$

where n'_i is the number of remaining concepts in layer I after the aggregations.

This criterion translates the accuracy of the simplified representation. According to this criterion, the best solution is the one that keeps all the initial concepts in each layer without any aggregations. The corresponding mean surface of the instances ($S(E)$) is then the lowest one possible.

However, these two objectives are opposite. The reduction of the number of concepts always leads to the union of the corresponding instances and then to an increase of the average surface (at least keeping the same value).

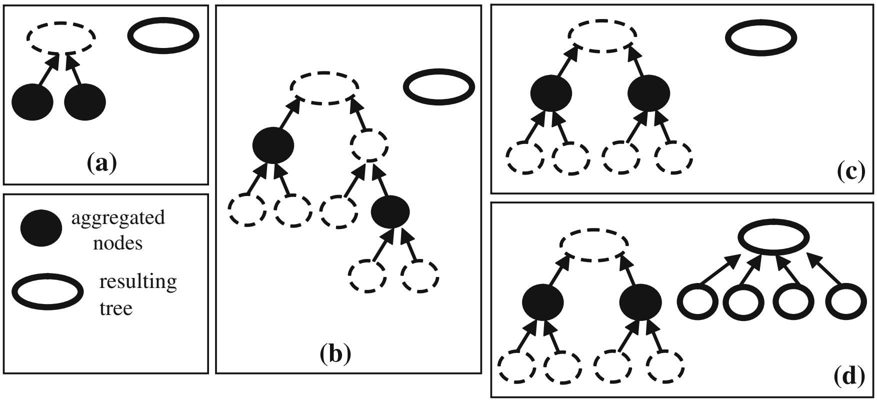


Fig. 4 Aggregation: semantic point of view

3 Resolution of the Problem

3.1 Aggregation of Two Concepts

The aggregation of two concepts, whatever the concepts, decreases $N(E)$ and increases $S(E)$.

3.1.1 Semantic Point of View

In a semantic point of view, we address the aggregated concepts. To build the hierarchy of concepts, the following rules are not compulsory but recommended.

The ontology should be a tree and not a graph.

At an iteration k:

- The aggregation of two terminal concepts (leaf nodes in the ontology) that have the same parent results in their parent becoming a terminal concept (Fig. 4a).
- The aggregation of two concepts (nodes in the ontology) that are not leaf nodes must be made with caution. In the case of two children of the same node, there are two ways to aggregate the concepts. In the first approach, the resulting subtree is a node inheriting the children of the aggregated nodes (Fig. 4d). In the second approach, we only keep the parent of the concepts (Fig. 4c). In the case of two random concepts in the ontology, we must aggregate the parents up to aggregate a common one (Fig. 4b).
- The aggregation of two terminal concepts that have different parents joins the second case of the previous item.

When two concepts with different parents are aggregated, the resulting ontology is either semantically inconsistent or the number of concepts in the ontology can be enormously reduced.

In our case, we want to iteratively build the Pareto front and keep one solution for each number of concepts. Therefore, we limit the authorized aggregations to terminal concepts with the same parent (Fig. 4a). This process is called a hierarchical aggregation (Miller and Han 2009).

3.1.2 Spatial Point of View

In a spatial point of view, the aggregation leads to the union of geographical objects (the instances of the concepts) and then to the modification of their shape and other geographical properties (surface, etc.). The aggregation of two concepts modifies the objects only if two instances of the concepts are adjacent. Figure 5 illustrates the aggregation of two concepts showing the modification of the instances: the instances of two concepts C_1 (pink) and C_2 (blue) before the aggregation are shown on the left, whereas the instances of the new aggregated concept C_{1-2} (purple) are shown on the right.

I_1^4 (left) and I_{1-2}^4 (right) are identical because this instance of C_1 does not share any border with an instance of C_2. On the contrary, I_2^2 and I_1^3 are unified into $I_{(1-2)}^2$ because they share a common border.

3.1.3 Features Computation

Each instance is annotated with a concept, its surface (computed with geometric tools within GIS), its shape, and position. Each concept is annotated with the surface (which is the sum of the instances' surfaces), the number of instance's, the instances and the mean surface.

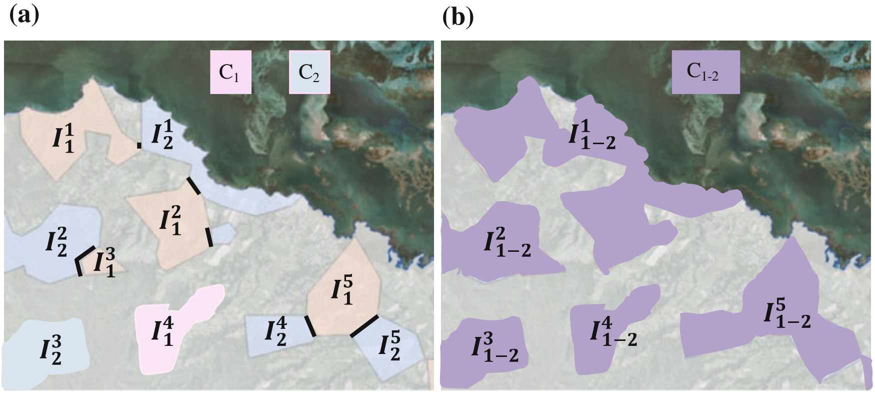


Fig. 5 Aggregation: spatial point of view. **a** Before aggregation. **b** After aggregation. I_i^k: k^{th} instances of concept C_i

Table 1 Objective computation after aggregation

Concepts	C_1					C_2				
Instances	I_1^1	I_1^2	I_1^3	I_1^4	I_1^5	I_2^1	I_2^2	I_2^3	I_2^4	I_2^5
Surface (km^2)	5.3	4.8	0.9	4.7	5.0	5.1	5.2	5.1	2.1	2.3
Mean surface		$S(E) = 4.05$			Nb. Concepts			$N(E) = 2$		

Concepts	C_{1-2}				
Instances	I_{1-2}^1	I_{1-2}^2	I_{1-2}^3	I_{1-2}^4	I_{1-2}^5
Surface (km^2)	15.2	6.1	5.1	4.7	9.4
Mean surface	$S(E) = 8.1$		Nb. Concepts		$N(E) = 1$

After aggregation, the surface of the new concept (C_{1-2}) is the sum of the surfaces of each concept C_1 and C_2, the number of instances, and the instances themselves that required geo-processing treatments. Table 1 shows the numerical evolution of the features (surface) and criteria ($N(E)$, $S(E)$) for the example presented in Fig. 5.

3.2 Complexity of the Problem

3.2.1 Exploration Space

The problem is a combinatorial problem that could not be solved with an exhaustive search. Indeed, consider n layers ($L_1, \ldots L_n$) with ($n_1, \ldots, n_n$) concepts, respectively. For each layer, the minimum number of possible aggregations is 1 and the maximum number is the Bell number (Miller and Han 2009) (Eq. 1.)

$$B_n = \sum_{k=0}^{n-1} C_k^n \cdot B_k \tag{1}$$

where C_k^n is the number of possible subsets of size k among n elements.

The Bell number is the number of partitions of a set of size n. A partition of a set S is defined as a set of nonempty, pairwise disjoint subsets of S whose union is S.

If we do not use an ontology tree to structure the concepts (no semantic relations between concepts), the size of the exploration space is $\prod_{k=1}^{n} B_n$. As an example, for 10 layers with 10 concepts, the size of the exploration space is $(B_{10})^{10} = 21147^{10} \approx 2.10^{50}$.

When arranging concepts, as shown in the right part of Fig. 1 (which is an advised initial arrangement), the number of aggregations is equal to the number of concepts. In this case, the size of the exploration space is $\prod_{k=1}^{n} n_k$. With the same example of 10 layers each having 10 concepts, the size of the exploration space is

10^{10}. In the general case, the size of the exploration space depends on the tree of the concepts.

In our applications, we use simplified known ontologies because we only keep useful concepts without cycles in the relations. When possible, we build a new ontology from the concepts by ranking them according to their proximity in order to have a tree with the same shape as in Fig. 1a. The choice of the ontology is important and guides the way we reorganize concepts.

3.2.2 Computation Time

The evaluation of the mean surface of the objects takes several seconds on a fast computer. Indeed, the computation requires polygon intersection computation and surface computation of hundreds of thousands polygons. If it only requires 1 s per evaluation, it will require more than 300 years to explore the whole exploration space. A exhaustive exploration is then forbidden.

3.3 Local Search Approach

To reduce the number of solutions explored, we chose a local search that deals with a unique solution that evolved with time. The successor of the current solution is chosen within a neighborhood. It is composed of all accessible solutions from the current solution, which includes all the solutions obtained after an aggregation of the authorized concepts, according to rules presented in Sect. 3.1.

In an algorithm iteration, there are too many possible aggregations to compute in an acceptable time. Therefore, the neighborhood must be reduced, but its composition has a great influence on the convergence of the search to an optimal or suboptimal solution. Taking these remarks into account, we reduce the computation time by selecting between the neighbors for a pool of interesting solutions based on two decision criterions: a fast semantic distance and a fast spatial distance. Then, we keep the Pareto optimal solutions from the decision space (ensuring quality) and project them into the objective space to compute the mean surface, keeping the best one.

3.3.1 Two-Phase Local Search Algorithm

By the analysis of the objective functions, we remark that one of them, $N(E)$, takes discrete values from 1 to $N = \sum_{k=1}^{n} n_k$. In this case, the Pareto front in the space $(S(E), N(E))$ contains at most N values. It contains exactly N values if we keep one solution per number of concepts. However, in the general case, if an aggregation does not increase the mean surface, then the corresponding solution will not be in the Pareto set and the Pareto front will not have a solution for this number of concepts.

With such particularities, we adopt a two-phase multiobjective local search, which will progressively shape the Pareto front by starting from one extreme solution (i.e., best solution regarding one of the objectives, such as $N(E)$) and by degrading the solution regarding this objective and improving it regarding the other one ($S(E)$). With the two-phase multiobjective local search, we select a pool of Pareto solutions in the objective space in $N = \sum_{k=1}^{n} n_k$ iterations.

The algorithm is summarized as follows:

1. Start with the solution composed of all the concepts ($N(E)$ maximized, $S(E)$ minimized).
2. Compute the semantic (Sd) and spatial distances (Sp) between all pairs of concepts that respect the authorized aggregations (fast).
3. Select the aggregations within the Pareto front of this decision space (it considerably reduces the number of pairs and ensures quality of the pairs).
4. Compute the mean surface $S(E)$ for each of the previous aggregations (each evaluation is slow but the number of evaluation is reduced).
5. Select the aggregation with the highest mean surface, apply this aggregation to reach the new solution.
6. Restart at 2, until $N(E) = 1$.

Step 2 considerably reduces the computation time and step 3 reduces the size of the neighborhood before computing the objectives (step 4).

At step 5, if the aggregation is not Pareto optimal in the objective space (no improvement of the mean surface due to nonadjacency of the instances of the aggregated concepts), we still keep the solution in the pool of interesting solutions because it could lead at the next iteration to an aggregation by improving the mean surface.

3.4 Semantic Distance

The semantic distance (Sd) is based on the ontology and measures the distance between two concepts. At the beginning of the process, the semantic distances between two leaf concepts are given by experts in the field of the layer. It must indicate which concepts are closer than others. After an aggregation, we have to compute the distance between one aggregated concept and the other ones (aggregated or not). In this case, we must define some rules to compute the new distances.

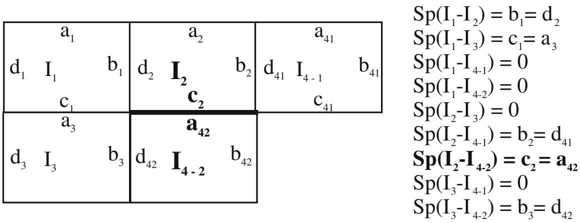


Fig. 6 Spatial proximity between instances

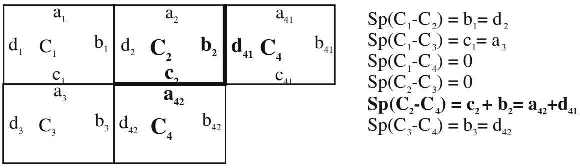


Fig. 7 Spatial proximity between concepts

The first rule directly derives from Sect. 3.1: we only authorize the aggregation of concepts with the same parent. So, the *Sd* between two concepts with different parents must be high.

The second rule aims to favor the aggregation of detailed concepts instead of general ones. The *Sd* between two aggregations could be the sum of the distance between each pairs of concepts, in order to increase the distance each time we aggregate them.

3.5 Spatial Distance

The spatial distance, or spatial proximity (*Sp*), between two instances (objects) is the sum of the length of the shared border between these instances.

Figure 6 illustrates the computation of *Sp*. The *Sp* between two concepts is the sum of *Sp* between each pair of instances (one instance belonging to the first concept, the other to the second one). Figure 7 illustrates the computation of *Sp*.

Figure 8 illustrates the decision space with the *Sd* and the *Sp*.

3.6 Computation of the Distances

At the beginning of the algorithm, we initialize the *Sp* and *Sd* by computing the union of the layers. We use geoprocessing and adjacency computation, which are time consuming but performed only one time. After each aggregation of two or more concepts, the computation of the *Sp* does not require a new spatial adjacency

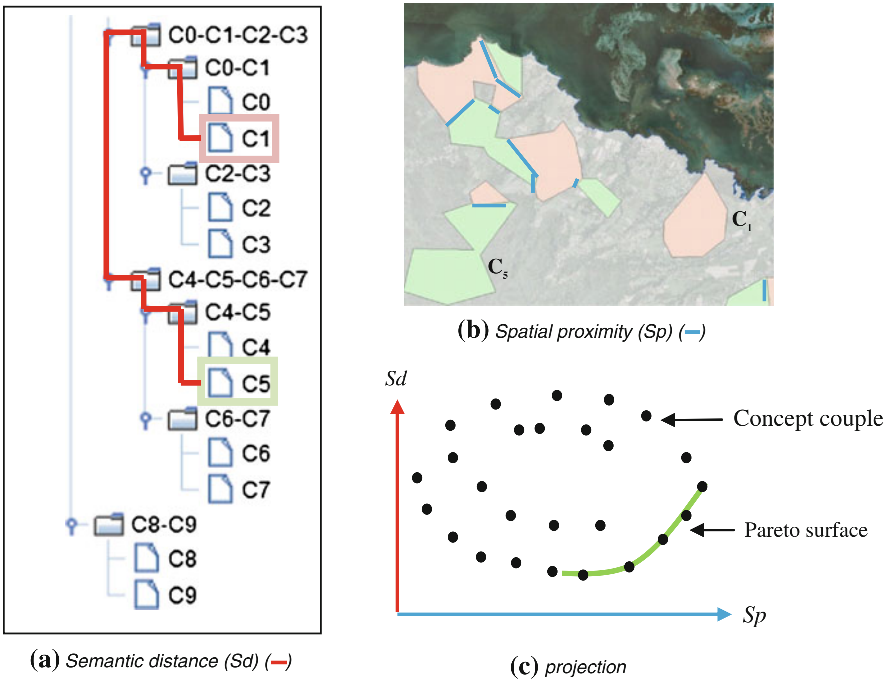


Fig. 8 Semantic (*Sd*) and spatial (*Sp*) distances

computation. As an example, $Sp(C_1 \cup C_2, C_k) = Sp(C_1, C_k) + Sp(C_2, C_k)$, $k \neq 1, 2$.

In the same way, the computation of the new *Sd* does not require time-consuming geoprocessing. $Sd(C_1 \cup C_2, C_k) = Sd(C_1, C_k) + Sd(C_2, C_k)$, $k \neq 1, 2$.

3.7 Two-Phase Local Search: Distance Space and Objective Spaces

Figure 9 illustrates how to project Pareto solutions from the decision space (*Sp, Sd*) to the objective space (*S*(*E*), *N*(*E*)) and how to select the best one (in green in the figure). Then, by applying successive aggregations, the two-phase local search shapes the Pareto front in the objective space (following the blue arrow's path).

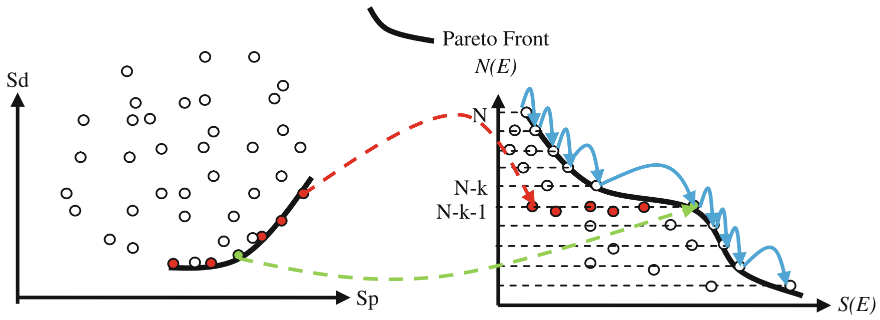


Fig. 9 Decision space and objective space

4 Experiments

The first part of the experiment presented in this chapter is made with simulated data in order to compare the results to the optimal solution. The quality of the result and the computation time are compared with the multiobjective genetic algorithm (*MOGA*) approach (Miller and Han 2009; based on the same criteria). Details on *MOGA* can be found in Noller and Smith (1987). *MOGA* is parameterized with the following values: the size of the population is set to 100, the number of generations is set to three times the number of concepts, the probability of crossover is 0.7, and the mutation probability is 0.001. This parameterization arises from the best results of Miller and Han (2009). The two-phase multiobjective local search is referred as *2PMOLS* in this section.

4.1 Computation Time Analysis on Simulated Data

We compare both *2PMOLS* and *MOGA* approaches with and without the decision space. With the decision space, the aggregations are sorted and ranked using *Sp Sd*; only Pareto optimal pairs are evaluated in the objective space (number of concepts, mean surface of the instances). Without the decision space, all explored aggregations are evaluated using the objective space. The trees of concepts are generated randomly.

As shown in Table 2, the use of the decision space considerably reduces the computation times with both approaches. The decision criterion takes less than 100 ms to be evaluated, while the mean surface computation takes around 30 s. Moreover, *2PMOLS* with and without the decision space is faster than *MOGA* and could be run in an acceptable time (a few hours at most for *2PMOLS* instead of several days for *MOGA*).

Table 2 Computation time comparison (*2PMOLS*/*MOGA*)

Number of layers/concepts/instances	Computation time: global/evaluation of *Sp* and *Sd*/evaluation of the *N(E)* and *S(E)*			
	2PMOLS		*MOGA*	
	With decision space	Without decision space	With decision space	Without decision space
3/30/3,000	45 min/2 s/45 min	1.5 h/–/1.5 h	2.2 h/2 min/2.2 h	7 day/–/4 day
5/50/10,000	1.2 h/5 s/1.2 h	4.1 h/–/4.1 h	3.8 h/4 min/3.8 h	7 day/–/7 day
10/100/100,000	2.5 h/20 s/2.5 h	16 h/–/16 h	7.6 h/7 min/7.6 h	14 day/–/14 day
15/150/300,000	4 h/45 s/4 h	37 h/–/37 h	11.1 h/10 min/11.1 h	Na/–/Na
20/200/500,000	5 h/80 s/5 h	2.7 day/–/2.7 day	15.2 h/14 min/15.2 h	Na/–/Na

2PMOLS, two-phase multiobjective local search; MOGA, multiobjective genetic algorithm; NA, not available.

4.2 Quality Analysis on Simulated Data

Table 3 presents a quality measure of the two approaches. Because both approaches return a pool of nondominated solutions (according to the explored solutions during the search), we compare the sets by counting for each approach (*2PMOLS* and *MOGA*) the number of solutions, obtained in the same conditions (with or without decision space), that dominates at least one solution of the other approach. We do the same with the solutions that are not dominated or that are dominated. Two identical solutions are considered as not dominated and not dominant. We remark that *2PMOLS* solutions often dominates *MOGA* ones. The lower quality of the solution returned by *MOGA* is probably linked to the fact that at an iteration of *MOGA* we only explore (with or without decision space) the aggregations

Table 3 Quality comparison of the returned solutions (*2PMOLS*/*MOGA*)

Number of layers/concepts/instances	(Number dominant, not dominated, dominated)			
	2PMOLS		*MOGA*	
	With decision space	Without decision space	With decision space	Without decision space
3/30/3,000	**(16, 11, 3)**	**(21, 9, 0)**	(2, 31, 67)	(0, 26, 74)
5/50/10,000	**(28, 21, 1)**	**(34, 14, 2)**	(3, 25, 72)	(2, 27, 71)
10/100/100,000	**(67, 33, 0)**	**(71, 29,0)**	(0, 33, 63)	(0, 31, 69)
15/150/300,000	**(124, 24, 2)**	**(0, 150, 0)**	(3, 37, 60)	Na
20/200/500,000	**(142,65,3)**	**(0, 200, 0)**	(2, 42, 56)	Na

2PMOLS, two-phase multiobjective local search; MOGA, multiobjective genetic algorithm.

Table 4 Quality comparison of *2PMOLS* with and without decision space

Number of layers/concepts/instances	(Number dominant, not dominated, dominated)	
	2PMOLS	
	With decision space	Without decision space
3/30/3,000	(0, 22, 8)	(8, 22, 0)
5/50/10,000	(0, 40, 10)	(10, 40, 0)
10/100/100,000	(**5**, **82**, 13)	(13, 82, 5)
15/150/300,000	(**9**, **130**, 11)	(11, 130, 9)
20/200/500,000	(**12**, **179**, 9)	(9, 179, 12)

2PMOLS, two-phase multiobjective local search; MOGA, multiobjective genetic algorithm.

accessible with a crossover between two individuals of the current population; however, with *2PMOLS,* the neighborhood of the current solution is composed of all possible aggregations. This leads to a better approximation of the Pareto front in the decision space in the case of *2PMOLS* and then better solutions as evaluated in the objective space.

Table 4 shows the same analysis for *2PMOLS* with and without decision space. Most of the solutions obtained with a decision space are not dominated (80–90%); that is, that they are equivalent to one solution obtained with the objective space but in a shorter time. Moreover, some aggregations reached with the decision space are dominant.

4.3 Visual Analysis on Real Data

Figure 10a shows the valuated adjacency graph between the instances. The value of an edge is the length of the shared border between the concerned instances. This graph is the basis to compute the *Sp* and is computed only one time. Figure 10b

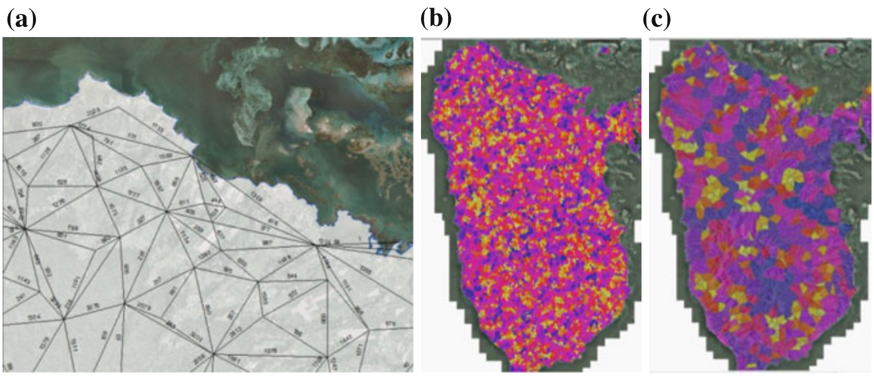


Fig. 10 Visualization of one solution for 3 layers, 30 concepts, and 3,000 instances. **a** Valuated adjacence graph. **b** Before aggregation. **c** After aggregation (17 concepts*)*

shows the initial map obtained with the union of 3 layers each having 10 concepts ($N(E) = 30$, $S(E) = 0.78$ km^2) and Fig. 10c shows one of the maps returned by *2PMOLS* algorithm and having 17 concepts ($N(E) = 17$) for 148 instances ($S(E) = 15.81$ km^2).

5 Conclusion and Perspectives

We present in this chapter a way to optimize the selection of relevant information after the union of several layers for a spatial and a semantic objective. The use of a two-phase local search and a decision space with two fast criterions (*Sp* and *Sd*) allows one to reach a good solution in an acceptable time. The algorithm outperforms previous approaches using genetic algorithms because it gives better results in a shorter time. We are now working on a way to completely avoid the use of the objective space (only using decision space) to reach the Pareto front in order to have a very fast algorithm (less than 1 min). The applications to environmental problems show useful results and maps. Nevertheless, the method could be improved by using local aggregation for the concepts. In this way, the instances of two concepts could be aggregated in an area of the map (for example, if the spatial arrangement of the instances in this area is complex) and the concepts could be kept separate in an other area. This will lead to the existence of different levels of abstraction of the same notion on the map, which has to be managed with care. We are working on a derived form of the algorithm to integrate a dynamic abstraction of the concepts, taking into account local criteria. Finally, we are working on a way to limit the influence of the ontology by transforming any ontology into a standard shape based on *Sd* between the concepts.

References

Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE Trans Evol Comput 6:182–197

Fonseca FT, Egenhofer MJ (1999) Ontology-driven geographic information systems. In: Proceedings of the 7th ACM international symposium on advances in geographic information systems, ser. GIS'99. New York, ACM, pp 14–19, [Online]. Available: http://doi.acm.org/10.1145/320134.320137. Retrieved Nov 2011

Jadaan OA, Rajamani L, Rao CR (2008) Non-dominated ranked genetic algorithm for solving multi-objective optimization problems: NRGA, J Theoret Appl Inf Technol, pp 60–67

Liefooghe A, Paquete L, Simoes M, Figueira JR (2011) Connectedness and local search for bicriteria knapsack problems, EvoCOP 2011, LNCS 6622, pp 48–59

Miller HJ, Han J (2009) Geographic data mining and knowledge discovery, 2nd edn. CRC Press, Boca Raton

Noller C, Smith VR (1987) Ultraviolet selection pressure on earliest organisms. In: Kingston H, Fulling CP (eds) Natural environment background analysis. Oxford University Press, Oxford, pp 211–219

Segretier W, Collard M, Grandchamp E (2012) An heuristic-based approach for merging information layers in a GIS, Geoprocessing

Tomlin CD (1990) Geographic information systems and cartographic modelling. Prentice-Hall, Englewood Cliffs

Parallel Algorithms of Discrete Fourier Transform for Earth Surface Modeling

Marina Chicheva

Abstract One of the important problems in the use of remote sensing from satellites is three-dimensional modeling of surface—fragments both dynamic (e.g., ocean surface) and slowly varying ones. Some researchers propose the use of methods that are based on the discrete Fourier transform (DFT). For such an approach in the dynamic case, the time of transform implementation is critical. By decreasing this length of time, one can increase the modeled fragment size, thereby improving quality and model authenticity. Multiprocessor systems and graphic processors allow the use of special technology for parallel computations to decrease implementation time. This chapter presents the published approaches, as well as the author's approach, to DFT parallel algorithms for this particular problem.

Keywords Remote sensing · 3D modeling · Discrete fourier transform · Parallel algorithms

1 Introduction

One of the important problems in the use of remote sensing from satellites is three-dimensional (3D) modeling of surface fragments—both dynamic and slowly varying ones. The large volume of processing data leads to lengthy computations; one of the ways to decrease the implementation time is parallelization. Many researchers have synthesized parallel algorithms for different problems, such as the processing of high-resolution spatial data (Widenera et al. 2012; Yinab et al. 2012) or computing visibility information on digital terrain models (Floriania et al.

M. Chicheva (✉)
Image Processing System Institute of RAS, Molodogvardejskaya St. 151,
Samara, Russian Federation443001,
e-mail: mchi@smr.ru

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_11,

1994). Some researchers proposed methods that are based on the discrete Fourier transform (DFT). Consider the dynamic case example of 3D modeling of water (ocean) surface, which is a typical problem in oceanology, 3D games, and other interactive applications. A statistical method (Tessendorf 2009) based on DFT has been used for such dynamic model constructions. For slowly varying fragments of Earth, there also are some problems for which the time of DFT is important, such as 3D shape retrieval (Frejlichowski 2011) based on processing two-dimensional (2D) projections with DFT. Another approach based on a covariance matrix calculated by means of DFT was described by Guan et al. (2011), who used parallel computations for similar parts in a geostatistical areal interpolation method. However, an additional speed-up can be achieved with parallelization of DFT.

For these approaches, the time of transform implementation is critical. By decreasing this time, one can increase the modeled fragment size, thus improving quality and model authenticity. This chapter presents the application of parallel algorithms of DFT for this particular problem and research regarding their efficiency.

2 Examples of 3D Modeling Methods

This section contains examples of the previously mentioned 3D modeling methods. Because the influence of DFT implementation time is more apparent in the dynamic case, the discussion focuses on 3D modeling of water surface.

2.1 Statistical Approach to Water Surface Fragment Forming

Algorithms of water surface fragment forming have been developed and successfully applied, such as Gerstner's method (Henry 2008), physical models (Finch 2004), and the summary of sinusoidal waves (Isidoro et al. 2002). This section considers Tessendorf's model (Tessendorf 2009), which is based on DFT.

The initial spectrum of the water surface is formed by the formula:

$$\bar{h}_0(k_1, k_2) = \frac{1}{\sqrt{2}}(\xi_{\text{re}} + i\xi_{\text{im}})\sqrt{P_h(k_1, k_2)},$$

where ξ_{re} and ξ_{im} are realizations of Gaussian random variables; $k_1 = \frac{2\pi n_1}{L_1}$, $k_2 = \frac{2\pi n_2}{L_2}$ are spatial coordinates; L_1, L_2 are sizes of the water surface fragment; n_1, n_2 are pixel numbers; and $P_h(k_1, k_2)$ is the Phillips spectrum. The latter is defined by:

$$P_h(k_1, k_2) = A\,\frac{\exp\left\{-\frac{1}{(k_1^2+k_2^2)L^2}\right\}}{\left(k_1^2 + k_2^2\right)^2} \cdot |k_1 v_1 + k_2 v_2|,$$

where $L = \frac{V^2}{g}$ is the maximal possible wave appearing due to continuous wind with speed $V = \sqrt{v_1^2 + v_2^2}$; v_1, v_2 are components of direction and power of wind; and A is the normalizing coefficient.

Then, for water surface animation, the next frames are defined by the relation:

$$\bar{h}(k_1, k_2, t) = \bar{h}_0(k_1, k_2) \exp\left\{ i\omega\sqrt{k_1^2 + k_2^2 t} \right\} + \bar{h}_0^*(-k_1, -k_2) \exp\left\{ -i\omega\sqrt{k_1^2 + k_2^2 t} \right\}. \tag{1}$$

Here, $\omega(k) = \sqrt{gk}$ is the dispersion relation for deep water.

In turn, the amplitudes of the wave field are obtained from the spectrum (1) by means of DFT:

$$h(m_1, m_2, t) = \sum_{(k_1, k_2)} \bar{h}(k_1, k_2, t) \exp\{i(k_1 m_1 + k_2 m_2)\}.$$

Obviously, during animation of the water surface by the described method, the resulting quality will depend on the implementation time of the chosen DFT algorithm. A high speed allows one to increase the size of the fragment or number of frames per second.

3 Parallel Algorithms of 2D DFT

Today, multiprocessor systems are common. Ordinary personal computers are even equipped with dual-core and quad-core processors. In addition, multiprocessor video cards with Cuda technology[1] are now available. Therefore, parallelization has become a more natural way to decrease implementation time. However, although the theory of parallel algorithms has quickly progressed, the propositions for multidimensional orthogonal transforms are limited to certain cases of data mapping and a component-wise approach (Honec et al. 2001; Noskov and Tutatchikov 2011; Takahashi 2003). 2D DFT uses a row-column algorithm with parallel implementation of one-dimensional DFTs. Here, the comparison of such an approach with the author's parallel algorithms of 2D DFT is described.

The main idea is the immersion of data in a special algebraic structure, which is four-dimensional (4D) hypercomplex algebra $\mathbf{B}_2$ or direct sum of complex algebras $\mathbf{C} \oplus \mathbf{C}$ isomorphic to $\mathbf{B}_2$. The details are described here.

[1] Cuda Tutorial http://llpanorama.wordpress.com/cuda-tutorial/

3.1 The Hypercomplex Discrete Fourier Transform

Let us consider the 2D hypercomplex discrete Fourier transform (HDFT):

$$F(m_1, m_2) = \sum_{n_1, n_2 = 0}^{N-1} f(n_1, n_2) W^{<\mathbf{m}, \mathbf{n}>}, \ W^{<\mathbf{m}, \mathbf{n}>} = \prod_{k=1}^{2} w_k^{m_k n_k}, \ w_k^N = 1 \quad (2)$$

It was introduced by Chernov (1993) as an auxiliary transform for efficient calculation of DFT for real signals. The main idea was the embedding of real signal values and complex roots in multidimensional algebra that contains a sufficient set of automorphisms for overlapping algorithms (Chernov 1993). Note that the complex roots w_k of the N-th exponent of unity are found in different subalgebras of 4D hypercomplex algebra $\mathbf{B}_2$.

Restoration of a complex spectrum from HDFT requires only two additions per pixel.

3.2 Necessary Information from Algebraic Theory

The 2^d-dimensional $\mathbf{R}$-algebra $\mathbf{B}_d$ with the basis

$$\Lambda = \left\{ \prod_{i \in I} \varepsilon_i^{\alpha_i}, \ \alpha_i \in \{0, 1\}; \ I = \{1, \ldots, d\} \right\}$$

is called commutative-associative hypercomplex algebra. Here $\varepsilon_1, \ \varepsilon_2, \ \ldots \varepsilon_d$ is the basis of d-dimensional linear space $\mathbf{V}$, $\varepsilon_i^0 = 1$, $\varepsilon_i^1 = \varepsilon_i$, and their multiplication rule is given by $\varepsilon_i \varepsilon_j = \varepsilon_j \varepsilon_i, \quad \varepsilon_i^2 = \beta_i, \quad \beta_i = \pm 1, \quad i, j \in I.$

An arbitrary element of $g \in \mathbf{B}_d$ takes the form

$$g = \xi_0 E_0 + \ldots + \xi_{2^d - 1} E_{2^d - 1} = \sum_{t \in T} \xi_t E_t \quad (3)$$

where

$$E_t = \prod_{i \in I} \varepsilon_i^{\alpha_i}, \ t = \sum_{i \in I} \alpha_i 2^{i-1} \in T = \left\{0, \ 1, \ldots, 2^d \ - \ 1\right\}. \quad (4)$$

In this hypercomplex algebra, the addition is implemented component-wise. The multiplication is defined by multiplication of basic elements:

$$E_t E_\tau = \Psi(t, \tau) E_{t \oplus \tau}, \ \forall t, \tau \in T,$$

where $\oplus$ denotes bitwise addition modulo 2,

$$\Psi(t, \tau) = \prod_{i \in I} \beta_i^{h_i(t, \tau)}, \ h_i(t, \tau) = \alpha_i \alpha_i', \ \tau = \sum_{i \in I} \alpha_i' 2^{i-1}$$

Proof of these relations can be found in Aliev (2002). Here, we will consider 2^d-dimensional hypercomplex algebra

$$\mathbf{B}_d \cong \underbrace{\mathbf{C} \oplus \mathbf{C} \oplus \ldots \oplus \mathbf{C}}_{2^{d-1}},$$

which can be represented as direct sum of complex algebras and guarantees the minimal number of real operations required for addition and multiplication of elements in $\mathbf{B}_d$ (Aliev 2002). In this case, at least one element α_i has corresponded to $\beta_i = -1$. Further, without loss of generality, we will consider $\beta_1 = -1$ and $\beta_i = 1$ for other values $i \in I$. Such a structure of the algebra allows us to develop an effective parallel algorithm for any computations in it.

The set of automorphisms in $\mathbf{B}_d$ has the form:

$$\varphi_k(g) = \sum_{t \in T} \prod_{i \in I} (-1)^{\alpha_i} \prod_{j \in I} (-1)^{\alpha_j} \xi_t E_t,\ k \in T \tag{5}$$

where $\{\alpha_i\}_{i \in I}$ and t are associated with Eq. (4), and $\{\alpha_j\}_{j \in I}$ and k are associated in a similar way.

3.3 The Basic Algorithm of 2D HDFT

The basic relationship for the radix 2 decomposition of HDFT (2) takes the form (Aliev 2002; Chicheva 2011):

$$F(m_1, m_2) = \sum_{a_1, a_2 = 0}^{1} \tilde{F}_{a_1 a_2}(m_1, m_2) w_1^{a_1 m_1} w_2^{a_2 m_2}, \tag{6}$$

where

$$\begin{aligned} \tilde{F}_{a_1 a_2}(m_1, m_2) &= \sum_{n_1, n_2 = 0}^{N/2-1} f(2n_1 + a_1, 2n_2 + a_2) \left(w_1^2\right)^{m_1 n_1} \left(w_2^2\right)^{m_2 n_2}, \\ & 0 \le m_j \le N/2 - 1,\ \ j = \overline{1, d} \end{aligned} \tag{7}$$

Values of the hypercomplex spectrum for other domains are defined without multiplications according to the known properties of DFT periodicity.

Such decomposition steps are fulfilled up to transforms of trivial lengths. One of the advantages of data representation in 4D algebra $\mathbf{B}_2$ is the possibility to use hypercomplex spectrum symmetric properties. For real input signal $f(n_1, n_2) \in \mathbf{R}$, the following symmetry relations are valid:

$$\begin{aligned} F(N - m_1, m_2) &= \varphi_1(F(m_1, m_2)), \\ F(m_1, N - m_2) &= \varphi_2(F(m_1, m_2)), \\ F(N - m_1, N - m_2) &= \varphi_3(F(m_1, m_2)), \end{aligned} \tag{8}$$

where φ_k are the automorphisms from the set (5). These symmetries allow us to fulfill computations only for the $0 \le m_j \le N/4 - 1$, $j = \overline{1, d}$ and fill in other domains based on symmetric properties.

3.4 Parallel Algorithm Based on Algebra Structure

First, let us show how parallel computations are fulfilled in the described algebra $\mathbf{B}_2$. Let an arbitrary element g of 2^d-dimensional algebra $\mathbf{B}_d$ be defined by Eq. (3). At $d = 2$, it takes a form:

$$g = \xi_0 + \xi_1\varepsilon_1 + \xi_2\varepsilon_2 + \xi_3\varepsilon_1\varepsilon_2, \tag{9}$$

where $\varepsilon_1^2 = -1$, $\varepsilon_2^2 = 1$, $\varepsilon_1\varepsilon_2 = \varepsilon_2\varepsilon_1$.

Let us divide the set $\{E_t\}_{t \in T}$ into two parts: $t \in T'$, if E_t does not contain ε_1, and $t \in T''$ otherwise. In our case, they are $\{1, \varepsilon_2\}$ and $\{\varepsilon_1, \varepsilon_1\varepsilon_2\}$. Then, we introduce the following change of variables:

$$u_0 = 1 + \varepsilon_2,\ u_1 = 1 - \varepsilon_2,\ u_2 = \varepsilon_1 + \varepsilon_1\varepsilon_2,\ u_3 = \varepsilon_1 - \varepsilon_1\varepsilon_2. \tag{10}$$

The multiplication rules for the new basis elements are given in the form

$$u_j^2 = \begin{cases} pu_j, & \text{if} \quad j < p, \\ -pu_{j-2^{d-1}} & \text{if} \quad j \ge p, \end{cases}$$

$$u_j u_k = \begin{cases} pu_k, & \text{if} \quad k = j + p, \\ 0, & \text{otherwise}, \end{cases}$$

where $p = 2^{d-1}$ (Table 1).

Note that there are many products equal to zero. This fact allows us to represent a multiplication of two arbitrary elements of algebra $\mathbf{B}_2$:

$$(xu_0 + yu_1 + zu_2 + vu_3)(\alpha u_0 + \beta u_1 + \gamma u_2 + \delta u_3)$$

in the following form of two independent products:

$$(xu_0 + zu_2)(\alpha u_0 + \gamma u_2) = 2((x\alpha - z\gamma)u_0 + (x\gamma + z\alpha)u_2),$$
$$(yu_1 + vu_3)(\beta u_1 + \delta u_3) = 2((y\beta - v\delta)u_1 + (y\delta + v\beta)u_3).$$

Table 1 Multiplication rules for the basic elements

	u_0	u_1	u_2	u_3
u_0	$2u_0$	0	$2u_2$	0
u_1	0	$2u_1$	0	$2u_3$
u_2	$2u_2$	0	$-2u_0$	0
u_3	0	$2u_3$	0	$-2u_1$

Each of them requires 3 real multiplications and 3 real additions (by analogy with multiplication of complex numbers). In such a representation, the calculation can be parallelized into two independent branches that do not require data exchange. Because the change of variables is linear, the addition saves the component-wise form.

It can be shown that, for an arbitrary $q \in \mathbf{B}_2$, the change to the new representation will require 4 real additions. However, for real and complex numbers, such a change does not require nontrivial arithmetic operations. The inverse change to the original representation also calls for 4 real additions. Thus, the presented technique allows us to parallelize any linear algorithm of multidimensional signal processing with a small number of additional operations and high efficiency.

Now, let us construct a parallel HDFT algorithm in such a technique. It is known that the previously described fast algorithm of DFT, like the multidimensional Cooley-Tukey scheme, requires only additions and multiplications. In our case, we need to operate with hypercomplex numbers. Let us use the representation (9) for computations during algorithm steps. The structure of the fast HDFT algorithm (Chicheva 2011) allows us to completely separate the calculation by the same principle used in representation (9). As a result, we obtain the following parallel algorithm of the 2D HDFT:

Step 1. Conversion from original representation (9) to representation in basis (10) (*trivial* because input data are real and roots w_k are complex)
Step 2. Distribution of data between two processors
Step 3. Calculation of transform (6) on every processor using an algorithm of Cooley-Tukey type (see Sect. 3.2)
Step 4. Returning results on one processor
Step 5. Reconstruction of the hypercomplex spectrum

This algorithm preserves an important feature of a sequential algorithm—using the symmetric properties of real signal hypercomplex spectrum (8). However, to use the symmetry, additional data exchange is required, thus resulting in a slight decrease of the general efficiency of parallelization.

3.5 Parallel Algorithm Based on Inner Parallelism of a Decomposition Scheme

The key operation of the sequential HDFT algorithm is the reconstruction (7) of the complete spectrum $F(m_1, m_2)$ from the known (derived) values of partial spectra $\tilde{F}_{a_1 a_2}(m_1, m_2)$. Suppose that every partial spectrum has been calculated on a separate processor. Note that the processing time is approximately the same for all the processors because the sizes and the calculation algorithms are the same. Next, every processor performs multiplications of the partial spectrum elements and the power of the roots w_1, w_2. Then, the values derived are transferred to one of the processors, where the hypercomplex spectrum is finally formed.

In this way, the process can be parallelized between any number of processors divisible by 2^d by using several decomposition steps of type (7). The expected time of the hypercomplex spectrum computation is inversely proportional to the number of processors because the main computational effort is attributed to computation of HDFTs of smaller sizes.

The advantage of this approach is twofold: reduction of transmitted data volume due to symmetry of the hypercomplex spectrum of real signals.

4 Experimental Research

In this section, the results of the application of parallel DFT algorithms to the problem of 3D modeling are given. Testing was performed on a computer with the following characteristics:

- processor AMD Phenom II X4 965 @ 3.40 Ghz;
- core memory 4 Gb;
- video card NVidia GeForce 450 GTS 1 Gb.

The computer used the following software:

- Windows;
- NVIDIA DevDriver 301.32;
- CudaToolkit 4.1.28;
- NVIDIA GPU Computing SDK 4.1;
- JDK 1.7.0;
- JCuda 0.4.1.

For comparison, three parallel algorithms were chosen:

- row-column algorithm with parallel calculation one-dimensional DFT along rows or columns of input data matrix (RC);
- algorithm based on decomposition scheme (DS); see Sect. 3.5;
- algorithm based on algebra structure (AS); see Sect. 3.4.

Figure 1 shows the implementation time of the algorithms using two processors. The results on four processors are given in Fig. 2. You can see the most effective approach is parallelization in algebra $\mathbf{C} \oplus \mathbf{C}$. Application of these algorithms for water surface modeling can be estimated by the number of frames per second, which can be obtained for every case using four processors. It is shown in Fig. 3, where the bold dashed line denotes the lower bound of comfortable vision for the human eye, which is equal to 24 frames per second. All algorithms are guaranteed to be acceptable quality for a synthesized fragment up to 256×256 pixels in size.

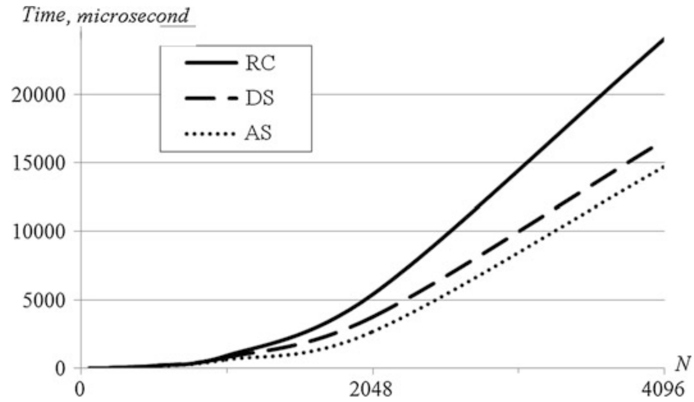


Fig. 1 Implementation time for two processors. RC, row-column algorithm; DS, decomposition scheme algorithm; AS, algebra structure algorithm

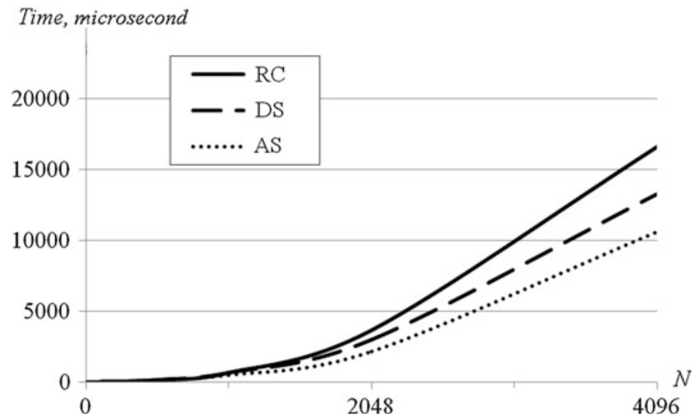


Fig. 2 Implementation time for four processors. RC, row-column algorithm; DS, decomposition scheme algorithm; AS, algebra structure algorithm

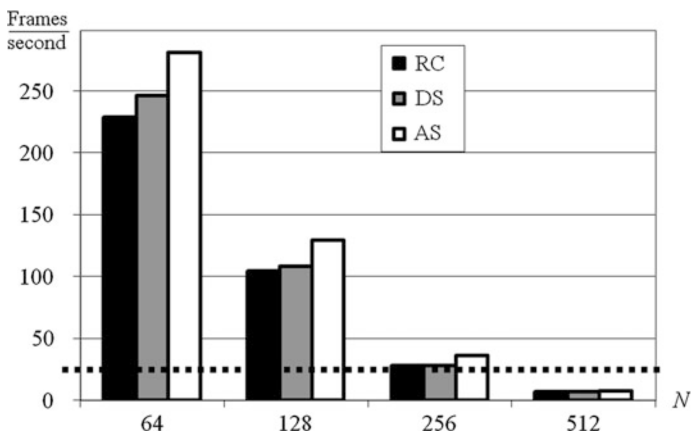


Fig. 3 Number of frames per second during implementation on four processors. RC, row-column algorithm; DS, decomposition scheme algorithm; AS, algebra structure algorithm

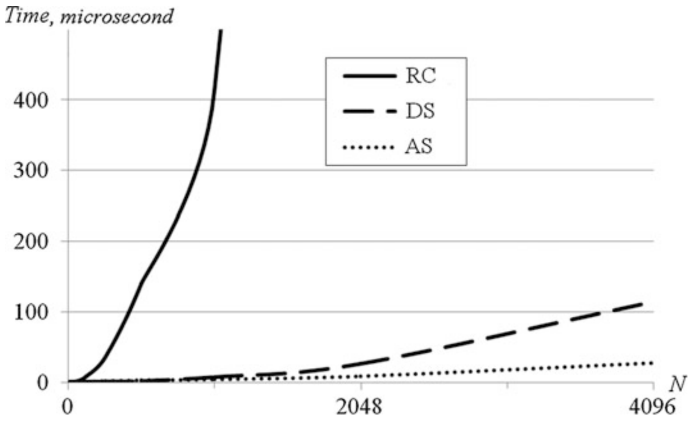


Fig. 4 Implementation time for graphics processor. RC, row-column algorithm; DS, decomposition scheme algorithm; AS, algebra structure algorithm

For large fragment modeling, let us consider algorithm implementation with a graphics processor. Implementation time is given in Fig. 4. The row–column algorithm is the slowest in this case. Further research (see Fig. 5) shows that this algorithm is not appropriate for large fragment modeling. At the same time, the algorithms for data immersion in special algebraic structures guarantee acceptable quality.

The examples of water surface fragments obtained during research are presented in Figs. 6 and 7 for small and large sizes.

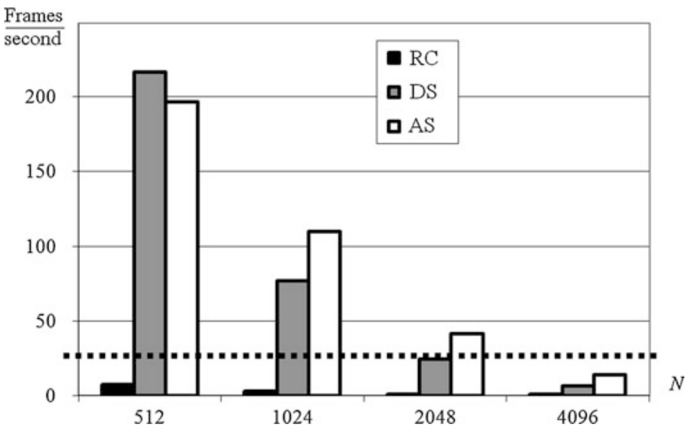


Fig. 5 Number of frames per second during implementation on graphics processor. RC, row-column algorithm; DS, decomposition scheme algorithm; AS, algebra structure algorithm

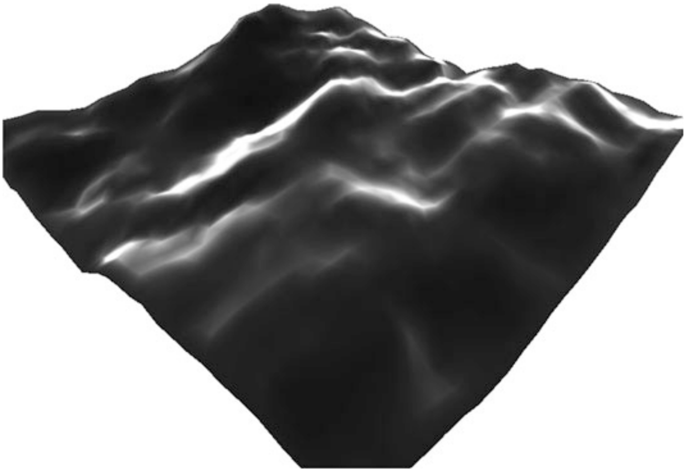


Fig. 6 Obtained water surface fragment of 32 × 32 pixels

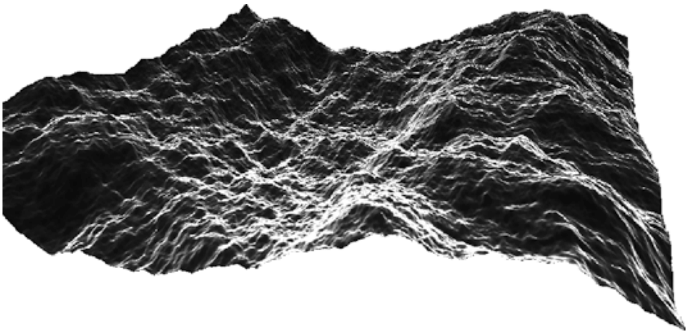


Fig. 7 Obtained water surface fragment of 512 × 512 pixels

5 Conclusion

In this chapter, the research on DFT parallel algorithm applications for the 3D modeling of the earth's surface has been described. Some approaches have been considered. For the example of the ocean surface, it was been shown that the traditional row–column method gives poor results, even when a graphics processor was used. In turn, the application of 2D DS in combination with data immersion into special algebraic structures allows one to construct highly effective parallel algorithms that improve the quality of 3D models.

Acknowledgments This research was financially supported by the Russian Foundation for Basic Research, project No. 12-07-00751, 12-01-00822, 11-07-12060, 11-07-12059.

References

Aliev MV (2002) Fast algorithms of d-dimensional DFT of real signal in commutative-associative algebras of 2d dimensionality over the real number field (in Russian). Comput Opt 24:130–136

Chernov VM (1993) Fast algorithms of discrete orthogonal transforms for data represented in cyclotomic fields. Pattern Recogn Image Anal 3(4):455–458

Chicheva MA (2011) Parallel computation of multidimensional discrete orthogonal transforms reducible to a discrete fourier transform. Pattern Recogn Image Anal 21(3):381–383

Finch M (2004) Effective water simulation from physical models. In: Fernando R (ed) GPU gems: programming techniques, tips and tricks for real-time graphics. Addison-Wesley, Reading, pp 5–29

Floriania LD, Montanib C, Scopignoc R (1994) Parallelizing visibility computations on triangulated terrains. Int J Geogr Inf Syst 8(6):515–531

Frejlichowski D (2011) A three-dimensional shape description algorithm based on polar-fourier transform for 3D model retrieval. In: SCIA'11 proceedings of the 17th scandinavian conference on image analysis. Springer, Berlin, pp 457–466

Guan Q, Kyriakidis PC, Goodchild M (2011) A parallel computing approach to fast geostatistical areal interpolation. Int J Geogr Inf Sci 25(8):1241–1267 Special Issue on Data-Intensive Geospatial Computing

Henry D (2008) On gerstner's water wave. J Nonlinear Math Phys 15(2):87–95

Honec J, Honec P, Petyovsky P, Valach S, Brambor J (2001) Parallel 2-D FFT implementation with DSPs. In: 13th international conference on process control, Strbske Pleso, Slovakia

Isidoro J, Vlachos A, Brennan C (2002) Effective water simulation from physical models. Rendering ocean water. Wordware Publishing, Inc., Plano, pp 347–356

Noskov MV, Tutatchikov VS (2011) About calculation of two-dimensional fast fourier transform. In: 8th open German-Russian workshop "pattern recognition and image understanding" (OGRW-8-2011), Nizhny Novgorod, pp 219–221

Takahashi D (2003) E–cient implementation of parallel three-dimensional FFT on clusters of PCs. Comput Phys Commun 152:144–150

Tessendorf J (2009) Simulating ocean water. In: ACM SIGGRAPH course notes #9 (simulating nature: realistic and interactive techniques). ACM Press, New York

Widenera MJ, Cragob NC, Aldstadta O (2012) Developing a parallel computational implementation of AMOEBA. Int J Geogr Inf Sci 26(9):1707–1723

Yinab L, Shawac S-L, Wangd D, Carre EA, Berryf MW, Grossg LJ, Comiskeye EJ (2012) A framework of integrating GIS and parallel computing for spatial control problems—a case study of wildfire control. Int J Geogr Inf Sci 26(4):621–641

Adaptive Multidimensional Measurement Processing Using Intelligent GIS Technologies

A. Pankin, A. Vitol, and N. Zhukova

Abstract Intelligent geographic information systems (IGIS) are currently very popular with end-users who need to obtain complex information about spatial technical or natural objects. The number of data sources and volume of data are constantly increasing necessitating the use of new technologies for data processing in IGIS. A significant amount of processed data are time series with measurements of various object parameters for both technical and environmental objects. Measurements of time series are usually complicated and not stationary; they contain noise, outliers, and gaps, so a wide range of methods are used for their processing. We propose a method for building adaptive processes for dynamic measurement analysis based on preliminary estimated data using a set of exploration analysis algorithms. The method aims to provide automated operative processing and analyses of heterogeneous data acquired from different sources.

Keywords Adaptive processing · Multidimensional measurements · Intelligent geographic information systems

A. Pankin · N. Zhukova (✉)
St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS), 39, 14 Linia VO, St. Petersburg, Russia199178,
e-mail: nazhukova@mail.ru

A. Pankin
e-mail: gna@oogis.ru

A. Vitol
St. Petersburg Electrotechnical University "LETI", 5, Prof. Popova street, St. Petersburg, Russia197376,

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_12,

1 Introduction

Modern users of information systems, including data processing and analysis systems, want tools that provide a comfortable work environment, access to necessary data for solving tasks in real time or with minimal delay for representation of initial data, and results provided in forms that can be easily interpreted. Information systems have to propose appropriate variants of solutions for user tasks and instruments for their comprehensive analysis. Because systems have to deal with heterogeneous data, it is important to support the processing and analysis of data that is quantitative and qualitative, structured and unstructured, certain and uncertain. Currently developed platforms for data processing and analysis (e.g., Hadoop,[1] Greenplum[2]) are applicable only for research purposes. To solve practical tasks, a set of business processes in the form of processing scenarios are usually developed. Efficient scenarios that handle Internet data, financial data, trade data, and medical data already exist. When new information systems are developed, these existing scenarios are adapted for the composition of practical tasks.

There are four levels of maturity of data processing systems: 1) systems that provide data aggregation and formation of statistical reports on users' queries; 2) systems that provide tools to process and analyze data; 3) systems that monitor environmental and technical objects; and 4) systems that predict an object's state based on specific models and/or statistical data. Currently, for solving tasks of data processing and analysis, end-users use tightly specialized systems, the maturity level of which varies between the second and third levels. This is due to the high cost of system development, which has almost exponential dependence on the system's maturity level. Moreover, recently research (e.g., that by Oracle Corporation) concluded that the existing software, despite resources spent for its updating, does not satisfy the requirements of end-users.

An important aspect of data processing and analysis systems development is the handling/processing of multidimensional measurements of technical and environmental object parameters. Active development of this sphere is associated with quick growth of the number of measuring instruments. Systems for measurement processing and analyses are usually based on geographic information system (GIS) technologies, which provide the most convenient environment for working with geospatial object parameters. These systems' are of the second maturity level, because the cost of the development of the measurement processing system is much higher. This is due primarily to the fact that very few scenarios for measurement processing have been developed so far. Scenarios developed for other spheres of information system applications cannot be used for measurement processing. The main reason for the complexity of the measurement analyses processes is that the processes have not been determined. They significantly depend

[1] http://hadoop.apache.org

[2] http://www.greenplum.com

on structure and quality of data being processed, accessible methods and algorithms, and end-user requirements for the results of processing. This implies that only top-level (general) processes can be described and formalized. Lower-level processes must be built dynamically. For building dynamic processes, a variety of artificial intelligence tools that are included in intelligent GIS (IGIS) (Popovich et al. 2009) can be efficiently used, such as an inference engine, knowledge base, or domain ontology. These measurements have several distinguishing features, including large volumes of data, time series that are complex and not stationary, and a lack of coordination in space and time. Therefore, measurements can be successfully processed (at the desired time and with the desired precision) only if an automated adaptive data processing approach is used.

To build adaptive processes based on IGIS, two tasks need to be completed. The first task is to develop a general method for building measurement analysis processes using artificial intelligence provided by IGIS in combination with methods of explorationary data analyses and data mining methods, which provide additional information about the analyzed data. The second task is to integrate the components for measurement adaptive processing and analyses into IGIS systems. The joint use of two technologies—adaptive data processing technology and intelligent GIS technology—may create an efficient instrument for processing and analysis of multi dimensional measurements according to end-user requirements.

2 Models for Multidimensional Measurement Processing

When completing a task in measurement processing and analyses, it is necessary to define the business process, form its detailed description, define the parameters of applied algorithms, and finally execute the process. Enumerated tasks correspond to three levels of process descriptions. The first one corresponds to the conceptual description, in which the main goals of processing are defined. The second level is used to describe the processes in detail, taking into account various internal and external factors that influence the obtained results. The third level is focused on the description of processes in the form demanded for their execution. The listed tasks are very similar to those used in modeling information system architecture. For example, consider two different frameworks—model-driven architecture (MDA; Kleppe et al. 2003) and the Zachman framework (ZF) (O'Rourke et al. 2003), which are included in architecture modeling techniques.[3] MDA is a design approach for the development of software systems; it provides a set of guidelines for the structuring of specifications, (which are expressed as models) and supports model-driven engineering of software systems. ZF was proposed by John A. Zachman as a way of conceptualizing what is involved in information systems architecture. When developing an information system using one of these

[3] http://www.agiledata.org/essays/enterpriseArchitectureTechniques.html

approaches, the system architecture is designed according to the business processes that must be supported by the system. If new business processes are added to the system, the system usually has to be restructured. When considering information systems for measurement processing, we have to solve the same task each time measurements from new data sources are acquired, which requires modification of the existing analysis processes. The basic principles that are proposed by the MDA approach can be efficiently used for building the dynamic processes of measurement analyses.

A main problem of MDA automating the mapping of different models. The main difficulties are related with mapping the computational independent business model with the platform independent component model because it cannot be considered as a technological translation from one model to another. Several solutions have been proposed, (e.g., Hammoudi et al. 2010), but they are all rather specialized. In measurement processing systems, problems of transformation can be solved using a composition of business rules and supplementary algorithms that provide additional information about data.

For describing the adaptive processes of measurement analyses, three modeling levels of processes specifications are proposed: the conceptual independent model, logical independent model, and physical specific model for process descriptions. These three levels of descriptions correspond to three levels of abstraction: the usage, meaning, and structures of processes for data analyses, respectively. In Fig. 1, the compliance of ZF components, models of the MDA approach, and models of the proposed model-driven architecture for processes (MDAP) approach for the description abstraction level are given.

Models of MDA. The computational independent model (CIM) specifies activities in the form of business processes that are being processed in the information system without displaying details. CIM has a "vocabulary" of domain models and represents the logic of subject domain functioning. It does not contain any information about models or artifacts that will be used for implementation of the information system. Its aim is to form the main requirements of the information

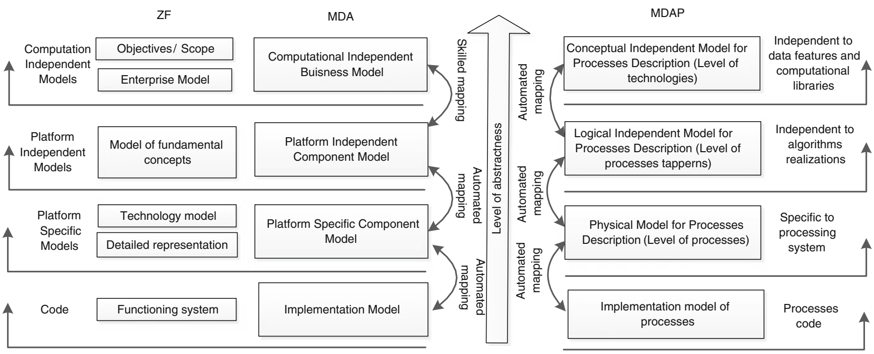


Fig. 1 Zachman framework components, models of the MDA approach, and models of the proposed MDAP

system and to describe the environment in which it operates. CIM is created by business analysts, domain experts, or domain users of the system.

The platform independent model (PIM) is a set of models (mostly unified modelling language models) that are suitable for use in various platforms. PIM describes the information system in detail but independently of the platform. In addition, PIM creates specifications for the required services of the information system. The main actors in this MDA level are software analysts.

The platform-specific model (PSM) describes the information system so that it can be realized within a certain platform. PSM extends the specification of PIM, with details that depend on the type of platform that will be used for the information system's implementation. This level is created by software architect.

The Implementation model (IM) defines all necessary information for system implementation. The Implementation model is based on the PSM and is represented in the form of program code.

Models of MDAP. The concept independent model for process description (CIMP) is presented in two levels—pragmatic and semantic. The pragmatic level is the level where problem-oriented analysis is done and general processes of measurement analyses are defined. General processes are defined by experts of the subject domain and can be presented as sequence of steps. For every process, a clear motivation or goal that can be analyzed together with a final process state must be defined. Hierarchies of goals and opportunities for their achievement can also be specified. It is important to structure general processes at the CIMP level because such knowledge provides motivation for various configurations of general process implementation at the semantic level. The semantic level corresponds to the description of general processes of measurement analyses at the technology level. Technology is a description of a general process or its separate steps for implementation. Alternative technologies or sequences of technologies for implementation of a general process step depend on the types of data sources and various external factors that can influence the process of data analyses. Each technology contains information about computational methods that are to be used for solving processing tasks.

The main entities at the CIMP level are the goals of data processing and analyses and requirements for results. The main entities at the pragmatic level can be linked by relations and are defined by object domain. As goals specialized computational tasks or tasks for building different formalized measurement representations according to Pankin et al. (2013) that can be used for solving tasks of objects and situations assessment are considered. A wide range of end-users can be consumers of measurement processing results. The requirements for the results of data processing are formed separately for each group of users.

The logical independent model of process description (LIMP) corresponds to the description of general measurement analysis processes at the level of process patterns. Patterns are variants for technology implementation. The goal of this level is to represent each element of technology in a set of process patterns. At this level, the hierarchy of process patterns is defined. Each process pattern used to implement technology or one of its elements is considered to be a high-level pattern, which can consist of lower-level process patterns that give more detailed descriptions. Process patterns are detailed until they are presented as a sequence of algorithms or groups of algorithms. The main entities at this level are input data-specific features. Most attention is paid to the features that influence (either directly or indirectly on) the applied data processing algorithms. The features of input data are retrieved using statistical methods and methods of exploration analyses. The list of input data features can be extended using formalized expert knowledge about the subject domain.

The physical model for processes (PMP) is used to describe implementation aspects of the process patterns. PMP allows process patterns to be represented in the form of a sequence of measurement processing and analyses groups of algorithms or algorithms. Each process that describes a process pattern can be represented as a sequence of lower-level processes that specify groups of algorithms that are used. Representation of each process of the lowest level as a sequence of algorithms is obligatory. The main entity at this level is data estimation. The data estimation element provides estimation of input data and acquisition of essential information for selecting algorithms. Sets of data estimations are provided by subject domain experts.

The implementation model of processes (IMP) provides a description of the general processes in measurement analyses in the form of executable processes. Executable processes are described in a special language that can be executed using standard engines. The goal of this model is to describe processes as a set of executable algorithms with a defined set of parameters and sequence of algorithm execution. The key element of this level is the processing result estimation. Processing result estimations, along with a priori knowledge, are used for defining algorithm parameters. Estimations are obtained after algorithms with defined sets of parameters are executed and appropriate estimation criteria are calculated.

Class diagrams for models of MDAP are given in Fig. 2.

3 Method for Building Adaptive Processes for Dynamic Measurement Analyses

The method for building adaptive processes aims to build executable processes from general processes of measurement analyses in the automation mode using MDAP. The method is based on using a set of developed technologies and process patterns for measurement processing that are defined a priori and are similar for various subject domains. Each technology or pattern is aimed to solve one task or a group of tasks with specified input data (the structure and quality of data are defined) and requirements for the obtained solutions. When building processes, the choice of process elements is based on previously accumulated knowledge and the results of exploratory data analysis, which considers the application of statistical methods and data mining methods.

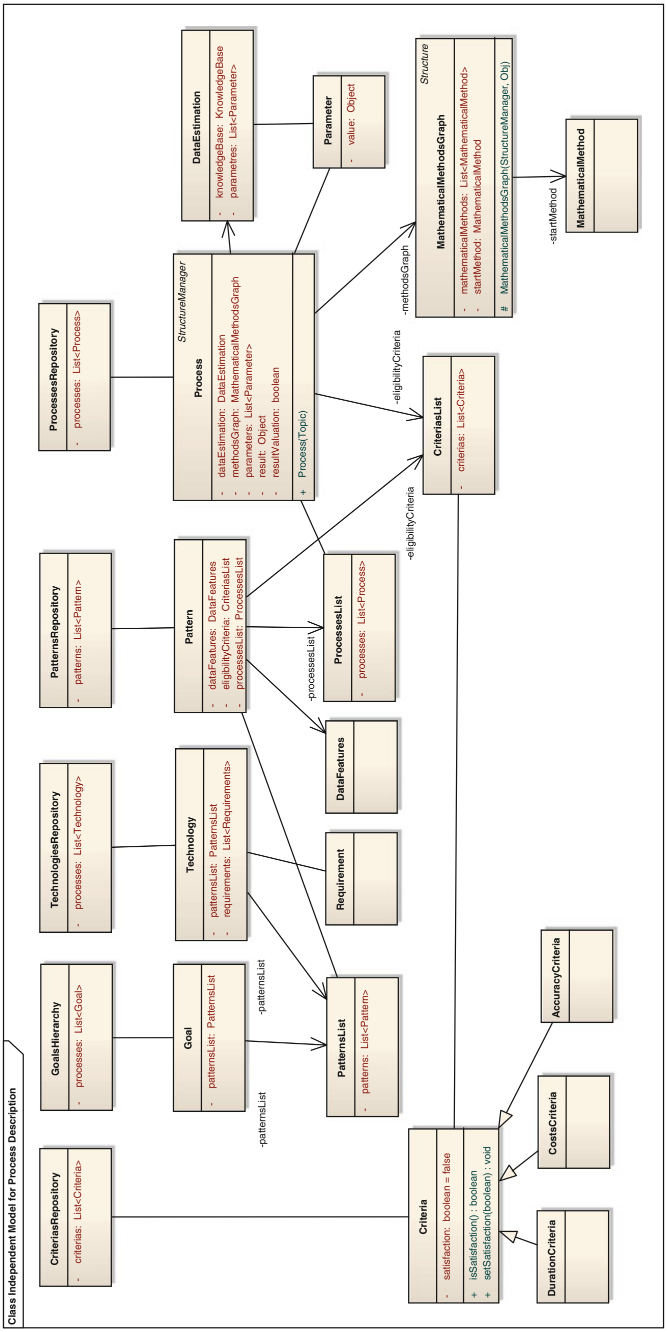


Fig. 2 Class diagram for models of MDAP

Methods for building adaptive processes suppose that the following stages are executed when a user task is defined (Fig. 3). First, a technology (or several technologies) from a set of technologies supported by the system is selected. Second, the selected technology is detailed by a set of less general technologies or by a set of process patterns. Process patterns include a description of groups of algorithms or separate algorithms and description of a sequence of algorithm applications for solving each element specified in technology. For each technology, one or more processing patterns are built. Each of the elements of the process pattern can be specified by a set of lower-level process patterns. Then, each process pattern is represented as a sequence of algorithms that are elements of a process. Third, for each algorithm, input parameters must be specified before the process can be executed. The executable process is represented in a sequence of actions that can be executed consequently or concurrently.

The choice of technologies, process patterns, and processes is based on the application of decision rules. Rules for choosing technologies, process patterns, and processes have different natures and are of different types. However, all rules have one aim—to solve end-user tasks and to obtain solutions of the required quality. The quality of a solution is estimated using the parameters of solution accuracy and processing time.

Three approaches for the method implementation can be considered. The first approach, which is currently widely used, determines technologies, process patterns, and processes in advance. The disadvantage of this approach is that when the structure or quality of data is changed, new methods and algorithms are developed; thus, it is necessary to make changes in the system at the level of program code. In addition, until the end of the process execution, the efficiency of processing cannot be estimated.

The second approach is based on the application of exploration analysis algorithms and overcomes the disadvantages of the first approach. This is achieved by selecting suitable process patterns and processes while problem solving, and if necessary, adjusting processes using the intermediate results of processing.

The third approach to build processes allows technologies to be chosen dynamically. When selecting technologies, the intelligent exploratory analysis of data is used, which is based on data mining methods. Such systems use domain ontology, so the processes of measurement analyses are built and executed with regard to objects of the subject domain. Formalized descriptions of the key elements that are used in the method of adaptive processing are listed in Table 1. A description of the method is given in Fig. 4.

The set of elements used for building adaptive processes can be described using a graph structure. Nodes of the graph are conditional transitions that are described as business rules or elements of adaptive processes. The space of all elements can be decomposed into a number of subspaces, each of which is associated with one or several end-user tasks.

The spaces of elements that contain technologies, process patterns, and processes are sets of multi level hierarchical structures. Nodes of structures contain conditional transitions called edges, which are elements of the corresponding

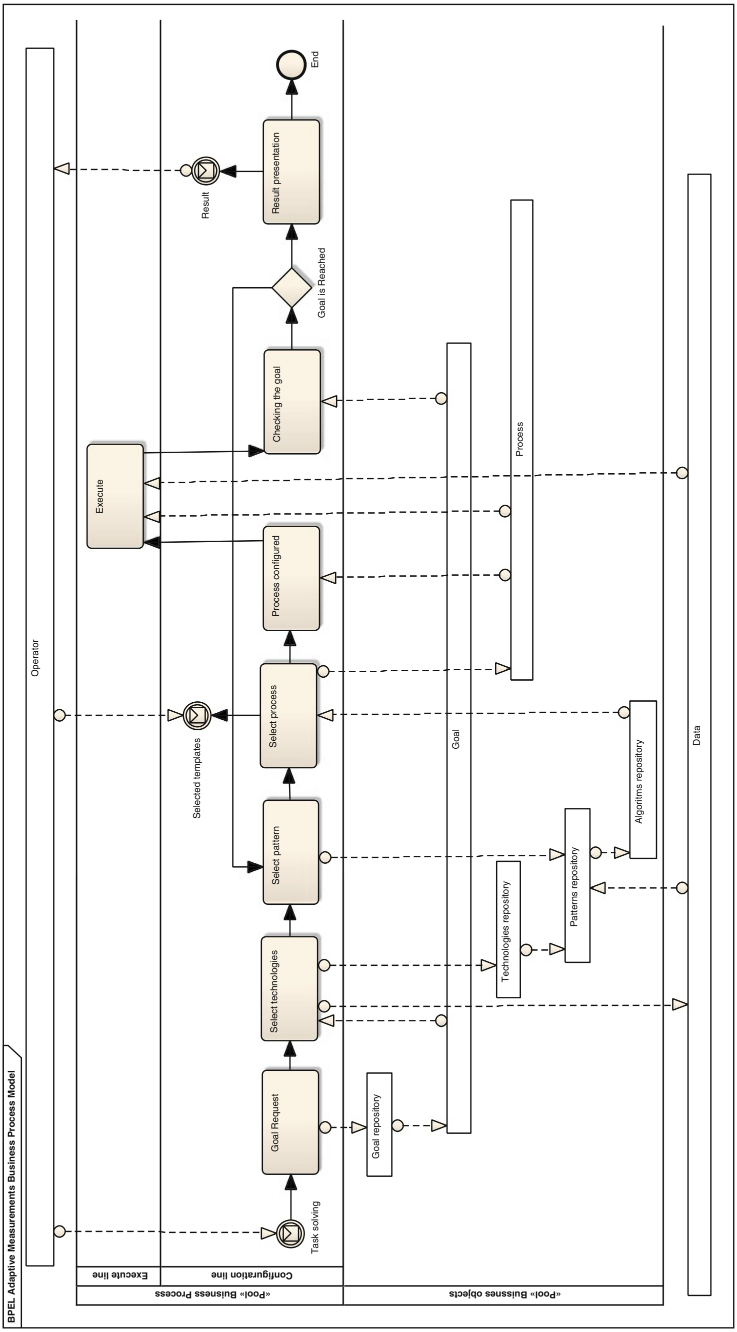


Fig. 3 General structure of the method for building adaptive processes

Table 1 Description of process elements

Element	Description	Parameters
Technologies (W)	$W = <I,\ C,\ W^R,\ z_w>$, $C = \{J\}_{i=1}^{N_C}$, $J = <d_1, \ldots, d_k, \ldots, d_{N_J}>$, $z_w : <C, W^R> \to J$	I—step of general measurements analysis process, C—set of technologies, J—a technology from set C, d_k—an element of technology, W^R—set of rules for selecting technologies, N_C—total amount of technologies, N_J—total amount of processing elements in J, z_w—a function that provides selection of a technology J from C using rules W^R
Patterns (G)	$G = <J,\ G_V, G_E, z_{G_E}, z_{G_V}> G_V = <b, r>$, $b \in G_V^A$, $G_E \subseteq G_V \times G_V$, $z_{G_V} : G_V \to G_V^A$, $z_{G_A} : G_E \to G_E^A$, $G_V^A = \{b_i\}_{i=0}^{1}$, $b_i = (\sigma, \varphi)$, $G_E^A = \{g_1^R, \ldots,\ g_{N_{G^R}}^R\}$	J—technology for which a process pattern is built, G_V—set of graph vertexes, G_E—set of graph edges, b—vertex type, r—pre and post conditions for processing elements described in a vertex, b_0—a vertex type that contains a named processing algorithm group $A = \{a_1,\ \ldots,\ a_{N_A}\}$, a_i—a processing algorithm, N_A—total number of processing algorithms, b_1—a vertex type that contains a named process pattern, G_V^A, G_E^A—attributes of vertexes and edges, respectively, σ—attribute type, $\varphi = (\varphi_1,\ \ldots,\ \varphi_{N_\sigma})$—vector of characteristics, associated with σ, N_σ.—number of elements in the vector of characteristics, z_{G_V}—function that assigns attributes to the vertices, z_{G_A}—function that assigns attributes to the edges, g_i^R—a rule defining conditions of moving from the current vertex towards the next, N_{G^R}—total number of rules
Processes (K)	$K = <G,\ A^d,\ O,\ z_d,\ z_a, z_c>$, $O = <a_1^d, \ldots,\ a_k^d>$, $a_i^d = \{a_i, r_i, p_i\}$, $i = 1, ..N_A$, $O \subseteq A^d \times A^d$, $z_a : A^d \to \{x_p, x_u\}$	O—sequence of algorithms, a_i^d—description of an algorithm, a_i—processing algorithm, p_i—preconditions for algorithm a_i, s_i—post conditions for algorithm a_i, z_d—description of algorithm execution order, z_a—type of algorithm execution, x_p—consequence execution of algorithms, x_u—parallel execution of algorithms, z_c—definition of conditions under which algorithms are executed consistently or in parallel
Executable processes (K^E)	$K^E = <K,\ C,\ z_o>$, $E = \{e_i\}_{i=1}^{N_E}$, $e_i = \{a_i,\ d_i\}$, $z_o : K \to K^E$	e_i—an element of executable process, a_i—a processing algorithm, d_i—a set of algorithm a_i parameters, N_E—total number of executable process elements, z_o—function that defines mapping of each element of the process K to element of an executable process K^E

Step 1 Initialization step	
1.1. **define** measurement processing and analysis steps $I' = f_1(I, S)$ according to the description of general processes for subject domain	
1.2. **define** technologies $W' = f_2(I', W, S, R, U)$ according to the defined set of steps from possible technologies W	
1.3. **define** process patterns $G' = f_3(W', G, R, h_1(U))$ according to the defined set of technologies from possible process patterns G, taking into account features of input data $h_1(U)$	
1.4. **define** processes $K' = f_4(G', K, h_2(U))$ according to the defined set of process patterns from possible processes G, taking into account processing result estimations $h_2(U)$	
Step 2 Definition of restrictions	
2.1. **calculate** the characteristic of measurement completeness $D_f = g_1(U)$ that defines a range of tasks that can be solved	
2.2 **calculate** characteristic of measurement reliability $D_r = g_2(U)$ that defines a range of requirements that can be fulfilled using analyzed data	
2.3. **calculate** characteristic of measurement quality $D_q = g_3(U)$ that defines a range of algorithms that can deal with analyzed data	
Step 3 Application of restrictions	
calculate according to restrictions	
	a reduced set of technologies $W'' = v_1(I'', W', D_f, D_r, D_q)$
	a reduced set of process patterns $G'' = v_2(W'', G', D_f, D_r, D_q)$
	a reduced set of processes $K'' = v_3(G'', K', D_f, D_r, D_q)$
Step 4 Building alternative processes	
4.1. **define** sets of knowledge-based business rules using results of exploration data analyses	
	for technologies selection $R_W = p_1(U)$ with application of data mining algorithms
	for process patterns selection $R_G = p_2(U)$ with application of statistical algorithms
	for process selection $R_K = p_3(U)$ with application of statistical and data mining algorithms

Fig. 4 Method for building adaptive processes

space. Each space can be divided into multiple crossed subspaces; for each step of a general process, a subspace of technologies is formed; for each technology, a subspace of process patterns is formed; and for each process pattern, a subspace of processes is formed. For each process, a structured set of algorithms can be defined.

3.1 Algorithm Description

In an algorithm, the input can be defined as follows: (1) I—set of steps of measurement processing and analysis; (2) R—set of requirements for the results of measurement processing; (3) S—goal of measurement processing and analysis; (4) U—input data description.

4.2. **define** functions			
	C_1 for comparing elements from a set of technologies W : $C_1 : w_i C_1 w_j \Rightarrow w_i \succ w_j$, $w_i, w_j \in W'''$		
	C_2 for comparing elements from a set of process patterns G : $C_2 : g_i C_2 g_j \Rightarrow g_i \succ g_j$, $g_i, g_j \in G'''$		
	C_3 for comparing elements from a set of processes K : $C_3 : k_i C_3 k_j \Rightarrow k_i \succ k_j$, $k_i, k_j \in K'''$		
4.3 **calculate** alternative			
	technologies $W''' = u_1(W'', R_W)$ using defined business rules R_W		
	process patterns $G''' = u_2(G'', W''', R_G)$ using defined business rules R_G		
	processes $K''' = u_3(P'', G''', R_K)$ using defined business rules R_K		
`Step 5 Selection of measurement analysis processes`			
5.1. **if** $K''' = \{\}$ **then** *Step 4*			
5. 2. **for each** $k \in K'''$			
	calculate $E_k = z_e(A, z_d)$, z_e - function of algorithms results for a priori estimation, A - set of applied algorithms, z_d - definition of algorithm application order.		
5.3. **find** $k : C_{\max} = \max\limits_{0 \le k \le N_K} \{E_k\}$, N_k - total number of analyzed processes			
5.4. **define** final process as $K_R = K_k'''$			
`Step 6. Building executable processes`			
6.1. **for each** processing element $k \in K_R$			
	define set of vectors of possible algorithm parameters T		
	for each $t \in T$		
		calculate set of output parameters $O = k\,(T)$	
		calculate estimation of results E	
		if (E satisfies R) then	
			define t as parameters of k
6.2 **if** ($\exists k$: t is not defined) **then** *step 4*			
	else *success*		

Fig. 4 continued

The output is the executable process K^E that solves the defined task according to the requirements.

Functions that are used for selecting processing elements at different steps of the described method are based on (1) a semantic description of objects of the subject domain, their properties, and relations; (2) a priori knowledge about objects of the subject domain; (3) set of a priori defined business rules; (4) set of tests, algorithms, and procedures for input data estimation; and (5) set of criteria for data processing results estimation.

4 Principles of Element Selection in the Adaptive Processing Method

For organization of the selection of process elements, three key components are needed: a model of subject domain objects, which is usually represented as an ontology; a set of knowledge-based business rules; and a library of algorithms for exploratory analyses, data classification, and estimation algorithms. These components allow processes for measurement analysis to be built in automation mode.

Model of subject domain objects. The model of subject domain contains the description of base concepts, description of time series, and description of processes elements. Base concepts of the subject domain model are presented in Table 2.

For processing a time series in automation mode, the initial time series, results of their processing, and methods for their processing have to be formally described and classified. For this, a list of classifiers (Table 3) has been determined.

Table 2 Base concepts of subject domain model

Concept	Description
Objects of subject domain	Both technical and environmental objects that are related to solve problems are considered. For each object, a set of properties, their types, and ranges of values are defined. Also, dependencies between objects are formalized.
Goals	Goals that are to be archived in the sphere of measurement processing are defined. Goals are described formally. The description of each goal contains information about the main actors that are involved, variants of goal decomposition into a sequence of lower-level goals or tasks, a sequence of processing steps that are to be executed to achieve the goal, and a list of situations when the goal can be set.
Users and their requirements	Categories of users are described. For each category, it is necessary to define a set of requirements for obtained solutions.
Data sources, data measurement instruments, and acquired data	Data sources are described using a list of measured parameters and their characteristics. Data measurement instruments are described using a set of technical characteristics defined for the instrument, including systematic errors. For acquired data, the source, measurement instrument, stage of processing, and results of processing are defined
Results of different steps of data processing	Results of different steps of data processing (represented in a form presented in Pankin et al. 2013) are given.
External factors	Description of external factors assumes enumeration of factors that can influence results of processing, estimation of expected degree of influence, and conditions when factors are to be taken into account.

Table 3 List of classifiers for processing time series

Classifier	Description
Classifier of time series	Time series are classified according to their behavior. For classification, various characteristics are used. The main goal of time series classification is to define the types of time series that can be efficiently processed using same groups of algorithms.
Classifier of time series components	Each time series can be decomposed into a set of components that can be separately described using simple statistical characteristics.
Classifier of methods, algorithms, and procedures of data processing	Methods, algorithms, and procedures of data processing are classified according to the tasks that can be solved using them. The main goal of classification is to provide the possibility to select algorithms according to a defined task.
Classifier of data type characteristics	Characteristics that can be used for description of data types are classified according to data features. This classifier is used to figure out if data possesses (or not) a defined set of features.
Classifier of various types of data descriptions	Classifier of descriptions of various types of data is used for defining feature space that fit most for representation of data of known type and for systematization of representation of different types of data.
Classifier of data estimation criteria	Classifier of data estimation criteria is organized according to data types. For each data type, a set of essential estimations that are to be calculated are presented.
Classifier of results of time series processing and analyses representation	Classifiers of results representation are based on data types and steps of data processing.

Process elements are used for building measurement analysis processes at different levels of abstraction. For process elements, the corresponding technologies, process patterns, and processes are defined in Table 1.

Business rules. A set of business rules is defined for objects of a subject domain. Business rules are an essential part of a knowledge base. They can be considered as a component that supports the adaptive behavior of processes. They are represented in the form of production rules: `if` (*head*) `then` (*body*), where *head* is a set of conditions and *body* is a set of consequences. The main feature that differentiates business rules in measurement processing systems from other information systems is that heads of rules are almost always not completely defined. Heads of rules usually contain conditions on quantity and quality characteristics of various object parameters. The values of characteristics can significantly depend on different external factors. As an example, tests for the operational processing of oceanographic data can be considered. Depending on the period and region where oceanographic data was acquired, border values of tests can differ. Therefore, values used in business rules should be defined by taking into account input data and conditions of solving measurement processing tasks. For defining values for business rule conditions, an approach that is based on historical data analyses is used (Vasiliev et al. 2007). In this approach, historical data is processed and, as the result of its processing, a description of the typical behavior of measured parameters and deviations of their behavior are obtained. These characteristics are used for business rule specification.

Business rules are used for building CIMP and LIMP and for mapping CIMP to LIMP and LIMP to SMP. When building CIMP, business rules are used for two main purposes:

- to select technologies for realization of general processes steps. This set of business rules is defined a priori (fixed business rules);
- to estimate a list of mathematical procedures and algorithms that are necessary for technology implementation. When estimating this list of mathematical methods, fixed business rules are also applied.

The LIMP model provides a description of technology implementation in the form of a graph that contains process patterns and groups of algorithms. For building this graph, business rules are applied. At this step, fixed business rules cannot be defined because input data features are to be considered when building rules. Due to the variety of data sources, the complexity of data, and its unpredictable behavior, it is impossible to specify business rules a priori. Business rules can be only defined in an undetermined form. For example, when estimating the quality of oceanographic parameters, business rules can be written as: `if` (*temperature_values* close_to *border_values*) `then` (estimate *salinity_values*). When information about a region where measurements were executed is acquired, border values for temperature parameters can be obtained using knowledge about the region or results of historical data processing. Operator close_to is defined using estimations of typical deviations of temperature parameters in the region.

Mapping from the CIMP to LIMP model means that it is necessary to define process patterns for selected technologies. Rules for selecting process patterns are not fixed and are formulated in terms of quantity characteristics of data sources and requirements of various groups of end-users. Quantity characteristics can be transformed into quality characteristics only in application to a concrete data source or end-user group.

When mapping from LIMP to SMP, for each group of algorithms defined in LIMP, one of algorithms is to be selected. Business rules that are used for selecting algorithms are the most complicated because the results of data feature estimation should be considered when algorithms are defined. For example, consider the task of solving measurement time series interpolation to standard depth levels. Measurement time series are usually complicated time series that contain several different components, so the application of one algorithm will provide unsatisfactory results. To solve this problem, it is reasonable to apply different algorithms for different components of the time series. Before the application of business rules, a set of specialized procedures that depend on features of processed data have to be called. Results of data estimation are used in business rules.

Business rules are formulated by subject domain experts using specialized editors or with the help of associated mining algorithms that are aimed to find dependencies in historical data. The application of business rules is organized using an inference engine.

Library of explorationary analyses, data classification, and estimation algorithms. The library is aimed to provide additional information about data that is processed for selecting appropriate elements when building processes. The library consists of three main components.

The first component contains a set of various tests, including tests for data quality control, for defining data type, for finding out if the data has a defined set of features, etc. Results of test executions are binary—tests are passed or tests are not passed. As a rule, for obtaining additional information about data, a sequence of tests is executed. For example, estimation of oceanographic data quality assumes execution of 15 tests on average.

The second component is for calculation of the statistical characteristics of data. For statistical measurement analysis, various methods can be used, such as those described in Rao et al. (2012). The set of calculated characteristics depends on the type of data. For defining the type of time series, the following characteristics can be calculated: median, standard deviation, variability, range, error of piecewise constant approximation, error of piecewise linear approximation, error of interpolation using polynomials or splines, complexity of the curve, change points, etc. These characteristics can also be used in business rules or as a feature space for further data processing.

The third, principle component is the intelligent analysis method. The most widely used groups of methods for intelligent processing and analysis of multidimensional measurement are the following: segmentation, cluster analysis, classification methods, methods of sequential analysis, and association rule mining. Descriptions of principles and application of data mining techniques are given in Vasiliev et al. (2007) and Zhukov (2006). The main useful features of data mining algorithms are the following:

1. Data mining algorithms provide formalized descriptions of data, identification of useful patterns, and dependencies in data based on analysis of all available historical data using mathematical methods. Analysis of different aspects of data, both independently and together, allows complex interconnections in data to be revealed. By using the results of data mining algorithms, statistical models of the statistical description of data can be built. The main advantages of statistical models are the relative simplicity of construction, flexibility, and high performance of application. Statistical models can estimate the dynamics of changes in object states, taking into account external factors that influence them. Comparing new data to statistical models gives information that is useful for building processes.

2. In processing automation, the important roles are using the results of analysis of processing execution history and history of system usage. Use of knowledge about earlier executed processes can significantly simplify the task of processing of newly received data. Intelligent methods allow data processing history to be analyzed and knowledge about efficient and inefficient processes to be extracted.

5 Description of Principles of Integration of Components of Measurement Intelligent Analyses into IGIS Systems

The use of adaptive processing of multidimensional measurements has opened entirely new opportunities for obtaining, processing, and presenting data in information systems. Such processing allows the consideration of not only particular problems of data analysis, but also complex statistical analysis of information received from heterogeneous sources. Obviously, the processing of multidimensional measurements is not the end task. It is a part of the process of solving tasks of higher levels. All this requires a tight integration with a range of other systems—with systems that provide data on hydrographical, hydrometeorological, tactical and other conditions; with systems that consume analysis products; and especially with systems with GIS interface components. This integration requires program and information exchange of measurement analysis components with external information systems.

According to the theory of information systems, a component in measurement analysis should be considered as an open system that is a part of information analysis system. To meet the requirements of open information system architecture and information management of measurement analysis, the component should support a single format of information exchange with other systems. Here, the format that is defined in unified information model (Pankin et al. 2006) is proposed for use.

The model provides harmonization of information of the measurement analysis component and other systems or subsystems, uniformly getting the features and characteristics that are required in the particular application for processing and analysis.

The model is implemented using multilevel metadata model concepts and is built on the basis of ontology. Before an object of a class of a unified information model is created, it is necessary to describe the class of the object, including a description of how many properties objects of this class will have, what type these properties are, the possible range of properties values, etc. Thus, class specification is metadata of objects from an object information model. This metadata is used to create specific program objects, assign specific values to properties, etc.

The model assumes realization of the following functions (Pankin et al. 2006):

- description of all categories of domain objects, which is used in information exchange with other systems or subsystems;
- support of universal mechanism of relations, allowing implementation of all kinds of dependencies between different information objects contained in data;
- selection of required information for different users;
- usage of the same data to solve various problems of analysis.

For all information, the same model is used. This model relies on the tree of inherited classes and the direct objects of these classes. Each class can have a set of properties with default values. Each class has only properties that distinguish it from its parent.

Subject domain objects are data models of real objects and are instances of domain classes. Each class can have a set of its instances—a subject domain object. Each object may inherit multiple classes. This makes it possible to integrate multiple sources of information and, at the same time, to use the same data for solving various problems. When a domain object is created, it receives all properties of classes, which are ancestors of the object. Moreover, properties are added sequentially from the root class to the direct ancestor class, so values of overridden properties are never lost. Using the class hierarchy and domain object description mechanisms, it is possible to create, store, modify, and delete information that describes the domain of all integrated components.

The relationship mechanism allows implementation of any kind of dependencies between different information objects contained in the data and makes it possible to describe properties that arise from the interaction of these objects, including system properties. Furthermore, through the relations mechanism, it is possible to indicate what algorithm and under what circumstances should be used for processing data obtained as a result of integration.

Various components must have access to a limited set of objects and their property values. This makes it possible to significantly reduce the flow of information between components and to improve information security by distinguishing access to data.

Thus, usage of the unified information model allows a system to be built using components that provide data processing and analyses with a given level of complexity, taking into account all necessary information available to the system.

6 Description of Oceanographic Data Processing System and Results of Experiments

The proposed approach to adaptive processing and analyses of measurements was used for processing and analysis of oceanographic data in a geographic information system of condition lighting (Попович et al. 2006). The condition lighting system provides end-users with integrated information about situations and the state of the environment in the interests of making justified decisions for a particular situation. The system provides access to geospatial and meteorological data, information about technical and environmental objects, data obtained from external sources, and provides the solution for mathematical problems (problems of the theory of search, sonar, radar, etc.) and modeling tasks. Extending the functionality of the system associated with the tasks of processing and analysis of oceanographic data is indicated by the necessity for performing sonar data calculations, in particular calculations of water sound speed, in which temperature and salinity parameters are used. The main source of oceanographic data is the most extensive network of oceanographic drifting buoys (Project Argo).[4] An additional source is measurements obtained from other networks and drifting floats, moorings, bathythermographs, and autonomous unmanned underwater vehicles (Ивакин et al. 2011). Components that implement algorithms for processing and analysis of oceanographic data and building grids for ocean water parameters are integrated into the system of condition lighting.

Performing processing and analysis of oceanographic data in order to build regular data grids includes two main steps: data verification and data regularization. The main purpose of the data verification step is systematic storage, analysis, and processing of data in order to prepare them for solving the problem of building data grids (Boyer et al. 2005; Zhuang et al. 2011). The main objective of the regularization stage is to build a regular grid using acquired measurements and estimate the accuracy of gridded data (Кораблев et al. 2007).

The data verification stage includes a set of substeps, the composition and objectives of the stages, as well as a list of used algorithms (Fig. 5). Harmonization of data includes specialized algorithms, each of which converts data from one or multiple formats into a format of the unified information model. Integration algorithms are aimed to merge data received from different sources. Data fusion involves building statistical models for various areas and data sources.

Data mining algorithms are applied to both the measurement verification step and measurement regularization step. The application of cluster analysis algorithms allows the identification of areas in which there are no abrupt changes and where it is possible to use algorithms of joint processing (substeps 2.1, 2.4, 3.2). The set of segmentation algorithms and algorithms for building patterns provide estimation of measurement time series similarity when identifying duplicates (step 2.2); algorithms for building measurement images are used to compare time series at the step of measurement quality estimation, based on historical data analyses (step 2.4). During the preprocessing of measurements, measurements are divided into segments for increasing the effectiveness of algorithms. Intellectualization of data fusion algorithms is, first of all, based on the application of intelligent

[4] http://www.argo.ucsd.edu/

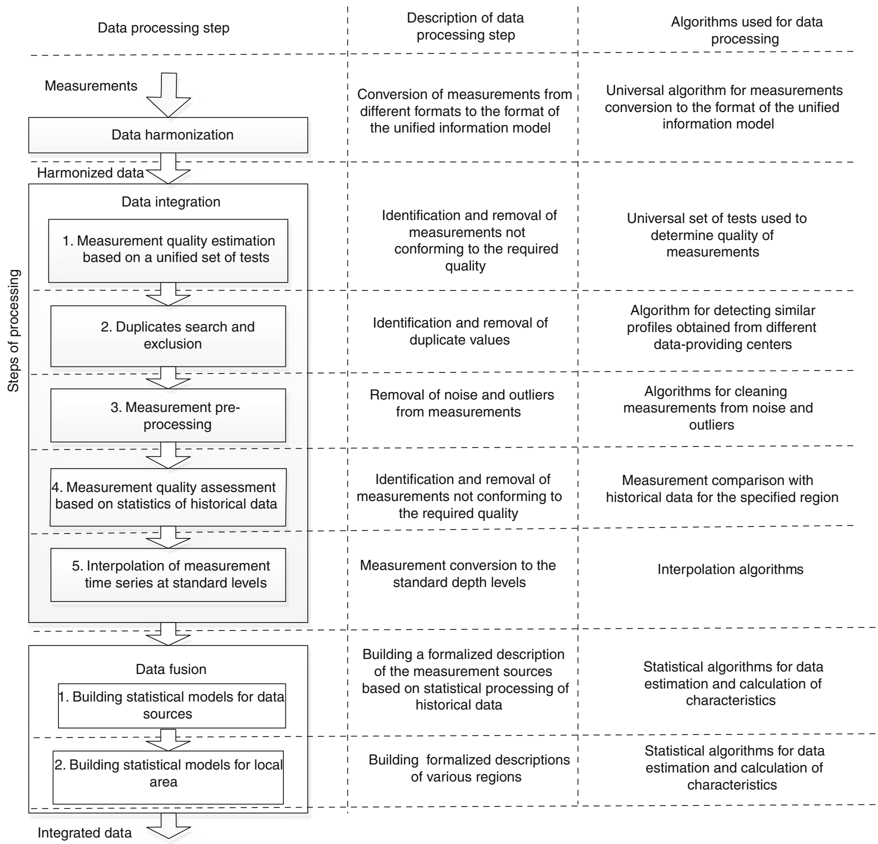


Fig. 5 Data verification step of ocean data processing and analyses

processing techniques that provide a formalized description of a measurement set that presents both standard behavior and deviations from it.

The step of data regularization involves building a general statistical model for the historical measurements, building a regular grid based on gathered statistics, and renewing the regular grid using reanalysis procedures. At the stage of regularization, data mining algorithms are used to solve two problems—the problem of building a general statistical model and the problem of improving the regular grid. The first problem is solved using algorithms for building a joint formalized description of measurements and algorithms for identifying dependencies in the data. The second is solved using the algorithms that determine the influence of the measurements' values at a single point on the measurements' values in a neighborhood.

7 Conclusion

The proposed approach to processing and analysis of multidimensional measurements using IGIS technology allows implementation of adaptive processing and analysis of parameter measurements of geospatial objects using artificial intelligence and algorithms for explorational analyses. Use of the adaptive approach provides a way to process data from new data sources, use new algorithms, and solve new tasks without making changes in the information systems at the level of program code.

It is necessary to note that the problem of dynamically building adaptive processes by sequential selection of the elements of adaptive processes at different abstraction levels (technologies, process patterns, processes) cannot be solved completely. This is due to the fact that the selection of the elements of adaptive processes is influenced not only by the quality, structure, and volume of data being processed, but also by intermediate results obtained during the processing. Therefore, when building adaptive processes, the following points are important:

- extend the performed processes of measurement analyses with processes that acquire necessary information about data that are being processed while executing exploratory data analysis algorithms;
- provide the possibility to rebuild processes if the resulting user task will not satisfy the requirements for quality or processing time according to analyses of intermediate results. A process can be rebuilt starting from any level of abstraction;
- consider all factors that can influence the decision when choosing elements of adaptive processes. If one or several factors cannot be estimated, then the feature space of alternative is incomplete. There are also situations when it is important to define probabilities for different factors. In these cases, decision-making methods for conditions of uncertainty can be used.

References

Васильев АВ, Геппенер ВВ, Жукова НА, Тристанов АБ (2007) Методы обработки телеметрической информации на основе алгоритмов Data Mining//Известия СПбГЭТУ “ЛЭТИ” (Известия Санкт-Петербургского Государственного Электротехнического университета). Сер. Информатика, управление и компьютерные технологии. Спец. выпуск. – СПб., – Вып. 1

Васильев АВ, Геппенер ВВ, Жукова НА, Тристанов АБ (2007) Применение алгоритмов кластеризации и классификации в задачах обработки и интерпретации телеметрической информации//Труды Российского научного-технического общества радиотехники, электроники и связи имени А.С. Попова. Сер. Цифровая обработка сигналов и ее применение. – М.: ИПРЖР, – Вып. IX-2

Boyer T, Levitus S, Garcia H, Locarnini RA, Stephens C, Antonov J (2005) Objective analyses of annual, seasonal, and monthly temperature and salinity for the world ocean on a 0.25° grid. Int J Climatol 25:931–945

Hammoudi S, Alouini W, Lopes D, Huchard M (2010) Towards a semi-automatic transformation process in MDA: architecture, methodology and first experiments. Int J Inf Syst Model Des (IJISMD) 1(4):48–76

Kleppe A, Warmer J, Bast W (2003) MDA explained, the model driven architecture: practice and promise. The Addison-Wesley Object Technology Series, Addison-Wesley Professional

O'Rourke C, Fishman N, Selkow W (2003) Enterprise architecture using the Zachman framework, course technology

Pankin A, Popovich V, Ivakin Y (2006) Data for GIS. CORP2005, University of Technology, Vienna, pp 302–310

Pankin A, Zhukova N, Vitol A (2013) Model for knowledge representation of multidimensional measurements processing results in the environment of intelligent GIS. Conceptual structures for STEM research and education. In: Proceedings of 20th international conference on conceptual structures, ICCS, Mumbai, India, 10–12 Jan 2013

Popovich VV, Claramunt C, Schrenk M, Korolenko KV (eds) (2009) Information fusion and geographic information systems: proceedings of the 4th international workshop, (lecture notes in geoinformation and cartography), Springer, Berlin, 17–20 May 2009

Rao TS, Rao SS, Rao CR (eds) (2012) Handbook of statistics: time series analysis: methods and applications. North Holland

Zhuang SY, Fu WW, She J (2011) A pre-operational three dimensional variational data assimilation system in the North/Baltic Sea. Ocean Sci 7:771–781

Жукова НА (2006) Использование алгоритмов ассоциации в интеллектуальных системах обработки телеметрической информации//Десятая национальная конференция по искусст, венному интеллекту с международным участием КИИ-2006: Сб. научн. трудов; г. Обнинск, 25-28 сент. 2006 г

Ивакин ЯА, Жукова НА, Панькин АВ (2011) Интеллектуальная геоинформационная система мониторинга дна моря. Материалы четвертой всероссийской научно-технической конференции « Технические проблемы освоения мирового океана » , 3–7 октября 2011г. Владивосток, - С. 20–25

Кораблев АА, Пнюшков АВ, Смирнов АВ (2007) //Создание океанографической базы данных для мониторинга климата в Северо-Европейском бассейне Арктики//Труды ААНИИ. Т. 447. - С. 85–108

Попович ВВ, Потапычев СН, Панькин АВ, Шайда СС, Воронин МН (2006) Интеллектуальная ГИС в системах мониторинга//Труды СПИИРАН. Вып. 3. Т. 1. С. 172–184

Part V
Data Security for GIS

Logical Inference Framework for Security Management in Geographical Information Systems

Igor Kotenko, Olga Polubelova, and Igor Saenko

Abstract A promising direction of research to ensure security in large-scale information systems, including distributed geographic information systems (GISs), is the development of software tools that implement logical inference based on knowledge about security information and events is frameworks that use logical languages and inference provide administrators with powerful and flexible means to verify complex security policies, generate efficient countermeasures against computer attacks, and maintain the required security level. This chapter outlines an approach for the development and implementation of a logical inference framework for security information and event management. The chapter considers the common architecture of this framework, as well as the architecture and implementation details of particular logical inference modules based on event calculus, model checking, and an ontological data repository.

Keywords Information security · Security information and event management · Logical inference · Ontology · Data repository · Event calculus · Model checking

1 Introduction

Security systems are critical elements for the functioning of large-scale geographically distributed information systems, including distributed geographic information systems (GISs), for several reasons. First, distributed GISs consist of a

I. Kotenko (✉) · O. Polubelova · I. Saenko
Saint-Petersburg Institute for Informatics and Automation of Russian Academy of Sciences (SPIIRAS), 39, 14th Liniya, Saint-Petersburg, Russia
e-mail: ivkote@comsec.spb.ru

O. Polubelova
e-mail: ovp@comsec.spb.ru

I. Saenko
e-mail: ibsaen@comsec.spb.ru

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_13,

large number of workstations, servers, network devices, etc., which are mutually linked with each other through computer networks, and, possibly, with other open public networks. Secondly, these objects of GIS infrastructures are constantly threatened by fairly large numbers of new penetration means with increased power to compromise information resources.

Many GIS platforms have developed means for information protection. For example, ArcGIS, which is an Enterprise GIS platform, has the following security mechanisms: authentication, authorization, crypto-protection, filtering, and logging. The last mechanism, logging, is very important for security and reliability of GIS applications, because it allows the detection of suspicious activity, which frequently provides early indications of system attacks. A possible research direction for logging is the creation and application of the next-generation software frameworks, carrying out constant monitoring and security management based on the analysis of security information and events. These data are generated in computer networks of the distributed GIS by multiple sources (web servers, operating systems, relational databases, etc.) and are transferred to the data repository of the security management system for subsequent storage and analysis (Miller et al. 2011).

In large-scale information systems, such as distributed GISs, there are requirements for the analysis of security events, such as the ability to handle incomplete and contradictory information, extraction of rare events, cross-level correlation of network events and business-process events, heuristic prognostics, etc., which have to be implemented in real or near-real time. A generally accepted way to meet these requirements to develop the subject domain-oriented logical inference systems based on advanced logical languages. In addition, the importance of implementing these requirements is stipulated by the need to build efficient data repositories for storage of security information and events. The trivial integration of existing logical inference systems with data repositories, in our view, cannot be considered to fully meet these requirements. This is largely due to the limited capabilities of used logical inference languages and security data modeling tools. This limitation was the main motivation for our investigations presented in the chapter. The main contribution of the chapter is the implementation of a logical inference framework for security management, which can be used in large-scale GISs.

For the logical languages, we selected event calculus and temporal logic (model checking). Analysis of security applications of these languages shows that they are, in our view, the most suitable for security information and event processing. Furthermore, we propose an ontological approach as it is a new solution used in the logical inference framework for implementation of the data repository. The main result of this approach is a hybrid data repository that is capable of representing security information and events in the form of relational data, XML data, and triplet storage.

The rest of the chapter is structured as follows. Section 2 provides an overview of the related work. Section 3 discusses the general architecture of the framework proposed. Sections 4 and 5 outline the issues related of logical inference modules based on event calculus and model checking, respectively. In Sect. 6, we discuss our proposals concerning the ontological approach for the security data repository. Section 7 presents the aspects of practical implementation of the logical inference framework. Section 8 concludes our work and contains the directions for further research.

2 Related Work

Logical inference systems are successful in security areas, which are associated with the analysis of safe states, events, and security policies. Thus, logical inference systems can be used successfully for event correlation, security information analysis, policy verification, and attack modeling. Let us mainly outline two types of languages and inference mechanisms which, according to our analysis, can be successfully used for the tasks of security information and event management.

Event calculus originally was defined in Kowalski and Sergot (1986) as a set of Horn clauses expanded by the rule "negation as failure"—meaning that the inference of the assertion is to prove the impossibility of logical negation of that assertion. As the result, event calculus belongs to a class of automatic theorem-proving systems.

Spanoudakis et al. (2007) proposed using event calculus to monitor the security requirements through the development and application of templates expressing requirements for confidentiality, integrity, and availability. Broda et al. (2009) propose the use of event calculus together with the forward-chaining deduction to map lower-level time-stamped sensor reading events into inferred output high-level compound events using specialized agents for each type of sensors. Farrell et al. (2004) used event calculus to deal with performance monitoring of service-level agreements. In addition, they proposed an XML-formalized ontology, which is intended to facilitate the automated monitoring of contract states for performance monitoring.

The application of event calculus for intrusion detection is closely linked to security monitoring. Evidence of this is provided in another work by Spanoudakis (Amalio and Spanoudakis 2008), in which a pattern-based approach to the run-time monitoring of security and threat detection requirements is presented. From the templates, the formulas in event calculus are generated. In Babamir and Jalili (2006) a logical-based approach to detect software intrusions is proposed, which realizes the abductive inference to retrieve the event calculus formulas from the specified tables.

A large number of works devoted to the use of event calculus are in the area of security policy verification. Bandara et al. (2003) suggest a method to transform the authorization and management policies, as well as the specifications of the system behavior, into the formal notation based on event calculus. Tishkov et al. (2005) and Kotenko et al. (2007) presented the architecture of a security checker intended to detect and solve the security policy conflicts. Conflict detection is demonstrated through the use of event calculus. Evans et al. (2010) considered the use of event calculus for specifications of security policies in electronic systems, which are often not able to provide an easy way for encoding requirements due to the need to take into account the interaction of the physical principles. Gaaloul et al. (2011) considered the possibility of applying event calculus to solve the problem of authority delegation in the electronic document management. Rouached and Claude (2006) presented a formal event calculus based model for the dynamic consolidation of security policies. The authorization of the Web service composition is considered as the case study for the proposed model.

The control of business process restrictions is an important area for security management where event calculus can be applied. Tsang et al. (2013) offered to formalize the dynamics of market behavior through simple market models based on event calculus. Montali et al. (2011) considered the task of monitoring business restrictions using event calculus and offered an event calculus oriented monitoring platform, which traces events and evaluates business restrictions.

The main purpose of *temporal logic* (Pnueli 1977) is to describe the properties of systems by using temporal formulas. This approach allows verification of the maintenance of desired properties and the existence of undesired properties. An approach to verification based on temporal logic called *model checking* is considered, for example, in Lu (2001). This approach allows for a given model of the system behavior with a finite number of states to check whether some logical condition in these states is performed or not. The task of the researcher in this approach is to develop an adequate system model and correctly formulate security requirements in a temporal logic language. In Winwood et al. (2006), a temporal logic is applied to implement an approach to security policy verification, based on 'proof carrying code'. This approach extends the range of checked security policies. Dixon et al. (2004) proposed an approach that uses a combination of temporal and epistemological logic. It can specify the systems that require one to simultaneously take into account the dynamic and informative aspects of knowledge.

In Das and Niyogi (2011), a theoretical platform for intrusion detection based on the monitoring of temporal logic specifications is offered. The idea is to specify various attacks against a system as temporal formulas and to monitor the system to detect violations of those formulas. In Simko and Sztipanovits (2012), it was proposed to use metric temporal first-order logic for monitoring. The monitoring system was developed on the basis of this class of temporal logic. It can integrate event streams and track the existence of temporal invariants. The basic idea of such online monitoring is the gradual building of internal representation of previous states without keeping all unnecessary details.

It is also possible to use the temporal logic for attack modeling and simulation. In Nowicka and Zawada (2006), the modeling of different temporal relationships between security events is considered to manipulate multi event attack signatures. The proposed formal conceptual model is based on interval temporal logic. It is shown how the temporal properties of multi event attack signatures can be expressed within this model.

Thus, the analysis of the related work shows that the logical inference methods considered (event calculus, temporal logics) have a wide application in modern academic research in the field of information security and can be successfully mapped into the logical inference framework for security management in GISs.

3 General Architecture

The architecture decisions for our logical inference framework development include the general architecture, the architecture of inference modules (event calculus and model checking), and the ontological data model. The ontological data model allows one to perform the effective logical inference over the data stored in the repository. The event calculus module can be used to analyze the behavior of GISs. The model checking module allows one to analyze the rules of security policy (for example, filtering rules). Figure 1 presents the general architecture of the inference system combining these three approaches. The inference system is considered as a superstructure over the repository.

The task of verification of role-based access control (RBAC) rules is chosen in the use cases for implementing the event calculus module in the logical inference system. It is suggested to conduct such verification every time after the modification of authorization rules to distribute the access permissions inside the GIS security management system. The verified rules are loaded to the repository and are further used by the RBAC subsystem.

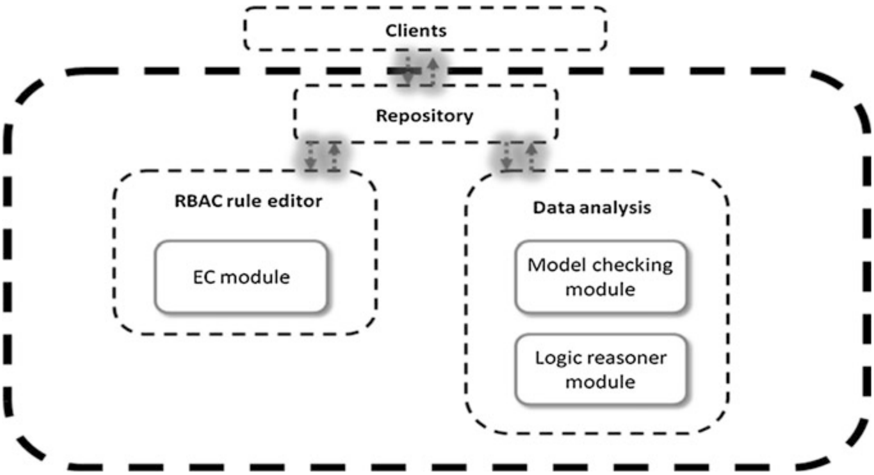


Fig. 1 General architecture of the logical inference system. EC, event calculus; RBAC, role-based access control

The model checking module can be applied for the data analysis, such as for the security policy and network traffic routing analysis. The input data for the module are selected from the events that are stored in the repository. These data are converted to the internal format of the module. The results of the module functioning—detected anomalies—are saved in the repository.

The logic reasoner, which uses the logical inference system, works through the repository application server (RAS). Exchange of requests and responses with clients is implemented via the simple object access protocol (SOAP). The ontological model of the component data was developed to implement this functionality.

4 Event Calculus Module

The generalized representation of the event calculus module is presented in Fig. 2. The input parameters in the given module are the computer system description, the policy description, and the anomalies (conflicts) description. For the verification, results, the module outputs "YES/NO", the conflict information, (including conflict type and conflicting rules), and the modified rules that represent the policy without conflicts. The module uses the CIFF (Endriss et al. 2004) abductive proof procedure, implemented in the CIFF 3.0 application. This implementation uses SICStus Prolog.

The main theoretical problem in the given module implementation is the development of the domain-dependent event calculus axiomatics. This axiomatics includes formalization of all input data: the system description on the system description language (SDL), security policy description on the security policy language (SPL), and anomalies (conflicts) definition. SPL rules are translated to the axiom set that shows security system reaction to events that occurred in the network. Therefore, inaddition to policy rules, there are events that activate a system action.

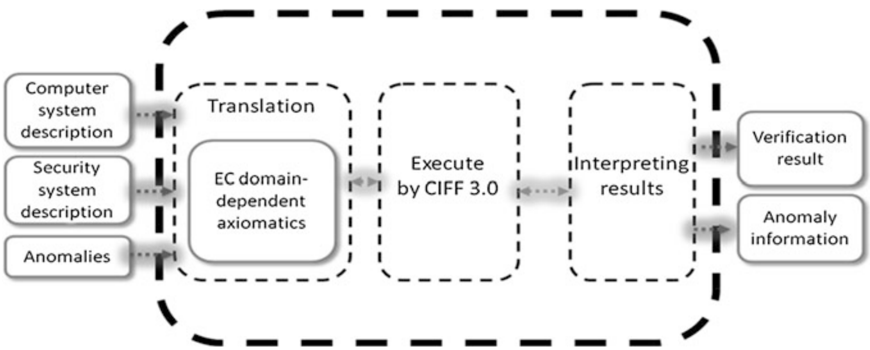


Fig. 2 Architecture of the event calculus (EC) module

The main verification principle implemented in this module is the following. The system behavior can be expressed in domain-dependent axioms based on fluent whose states are changed by event occurrences. The abductive proof procedure takes as an input the axiomatic and the query that represents conflicting system state and outputs the number of events that can lead to this system state. If the scenario is found, rules that participated in this scenario are analyzed.

5 Model Checking Module

The generalized representation of the model checking module is presented in Fig. 3. The input parameters and results for the given module are the same as in the event calculus model. The main theoretical problem of the module implementation is the construction of such model that can simulate the behavior of the network, the security system, and the subjects, which make requests for the network resources. The security policy verification consists in the model checking of the constructed model and further analysis of the output. Inconsistencies are expressed in the linear temporal logic and added to the model as conditions for consistent states. If some condition is failed, the module provides the inconsistency type and the sequence of events that lead to the inconsistent state. Then, this trace is analyzed in order to define the contradictory rules involved, and one of the appropriate resolution strategies is applied.

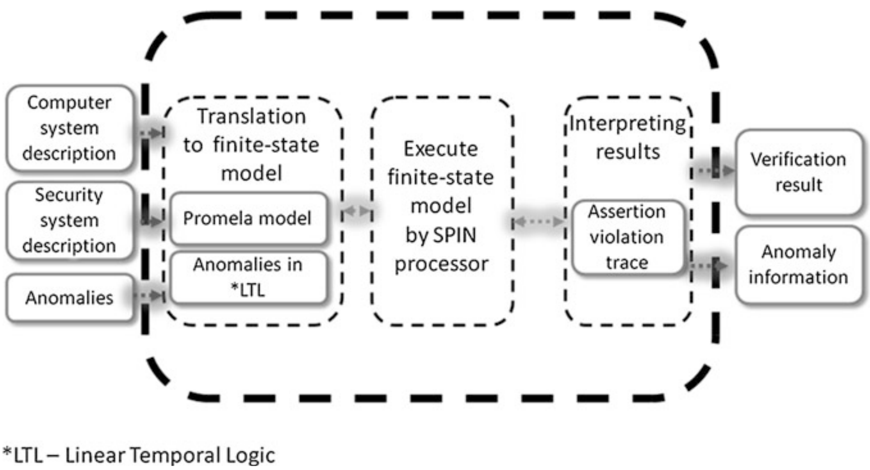


Fig. 3 Architecture of the model checking module

6 Ontological Repository

Frequently, it is difficult to describe the logic of relations between entities using the relational representation. In the process of model construction, a large amount of tables are created. Queries for such data can be very resource intensive. Some data cannot be represented via the relational algebra at all. These data are stored in the database in a format that requires additional processing. The set of samples from the database in this case can be very big, and the program running time increases significantly. For example, in the process of vulnerability description, the list of products that can result in the vulnerability occurrence is formed with logical operators OR and AND. This list of products can have any length.

In the relational data model, the list of all products that describe vulnerability, along with logical operators, is stored as a row in the table. Such representation does not allow one to specify a parameterized query that defines product names, versions, etc. It is necessary to load all vulnerabilities and process them in the program. This process takes a lot of time.

Figure 4 presents the ontology, which describes the concepts of vulnerabilities, software/hardware, vendors, etc. In this model, the relations between software products and hardware components (the combination of which leads to the vulnerability occurrence) were specified via the description logic. Relations between concepts are described via subclasses instead of the characteristics of objects. Thus, the logical inference consists of the classification problem and the speed of processing increases.

This approach to the vulnerability representation allows a significantly lesser volume of data to be loaded from the repository. It also gives the possibility to avoid the program processing, because the task of analysis is reassigned to the logical inference system.

7 Implementation

In this section, we outline the software implementation of the particular inference modules considered in the previous sections. These prototypes implement approaches to data analysis that use event calculus and temporal logic (model checking) for the objectives of security management in GISs.

7.1 Event Calculus

Event calculus formalizes the common-sense principle of inertia: "normally, nothing changes." *Fluent* (time-varying property of the world) holds at particular *time points* if it was initiated by an *event* occurrence at some earlier time point and was not terminated by another event occurrence in the meantime.

Fig. 4 Ontological model for vulnerability representation

Similarly, a fluent does not hold at a particular time-point if it was previously terminated and was not initiated in the meantime. Event calculus is a first-order language with fluents, events, and time points as sorts.

An example of authorization conflict detection for the RBAC model is presented here.

Authorization policy is defined with quadruple: "subject-action-object-access permission" (`allow/deny`).

The subject can be represented by a role or a specific user. The object can be composite and contain the collection of subobjects. A conflict appears if for the same triple "subject-action-object" allow and deny are defined or deduced simultaneously.

On the set of roles and users, a hierarchical structure with the relation of partial order is defined. This relation defines the priority between two roles or belongs of the user to the role. Authorization rules are considered for each object and each action in which these object and action participate.

Each rule assigns permission or prohibition to one of the nodes of the role hierarchy graph. If the node is not assigned with any rule, then the access permission is deduced from the values of upper nodes. Each from these algorithms can lead to the contradiction. Such contradiction is called authorization conflict: when for the same triple subject-action-object the contrary privileges are deduced.

For example, an authorization conflict can occur if the following conditions are met: there are two rules with the same "object-action" pair; roles used in these rules are different; or permissions defined for the roles are opposite.

Belongings of the user of the role and the role hierarchy are defined with the `subject_role` predicate. The hierarchy of roles and users (Fig. 5) is formalized as defined in Listing 1.

Listing 1. Definition of the initial state of the dam control system

```
subject_role(U, R) :- R = employee, subject_role(U, adm-staff).
subject_role(U, R) :- R = adm-staff, subject_role(U, operator).
subject_role(U, R) :- R = operator, U = mary.
subject_role(U, R) :- R = adm-staff, subject_role(U, dean).
subject_role(U, R) :- R = dean, U = john.
subject_role(U, R) :- R = adm-staff, subject_role(U, chair).
subject_role(U, R) :- R = chair, U = tom.
subject_role(U, R) :- R = employee, subject_role(U, security-officer-staff).
subject_role(U, R) :- R = security-officer-staff, subject_role(U, chair).
subject_role(U, R) :- R = security-officer-staff, subject_role(U, security-
officer).
subject_role(U, R) :- R = security-officer, U = tom.
subject_role(U, R) :- R = security-officer, U = nick.
subject_role(U, R) :- R = security-officer, U = david.
subject_role(U, R) :- R = security-officer-staff, subject_role(U, assistant).
subject_role(U, R) :- R = assistant, U = alex.
subject_role(U, R) :- R = assistant, U = nick.
```

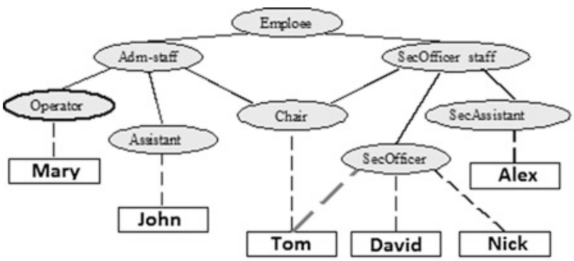


Fig. 5 Role hierarchy in the event calculus module

The authorization conflict occurs, for example, if `tom` gets "deny" according to the role `security-officer` and "allow" according to the role `Adm-staf` on some action over some resource.

The subject axiomatic contains the fluents `authorization_allowed` and `authorization_denied`. This set of fluents should not be satisfied simultaneously for the same set of characteristics `User`, `Activity` and `Target`. This set of fluents is initiated with the event `authorization_request`. Initially, we consider that fluents are not satisfied for any characteristics (Listing 2).

Listing 2. Definition of the initial state of the dam control system

```
initially_false(F) :- F = authorization_allowed(User, Activity,
   Target, RuleName).
initially_false(F) :- F = authorization_denied(User, Activity,
   Target, RuleName).
```

Fluents and the event `request_authorization` are connected with the predicate `initiates`. For example, for the role `adm-staff`, the allowing privilege to write on the FTP server is specified as shown in Listing 3.

Listing 3. Allowing privilege to write on the server

```
initiates(E,F,T) :- subject_role(User,adm-staff), E =
   request_authorization(User, write, server), F =
   authorization_allowed(User, write, server, fTPWriteAdmRule).
```

Privileges for other roles and users are defined similarly.

Detection of the authorization conflict is performed with the `ciff` request (Listing 4).

Listing 4. Ciff request

```
?-run_ciff([policies],
   [holds_at(authorization_allowed(User,Action,Target,AllowRuleName),T),
   holds_at(authorization_denied(User,Action,Target,DenyRuleName),T)], A).
```

This request finds out if the situation, when the fluents `authorization_allowed` and `authorization_denied` are satisfied simultaneously for the same set of characteristics `User, Action, Target,` is possible. The fourth parameter of the fluent (`AllowRuleName` and `DenyRuleName`) allows defining the rules that resulted in the authorization conflict.

Listing 5 outlines the response from SICStus Prolog, which contains the substitutions for all transferred variables. It includes the values for the variables `User, Action, Target, T, AllowRuleName, DenyRuleName`.

This verification mechanism allows detecting authorization conflicts for increasing the security level of the RBAC system. It provides the correct control of access rights to resources.

Listing 5. Response from Prolog

```
Action = write,
AllowRuleName = fTPWriteAdmRule,
```

```
DenyRuleName = fTPWriteSEcRule,
User = tom,
T = _63765,
Target = server.
```

7.2 Model Checking

Usage of temporal logic is shown in the example of analysis of filtering rules. We use the SPIN model checker as a verificator. Verification of the filtering rules is related to the category of analysis of network traffic routing. It allows detecting different anomalies of filtering. We consider the contradictions and anomalies inside an access control list (shadowing, generalization, correlation, redundancy) and between the lists of different firewalls (shadowing, fictiousness, correlation, and redundancy) suggested in Al-Shaer et al. (2005).

Inputs in the model checking verificator are the formal system specification and the formal requirements specification. The formal system specification is described with Kripke structures. The formal requirements specification is defined with linear temporal logic formulas (Manna and Pnueli 1995).

The computer network model represents a parallel system with a finite number of states. It is defined with the Kripke model. Specification of the modeled system looks like a set of temporal formulas. These formulas express a required system behavior. They will express the absence of anomalies in filtering rules. So, the verification task consists of checking the satisfiability of these formulas on the Kripke model.

The suggested *model of the computer network* is intended to represent the network structure, its main components, and network processes. It includes two basic components: the network topology model and the network traffic model. For the verification of filtering rules, it is necessary to outline the position of hosts and firewalls in the common network topology. In the process of traffic generation, all address space is decreased to a minimum, which is necessary to detect all possible anomalies. Only the hosts specified in the filtering rules are used.

The firewall model is used to represent the firewall and the algorithms of its functioning. The main components of this model are the network identifiers, the defined sets of filtering rules, and the algorithm of network traffic processing.

Contradictions or anomalies between access control lists are caused by the intersection of the rules conditions from different access lists (shadowing, factiousness, correlation, redundancy).

The main phases of the method proposed are shown in Fig. 6. The input for the method includes three kinds of specifications: the specifications on the SDL for network topology representation; the specifications on the SPL for filtering rules definition; and the anomalies taxonomy.

The black circle in Fig. 6 is the finite state of the model.

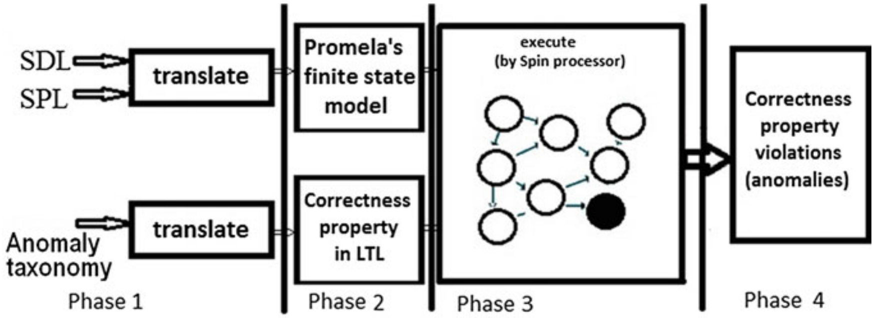


Fig. 6 Verification phases LTL. SDL, system description language; SPL, security policy description

In *the first phase*, the input data are translated to the internal format of verification system Promela (Process or Protocol Meta Language). Promela is a verification modeling language used in SPIN.

In *the second phase*, the common verification model (CVM) is designed. It is a finite state model consisting of the computer system, the firewall, and anomalies models, which are initialized by data in an internal format.

This model should provide mapping of the computer network; generating the set of network address; network traffic in the defined range of network addresses; network data processing with accumulation of statistics; and network statistics analyzed in order to detect anomalies.

Generated network addresses should cover the complete range of network addresses defined in the rules. Anomalies in the model are expressed as assertions. Those assertions are the properties of the CVM. Their violations lead the CVM to an incorrect state. The peculiarity of this work is the ability to use temporal logic to specify temporal parameters in filtering rules that set constrains to the network traffic, taking into the account the time, the day of week, etc.

In *the third phase*, the CVM is verified with special program tool that implements model checking. In this work, it is SPIN. In the verification process, all incorrect system states are detected.

In *the last phase*, the obtained verification results are interpreted. If anomalies have been detected, the user gets the firewall address, the rules leading to appearance of anomalies, and the anomaly type.

8 Conclusion

The main goal of the chapter was to develop recommendations for using the logical inference framework, which allows solving the tasks of comprehensive analysis of the security information and events stored for security management in GISs. The languages and tools of the temporal logic, event calculus, and the description logic (for ontologies) were selected as the main directions of the research and development. The choice was stipulated by the fact that these languages and tools are currently being actively developed and researched, they and demonstrate acceptable functionality for the tasks of security information analysis.

We analyzed the related work in using logical inference mechanisms for security information processing. This analysis demonstrated that these tools can be successfully used in different security areas that are important for the security management in GIS. We demonstrated that model checking allows to analyze security policy rules (for example, filtering rules), event calculus can be used to analyze the behavior of critical infrastructure systems, and the ontological data model allows performing effective logical inference over the data stored in the repository.

We considered two implemented software prototypes to show for each language its possibilities for security information processing. By using the temporal logic approach, we developed the module of filtering policy verification. The event calculus technique was demonstrated for analysis of RBAC authorization rules, which allows the correct distribution of resources between user roles. We also suggested architecture solutions for using logical inference as an additional layer for realizing the common repository in the security management system. As an example of using ontologies for logical inference, we demonstrated the vulnerability data model. The software implementation of proposed approach may be included in the arsenal of information security means available to the network security administrator. It may be actively used to create dynamically configurable schemes of custom access to network resources. In future work, we are planning to extend the approaches suggested and experiment with them.

Acknowledgments This research is supported by grant from the Russian Foundation of Basic Research, Program of Fundamental Research of the Department for Nanotechnologies and Informational Technologies of the Russian Academy of Sciences (contract #2.2), State contract #11.519.11.4008, and partly funded by the EU as part of the SecFutur and MASSIF projects.

References

Al-Shaer E, Hamed H, Boutaba R, Hasan M (2005) Conflict classification and analysis of distributed firewall policies. J Sel Areas Commun 23(10):2069–2084

Amalio N, Spanoudakis G (2008) From monitoring templates to security monitoring and threat detection. SECURWARE '08. 2nd International Conference on emerging security information, systems and technologies, pp 185–192 25–31 Aug 2008

ArcGIS Resource Center (2013). Logging and Auditing. http://resources.arcgis.com/content/enterprisegis/10.0/logging_mechanism

Babamir SM, Jalili S (2006) A logical based approach to detection of intrusions against programs. In: Proceedings of the 2nd conference on global E-security, (ICGeS-06). London, pp 72–79

Bandara AK, Lupu EC, Russo A (2003) Using event calculus to formalise policy specification and analysis. Policies for distributed systems and networks. In: Proceedings POLICY 2003. IEEE 4th international workshop on 4–6 June 2003, pp 26–39

Broda K, Clark K, Miller R, Russo A (2009) SAGE: a logical agent-based environment monitoring and control system. In: Proceedings of Aml'09, Zalzburg, Austria, 18–21 Nov, pp 112–117
Clarke EM, Grumberg OS, Lu, JHA Y, Veith H (2001) Progress on the state explosion problem in model checking (vol 2000). Lecture notes in computer science, Springer, Heidelberg
Das P, Niyogi R (2011) A temporal logic based approach to multi-agent intrusion detection and prevention. Int J Commun Network Secure 1(1)
Dixon C, Gago M-C F, Fisher M, Van der Hoek W (2004) Using temporal logics of knowledge in the formal verification of security protocols. In: Proceedings of the 11th international symposium on temporal representation and reasoning (TIME'04). IEEE 1530-1311/04, pp 148–151
Endriss U, Mancarella P, Sadri F, Terreni G, Toni F (2004) The CIFF proof procedure: definition and soundness results. Technical report 2004/2, Department of Computing, Imperial College London
Enterprise GIS Security (2013). http://resources.arcgis.com/content/enterprisegis/10.0/security
Evans D, Eyers DM, Bacon J (2010) Linking policies to the spatial environment. In: Proceedings of policies for distributed systems and networks (POLICY), pp 73–76
Farrell ADH, Sergot MJ, Salle M, Bartolini C (2004) Performance monitoring of service-level agreements for utility computing using the event calculus. In: Proceeding of workshop on contract languages and architectures (CoALa2004). 8th international IEEE enterprise distributed object computing conference, Monterey, pp 17–24
Gaaloul Kh, Proper HA, Zahoor E, Charoy F, Godart C (2011) A logical framework for reasoning about delegation policies in workflow management. Int J Inf Comput Secur 4(4):365–388
Kotenko I, Tishkov A, Chervatuk O, Sidelnikova E (2007) Security policy verification tool for geographical information systems. Lecture notes in geoinformation and cartography, pp 128–146
Kowalski R, Sergot M (1986) A logic-based calculus of events. New Gener Comput 4(1):67–95
Manna Z, Pnueli A (1995) Temporal verification of reactive systems: safety. Springer, New York
Miller DR, Harris SH, Harper AA, VanDyke S, Black CH (2011) Security information and event management implementation. McGraw–Hill Companies, New York
Montali M, Maggi FM, Chesani F, Mello P, Van der Aalst WMP (2011) Monitoring business constraints with the event calculus, 97. DEIS Technical report no. DEIS-LIA-002-11, LIA Series
Nowicka E, Zawada M (2006) Modeling temporal properties of multi-event attack signatures in interval temporal logic. In: Proceedings of the IEEE/IST workshop on monitoring, attack detection and mitigation (MonAM 2006), Tuebingen, Germany, Sept, pp 89–93
On-The-Fly, Ltl Model Checking with SPIN (2013). http://spinroot.com/spin/whatispin.html
Pnueli A (1977) The temporal logic of programs. In: Proceedings 18th IEEE symposium on foundations of computer science, Washington, DC, USA, pp 46–57
Rouached M, Claude G (2006) Securing web service compositions: formalizing authorization policies using event calculus, vol 4294. Lecture notes in computer science. Springer, Heidelberg, pp 440–446
Sattler U Description logic reasoners (2013). http://www.cs.man.ac.uk/~sattler/reasoners.html
Simko G, Sztipanovits J (2012) Active monitoring using real-time metric linear temporal logic specifications. In: Proceedings of the international conference on health informatics. Vilamoura, Algarve, 1–4 Feb, pp 370–373
Spanoudakis G, Christos K, Androutsopoulos K (2007) Towards security monitoring patterns. In: Proceedings of SAC'07, Seoul, Korea, 11–15 March, pp 1518–1525
The CIFF Proof Procedure for Abductive Logic Programming (2006). http://staff.science.uva.nl/~ulle/ciff/
Tishkov A, Kotenko I, Sidelnikova E (2005) Security checker architecture for policy-based security management, vol 3685. Lecture notes in computer science. Springer, Heidelberg, pp 460–465

Tsang E, Olsen R, Masry SH (2013) The event calculus on high-frequency finance. http://www.bracil.net/finance/papers/TsangOlsenMasri-EventCalculus-CCFEA2010.pdf

Winwood S, Klein G, Chakravarty MMT (2006) On the automated synthesis of proof-carrying temporal reference monitors. Lecture notes in computer science, Springer, Berlin, pp 111–126

Dynamical Attack Simulation for Security Information and Event Management

Igor Kotenko, Andrey Shorov, Andrey Chechulin, and Evgenia Novikova

Abstract The chapter considers a simulation-based approach to analysis of network resilience to botnet attacks in security information and event management (SIEM) systems, which can be applied to distributed geographic information systems (GISs). On the other hand, SIEM systems can use GIS technology for network awareness, taking into account the geographical location of hosts and network segments. To be able to protect the network against botnet attacks, it is necessary to investigate the processes occurring on all stages of the botnet life-cycle (propagation, control, and attack). The suggested approach can detect the critical nodes in the network, as well as determine and evaluate the protection mechanisms against botnet attacks. We propose the architecture of the dynamic attack simulation component (DASC) and describe its interaction with other SIEM components. The component prototype is presented and results of the implemented experiments are discussed.

Keywords Network security analysis · Infrastructure attacks · Dynamic simulation · Botnets · Security information and event management

I. Kotenko (✉) · A. Shorov · A. Chechulin · E. Novikova
Saint-Petersburg Institute for Informatics and Automation of Russian Academy of Sciences (SPIIRAS), 39, 14th Liniya, Saint-Petersburg, Russia
e-mail: ivkote@comsec.spb.ru

A. Shorov
e-mail: ashorov@comsec.spb.ru

A. Chechulin
e-mail: chechulin@comsec.spb.ru

E. Novikova
e-mail: novikova@comsec.spb.ru

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_14,

1 Introduction

Security information and event management (SIEM) systems are very useful when monitoring information system state and identifying security events that are both obvious (such as DDoS—distributed denial of service) and elusive (as a violation of policy or unauthorized access attempts).

One of the important advantages of modern SIEM systems is an ability to plug in different third-party security components, thus extending the functionality of the system. The most of the existing SIEM components function in real-time to provide information about attacks being implemented online. But the countermeasures taken against DDoS attacks or network worms cannot be considered effective because the time slot between attack start and its goal achievement is very small. Considering the high complexity of the infrastructure attacks and user latency, the detection of infrastructure attacks in real-time does not allow the system administrator to take efficient countermeasures.

Thus, it is necessary to detect the most vulnerable hosts, forecast the possible attack scenarios, figure out the possible sequences of attacks, preemptively identify the security objectives that are most likely to be targeted by the attacker, and identify the appropriate sets of countermeasures. Attack simulation is one of the possible solutions for this task that can be incorporated in SIEM systems.

GIS technology provides a good tool for the visual interpretation of the results of the attack simulation. It can present the geographical location of the hosts, thus enabling analysis of the attack development in space. Apart from this, geographical representation of the network topology allows users to maintain awareness in areas of their organizational responsibility.

Currently, the authors are developing a component of the SIEM system that implements analytical attack modeling and dynamic network attack simulation—called the attack modeling and security evaluation component (AMSEC). This component detects and corrects the errors in network configuration, revealing the possible assault actions for different security threats, determining the critical network resources, and choosing the security policy and security tools/mechanisms appropriate to current threats.

Analytical attack modeling analyzes the system by building high-level analytical models of attacks and countermeasures. It is described in Kotenko et al. (2012) in detail. Use of this approach allows us to expose many security problems at early stages, even at the stage of system design. But the analytical approach has some disadvantages—it cannot model low-level network communication (e.g., packets at transport and network layers) and, as a consequence, it cannot model several types of attacks (e.g., resource exhaustive attacks). Because of that, a new approach for attack modeling is suggested—dynamical attack simulation.

As opposed to analytical attack modeling, dynamical attack simulation allows us to simulate any kind of attack (limited only by resources and the precision of the model). It also allows us to estimate the network hardware overload when installing countermeasures as their usage can cause changes in the network flow routes and reveal their side effects.

Dynamical attack simulation can be used in order to reveal a network's weaknesses, providing a secure and relatively cheap (for both resources and time) dynamic model to detect various problems in the investigated system, optimize its performance, and implement this system in the real world.

The approach novelty consists in application of the dynamical attack simulation as the key element of the common approach for the attack modeling and risk assessment in SIEM systems. It allows calculation of a number of metrics, which in turn allow assessment of the possible countermeasures that can be used as initial parameters of the decision support and response component of SIEM systems.

The rest of the chapter is organized as follows. Section 2 discusses the state-of-the-art in dynamical attack simulation. In Sect. 3, we describe the architecture of the dynamic attack simulator and its implementation, as well as discuss its interaction issues with AMSEC. Section 4 presents the results of experiments. The chaption concludes with Sect. 5, which analyzes the chapter results and defines future work.

2 State-of-the-Art in Dynamical Attack Simulation

The task of network attack simulation imposes certain requirements on the choice of simulation tools and methods. The requirement for high scalability does not allow us to simulate all physical processes when analyzing infrastructure attacks. The complexity of dynamic behavior of the computer network model and protection mechanisms make the task of analytical modeling of these processes difficult to fulfill. Therefore, dynamical simulation is seen to be an efficient way to create complex dynamic models with a high level of adequacy and scalability.

Many studies have been devoted to dynamical simulation of security events in computer networks. Research in this area mainly focuses on methods of discrete-event simulation of processes being executed in the network structures (Owezarski and Larrieu 2004), as well as on trace-driven models initiated by trace data taken from actual networks (Simmonds et al. 2000). For worm propagation experiments, Wagner et al. (2003) used a simulation system of their own development. Riley et al. (2004) implemented the GTNetS simulation environment to build a network worm propagation model. Suvatne (2010) suggested a model of "Slammer" worm propagation by using the "Wormulator" simulation environment (Krishnaswamy 2009). Schuchard et al. (2010) presented a simulation environment that allows simulating a large-scale botnet containing 250,000 nodes as well as protection mechanisms targeted against botnet distribution. Gamer and Mayer (2009) considered a DDoS simulation tool called Distack which is based on the OMNeT++ discrete simulation system. Li et al. (2002) used their own simulation environment and testbeds to estimate efficiency, scalability, and the cost of implementation for the protection mechanism SAVE.

In this chapter, we propose use of the dynamical attack simulation approach based on discrete events, and network protocols are presented at a layer of packet transmission. We show how it could be applied for analysis of the computer network protection in the context of various network attacks.

Firstly, the dynamical attack simulation can be used to evaluate the resilience of the analyzed computer network against infrastructure attacks by assessing the time of node infection in the case of network worm distribution attack, as well as the amount of harmful traffic incoming to the attacked node in case of DDoS attacks. It can be helpful when detecting the routes of malicious traffic and revealing the weaknesses of the network topology and packet routing.

Second, the usage of the dynamical simulation supposes the analysis of computer network security when different protection mechanisms are chosen. The efficiency of the selected protection mechanism is determined, for example, by estimation of the following parameters: the amount of false-positive and false-negative errors, the reduction of the infection speed for vulnerable nodes, and the amount of malicious traffic coming to an attacked node. The application of protection mechanism can cause indirect costs, such as network overloading, because security systems can generate a lot of false-positive errors, thus making the work of legitimate users significantly complicated. The dynamical attack simulation can reveal side effects of the protection mechanisms.

3 Dynamic Attack Simulator in AMSEC

The architecture of the analytical attack modeling component and its functionality in context of the AMSEC is presented in Kotenko et al. (2012) in detail. In this section, we discuss the architecture of the dynamical attack simulation component (DASC) and its interaction mechanisms with the AMSEC.

The main functional AMSEC components and information flows between them are shown in Fig. 1. All main components needed for functioning of the DASC are explained below.

The architecture of the DASC consists of three main components (Fig. 2): (1) the basic simulation framework; (2) the internet simulation framework; (3) the subject domain library; and (4) the interaction interfaces with the AMSEC components. Interaction is done via the graphs database, the network specification database, and the supporting database.

After network attack simulation in accordance with assigned parameters, the DASC passes the obtained protection estimations for the modeled computer network to the graphs database (DB). Along with the estimations obtained from the attack graph generator component, these estimations will be processed by the security evaluator component (see Fig. 1).

The *basic simulation framework* consists of modules that exchange messages with each other. Modules that directly generate events are called simple modules, which can be united into compound modules with some hierarchy. The general

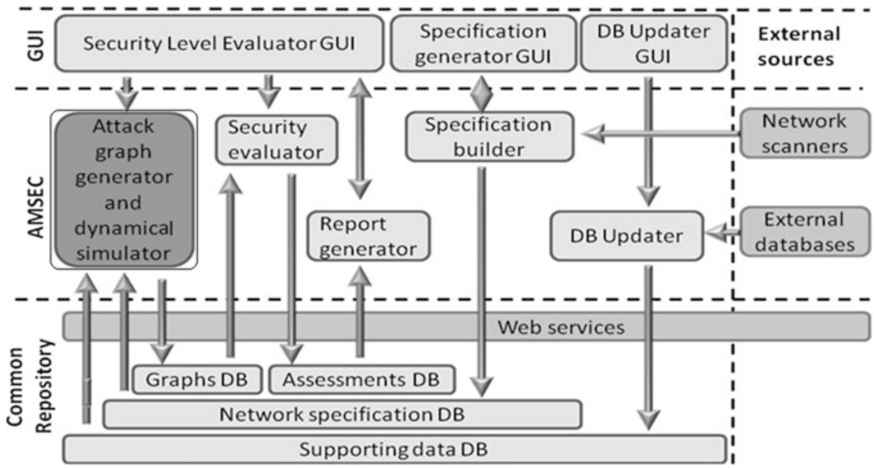


Fig. 1 Attack modeling and security evaluation component functional components and information flows. DB, database; GUI, graphical user interface

model of the network is a compound module. Modules exchange messages via connections.

The *internet simulation framework* is used for the Internet nodes and protocol simulation. It contains modules that generate real network topologies as well as network application models whose behavior is close to the behavior of the corresponding real network applications and the protocol models of transport, network, and channel layers. This component represents the computer network model as a network with nodes, including TCP/IP stack.

The *subject domain library* facilitates simulation of the subject domain processes and includes the components of attack and protection mechanisms and the modules extending IP node functionality, such as the filtering table, the packet analyzer, the legitimate user models, etc. The set of components of the subject domain is divided into two groups—attack network components and protection network components. These components are implemented in the form of simulation models.

The model of the attack network specifies a set of processes generated by the attack network. It is implemented as a bundle of three relatively independent models—a model of attack distribution, a model of attack network management, and a model of attack execution phase. For each initial process of the attack network, the protection network implements a reverse process targeted to counteract the attack implementation.

The protection network model consists of the models of counteraction against the attack network distribution, the attack management, and the attack implementation. Note that each countermeasure action contains self-organization processes.

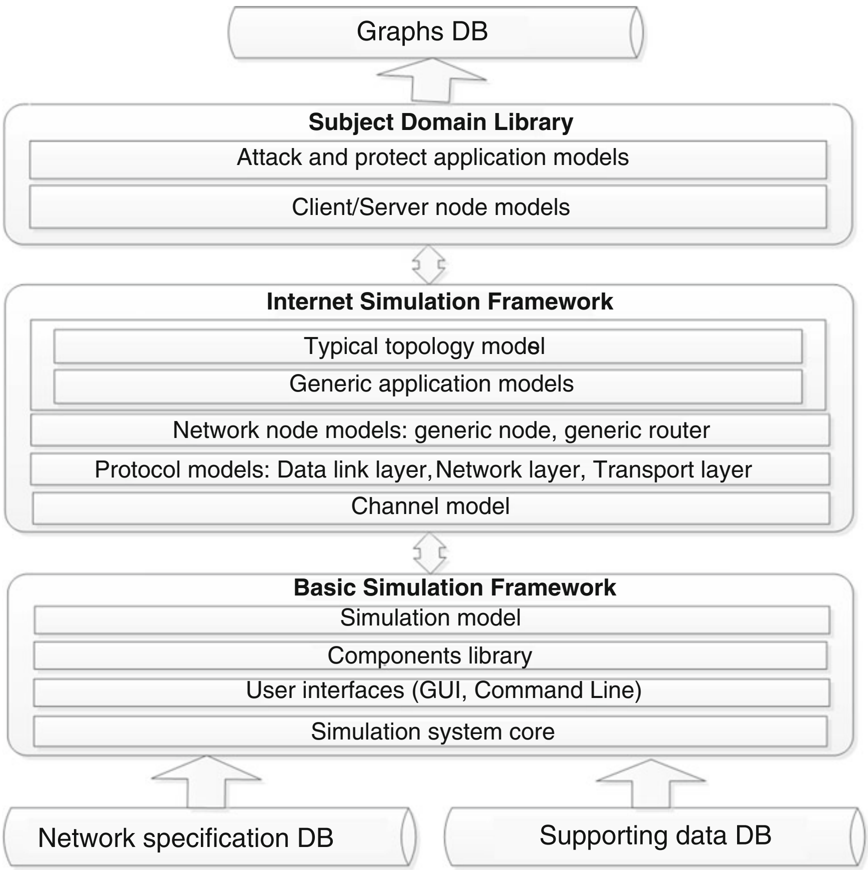


Fig. 2 Dynamical attack simulation component architecture. DB, database; GUI, graphical user interface

All models are implemented as parameterized modules, which can be installed on the nodes of the simulated network at different layers (network layer, transport layer, and application layer). To achieve this, a special interface is implemented to provide the interaction between the typical nodes and the modules of attack and protection.

The DASC has two types of input data: (1) the network specification obtained from the network specification database and (2) the supporting data (experiment time; vulnerable nodes; botnet settings; protection mechanisms settings, etc.) which is stored in the supporting data DB. The structure of the input data for the DASC is presented in Fig. 3.

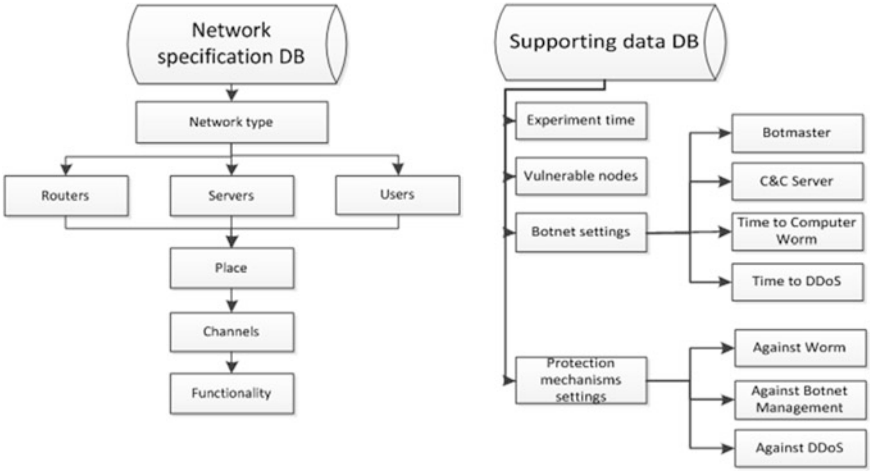


Fig. 3 Dynamical attack simulation component input. DB, database; DDoS, distributed denial of service

The output of the DASC is as follows:

- The time required for infection of vulnerable nodes when implementing the network worm attack in the conditions of realizing the active and inactive protection mechanisms;
- The quantity of the nodes infected for unit of time or the total amount of infected nodes in comparison to all vulnerable ones;
- The quantity of malicious traffic coming directly to the attacked node when executing DDoS attacks with active and inactive protection mechanisms;
- The volume of network traffic passing through the routers, divided into malicious and legitimate;
- The amount of FP packets per protection mechanism in order to determine a load created by the protection mechanisms on the computer network.

The output is stored in the common repository as a set of records of the specified format needed for further processing by the security evaluator of the AMSEC (see Fig. 1). Thus, the data obtained at this modeling stage can be used to optimize the system security and evaluate the cost of different protection mechanisms.

The *basic simulation framework* is implemented using the discrete event simulation environment OMNeT++ (Varga 2010). OMNeT++ provides the tools for simulation of network structures and message propagation in these structures. The library INET Framework[1] is used for simulation of packet-switching networks.

[1] The INET Framework is an open-source communication network simulation package for the OMNeT++ simulation environment. http://inet.omnetpp.org/

This library provides the components implemented as the OMNeT++ modules and contains a large variety of models of network devices and network protocols for wired and wireless networks.

Simulation of realistic computer networks is carried out using the ReaSE library[2]. This library is an extension of INET Framework (see footnote 1). It provides tools for creating realistic network topologies whose parameters are statistically identical to the parameters of real computer network topologies. ReaSE includes also a realistic model of network traffic, modeled at the packet level (Li et al. 2002). The models of network traffic are based on the approach described in Vishwanath and Vahdat (2006). This approach allows generating the packet-level traffic with parameters, which are statistically equivalent to the traffic observed in real computer networks.

Direct modeling of the subject area is conducted through a set of components implemented by us. These components are included into the BOTNET Foundation Classes and contain the models of network applications related to botnets' life-cycle. Being the C++ classes, the components of each level are united into the component libraries, thereby facilitating a modularity and code reuse properties.

4 Experiment Preparation and Implementation

For computer network simulation, it is necessary to conduct the following stages: (1) the preparatory stage, (2) the parameter setup, (3) the implementation (simulation), (4) and finally the output parameters analysis.

4.1 Preparatory Stage

The preparatory stage serves to define network parameters, such as quantity of nodes, their types, and links between them. To create a model of a computer network, one can use the generator of realistic topologies ReaSE or download it from the Network Specification Database; otherwise, the network is created manually using the graphical editor built into the DASC or any other text editor. It is necessary to define the parameters of infrastructure attacks such as its type, sources, etc. When testing the network protection level, the protection mechanisms need to be selected. These mechanisms are programmed as simple modules, which are connected to the network nodes via specialized interfaces.

The network topology and configuration are modeled at two levels. At the first level, the network topology is modeled as a set of autonomous systems (AS). To

[2] ReaSE Realistic Simulation Environments for OMNeT++. https://i72projekte.tm.uka.de/trac/ReaSE

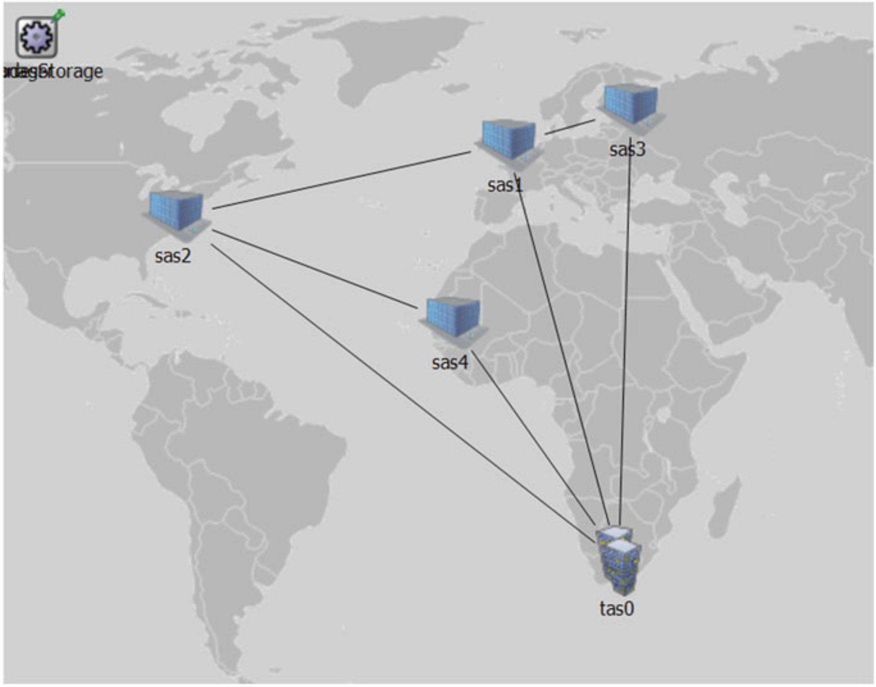


Fig. 4 Top-level network topology of the organization

generate the topology of a computing network at the AS level in our work, we applied the positive-feedback preference method (Li et al. 2004). At the second level, an internal topology of each AS is modeled (Router-level topology). We used the heuristically optimal topology model (Li et al. 2002) as the internal topology model.

Let us consider the following initial parameters for the simulation environment, network topology, and network configuration. An organization has a head office (tas0) and four branches (sas1, sas2, sas3, and sas4) (Fig. 4). The office and branches are connected to each other via the computer network, which consists of 934 hosts.

The network contains 8 servers (5 web servers, 2 email servers, and 1 DNS server). Approximately 30% of all client hosts have vulnerabilities. We defined the master node and the command center node in the analyzed network. The first one serves as the source of initial worm distribution and the initiator of commands for controlling the botnet, whereas the latter translates the commands of the botnet master to the infected nodes.

All nodes in the subnets are united via edge routers. In each subnet, a root gateway and a core that unite the subnets with each other are determined.

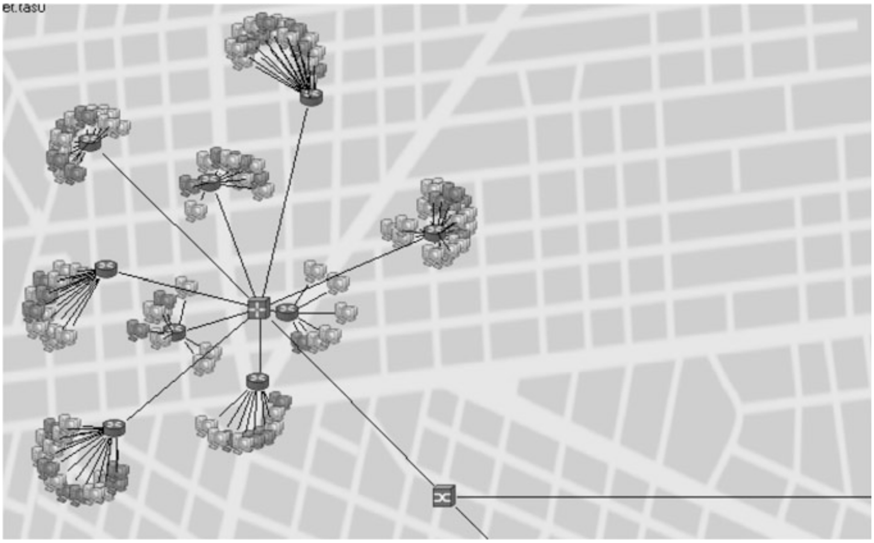


Fig. 5 Visual presentation of the subnet tas0

Figure 5 shows the subnet tas0 at the second presentation level. The subnet consists of 169 nodes.

Subnets sas1, sas2, sas3, and sas4 of the branches have a structure similar to subnet tas0. We use typical office personal computers as client nodes; they send queries to servers, thus creating legitimate traffic.

The model of standard protocol stack, including the protocols of channel, network, transport, and application layers (PPP, LCP, IP, TCP, ICMP, ARP, UDP, etc.) is installed on each node. Depending on the functional role of a node, the additional models of network components implementing the corresponding functionality can be installed.

4.2 Parameter Setup

At this stage, the main parameters for simulation network functioning, infrastructure attack implementation, and the protection mechanism operation are defined.

The *network parameters* are the set of variables describing the amount of active modes, the time of their activation, the type, and the behavior of a legitimate node (quantity of requests to the server, intervals between requests, packet size, and number of sessions), the protocols' configurations, etc. Legitimate user behavior is specified using application profiles. These profiles can be edited using the ReaSE utility. They have the following parameters:

- Profile number, which identifies a bundle of parameters for the work with certain types of server nodes;
- Probability of a start of the node operation according to the given profile;
- Length of requests in bytes;
- Amount of requests during a session;
- Length of a response from the server in bytes;
- Number of responses from the servers per one request;
- Time interval between requests;
- Time latency before a response to a request;
- Time interval between sessions.

In experiments, legitimate nodes were activated randomly during 200 s.

Infrastructure attack parameters are as follows: the number of vulnerable nodes or nodes that implement the attack, the IP addresses of attack sources and their ports, the starting time of the attack, the attack target (its address and port), the intensity (it can be constant or variable), and the address substitution mode of the attack. Nodes having vulnerabilities or being sources of the attack can be specified manually or with the use of a random sample from legitimate nodes. It is necessary to determine the protocols used for attack implementation.

When conducting experiments, the following parameters values were used: the vulnerable nodes and nodes implementing the attack were chosen randomly among 281 legitimate nodes. The mechanism of worm distribution had the following parameters: the starting time, which was the 100th second of the simulated time; the scanning method, which was random; the frequency of packet sending, which was 6 packets per second; the port of the vulnerable service, which was 8,081; the address and port of the command center to which the compromised computer connects, which was host_cc_srv:8,080. The DDoS attack implementation had the following parameters: the starting time of the attack, which was the 400th second of the simulated time; the attack type, which was SYN flooding; the sending packet frequency, which was 10 packets per second; the port for sending malicious packets, which was 8,080; and the address and port of the computer under the attack, which was tas0.WebServer1:80.

The user needs to specify the following *protection mechanisms parameters*: the number of the nodes with installed protection mechanisms, the type of installed protection mechanisms, the interaction mode, and parameters specific to each protection mechanism type. Nodes with the installed protection mechanism can be chosen manually or randomly. It is also possible to install several different protection mechanisms on a single node.

To protect a computer network from the worm distribution, the protection mechanism based on the failed connection (FC) approach (Chen and Tang 2004) was used. The protection mechanism had the following parameters: the buffer for sender IP address, which was 400, and the threshold value of the amount of RST packets for each IP address of the sender, which was 3. It was allowed to decrease the amount of received RST packets relatively to threshold every 5 s; if the IP address was blocked, it could be deleted from the list of blocked IP addresses. For protection against DDoS attacks, we applied the protection mechanism SIM (Peng et al. 2004). The threshold value of the number of new clients per second was set to 100.

When conducting simulation, the following *general experiment parameters* should be set: the duration of simulation, the number of experiments, the experiment visualization mode (graphical and command), the input and output data parameters, etc. The duration of experiments was set to 600 s of the simulated time. The experiments were conducted without a graphical presentation to increase their performance.

4.3 Simulation Stage

We simulated the infrastructure attacks of two types—the network worm distribution and the DDoS attack.

At the 100th second of the simulated time, a master node starts the network scanning in order to discover vulnerable nodes and connects to an IRC channel that is used to send commands to the command center for their further transmission to zombie nodes. After infection, the compromised node becomes itself a source of worm distribution. It makes 30 attempts to establish connection to the IRC channel for further commands.

At the 400th second of the experiment fulfillment, the "master" sends a command to start the DDoS attack on the web server (tas0.WebServer1). The master sends a message, indicating the attack target via the IRC channel to the command center, which in its turn uses the IRC channel to distribute the message to compromised nodes. After receiving a message, a compromised node extracts information about the attack target from the message and immediately joins to the DDoS attack.

The results of this stage are the set of the records of the predefined format containing the data on simulation execution. These records are used to analyze obtained information at the next stage.

4.4 Output Data Analysis

In this subsection, we present results on the simulation of the infrastructure attacks with parameters described in previous subsections. They are given mainly in graphical form.

In the case of network worm distribution without protection mechanisms, the infection of all vulnerable computers is completed in 8 s. When using the protection mechanisms, the infected computers are blocked and the worm distribution is terminated. Figure 6 illustrates the number of infected vulnerable computer in the network.

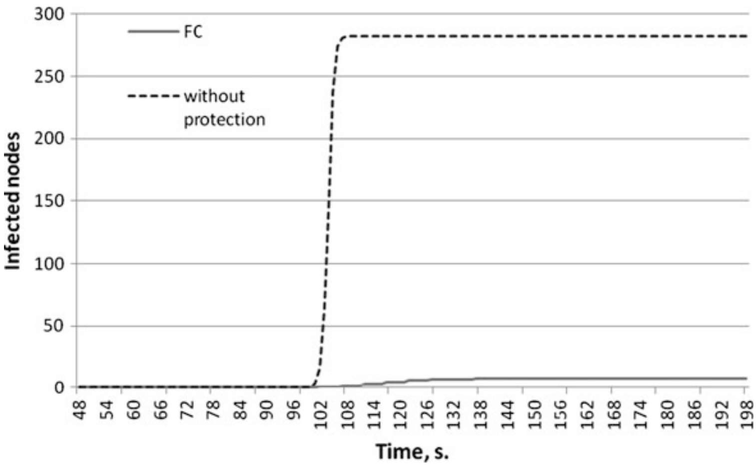


Fig. 6 Speed of network worm distribution. FC, failed connection

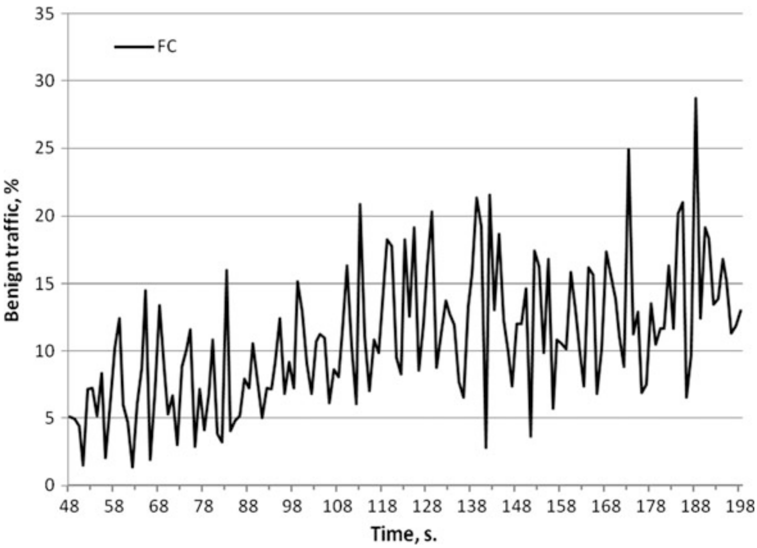


Fig. 7 Quantity of filtered legitimate traffic. FC, failed connection

However, the protection mechanism creates a certain load on the network, as it also blocks a legitimate traffic. Figure 7 shows that protection mechanism FC can filter from 5 to 28% of a legitimate traffic when the amount of malicious traffic in the network increases.

Figure 8 shows dependencies between the quantities of false-positive and false-negative errors, and correct detections in the experiment. These data are obtained

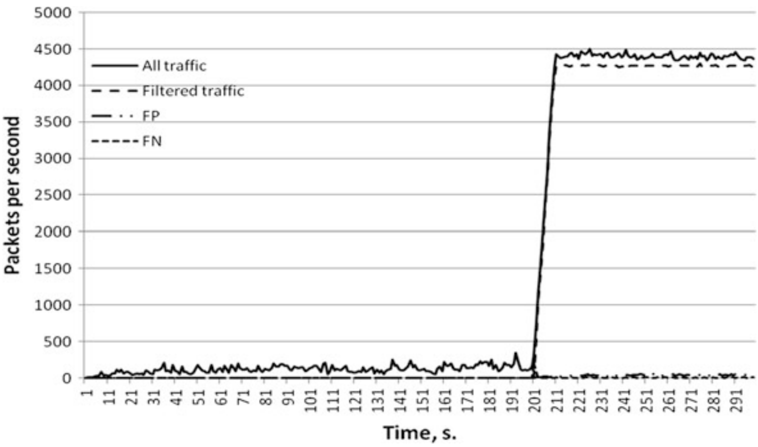


Fig. 8 Dependencies of traffic volume on modeled time

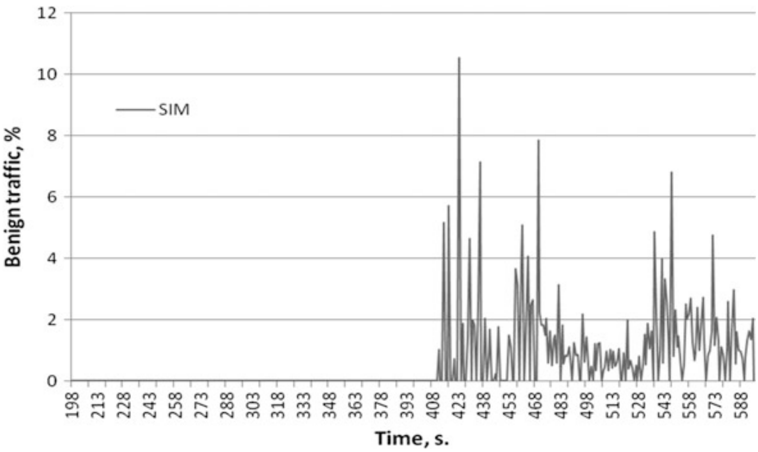


Fig. 9 Amount of filtered legitimate traffic

when processing network packets by the protection mechanism SIM. Packets going to the web server are not filtered by other protection mechanisms.

Figure 9 presents the amount of the filtered legitimate traffic (as a percentage of the total legitimate traffic) passing through the protection mechanism SIM when setting protection mechanisms on 100% of the routers. The protection mechanism SIM demonstrated a high level of TP, whereas the level of FN was low. Only at the beginning of the attack was a small growth of the FN level packets observed.

However, as the protection mechanism starts blocking packets with unknown addresses when the DDoS attack begins, the FP level gradually grows; meanwhile, the percentage of filtered legitimate packets can raise up to 10–15% of total legitimate traffic (Fig. 9).

5 Conclusions

In this chapter, we described an approach based on dynamical attack simulation for network security evaluation, which can be applied in SIEM systems. We have shown that this approach has the following features: simulation of attacks that cannot be analyzed by the analytical attack modeling component (e.g., resource-exhaustive attacks [DDoS], etc.); simulation of host low-level interaction (at network, transport, and channel layers); detection of network weak places (e.g., hosts likely to be used in DDoS attacks); assessment of the protection mechanisms' efficiency and their side effects; sharing the network models with the analytical part of AMSEC; and generation of the extra metrics that allow obtaining more precise risk analysis results.

The main contribution of this chapter consists in the development of an integrated simulation environment for network security evaluation, including libraries with implemented models of hosts and malefactors. The chapter describes the key issues of the suggested approach and presents the generalized architecture of the integrated simulation environment in the context of the attack modeling and security evaluation component of the SIEM system. The results of the conducted experiments on simulation of botnets, DDoS attacks, and protection mechanisms against them are shown in the chapter. They can be used when assessing the protection mechanism efficiency and making decision on protection mechanism selection.

The future steps of the research will be devoted to the detailed elaboration of the interaction interfaces with AMSEC components, the development of techniques that can cope with large networks, and the definition of more sophisticated security metrics based on the results of dynamical attack simulation.

Acknowledgments This research is being supported by a grant from the Russian Foundation of Basic Research, Program of fundamental research of the Department for Nanotechnologies and Informational Technologies of the Russian Academy of Sciences (contract #2.2), State contract #11.519.11.4008, and partly funded by the EU as part of the SecFutur and MASSIF projects.

References

Chen S, Tang Y (2004) Slowing down internet worms. Proceedings of the 24th international conference on distributed computing systems (2004)

Gamer T, Mayer C (2009) Large-scale evaluation of distributed attack detection. Workshop on OMNeT++. Rome, Italy, pp 1–8

Kotenko I, Chechulin A, Novikova E (2012) Attack modeling and security evaluation for security information and event management. SECRYPT 2012:391–394

Krishnaswamy J (2009) Wormulator: simulator for rapidly spreading Malware. Master's Projects, San Jose

Li J, Mirkovic J, Wang M et al. (2002) Save: source address validity enforcement protocol. Proceedings IEEE INFOCOM, NY, pp 1557–1566

Li L, Alderson D, Willinger W et al. (2004) A first-principles approach to understanding the internet's router-level topology. ACM SIGCOMM computer communication review, pp 3–14

Owezarski P, Larrieu N (2004) A trace based method for realistic simulation. Communications, IEEE International Conferences Toulouse, France, pp 2236–2239

Peng T, Leckie C, Ramamohanarao K (2004) Proactively detecting distributed denial of service attacks using source IP address monitoring. Lect Notes Comput Sci 3042:771–782

Riley G, Sharif M, Lee W (2004) Simulating internet worms. Proceedings 12th international workshop on modeling, analysis, and simulation of computer and telecommunication systems (MASCOTS). Atlanta, pp 268–274

Schuchard M, Mohaisen A, Kune D et al. (2010) Losing control of the internet: using the data plane to attack the control plane. Proceedings 17th ACM conference on computer and communications security, CCS '10. ACM, USA, pp 726–728

Simmonds R, Bradford R, Unger B (2000) Applying parallel discrete event simulation to network emulation. PADS '00. Proceedings of the fourteenth workshop on parallel and distributed simulation. Washington, pp 15–22

Suvatne A (2010) Improved worm simulator and simulations. Master's Projects, San Jose

Varga A (2010) OMNeT++. Modeling and tools for network simulation. Wehrle K, Günes M, Gross J (eds) Springer, Berlin (2010)

Vishwanath KV, Vahdat A (2006) Realistic and responsive network traffic generation. Proceedings of the conference on applications, technologies, architectures, and protocols for computer communications

Wagner A, Dubendorfer T, Plattner B et al. (2003) Experiences with worm propagation simulations. Proceedings of the ACM workshop on rapid Malcode. NY, pp 34–41

Part VI
Urban GIS

A Simulation Model of Urban Growth Driven by the Bosphorus Bridges

Ismail Ercument Ayazli, Fatmagul Kilic, and Hulya Demir

Abstract Istanbul, which joins Asia and Europe, has always attracted attention thanks to its cultural, natural, and environmental heritages. However, an increase in population has caused an enormous transportation problem. To overcome this problem, two bridges were built on the Bosphorus strait, and a third bridge will be built on the north side of the Bosphorus. Shortly after the first two bridges were built, each bridge created its own traffic and triggered urbanization northward into Istanbul. The main purposes of this chapter are to determine land use changes driven by the Bosphorus bridges, as well as the probable impact of a third bridge on land usage. For these purposes, an urban growth simulation model was created for the year 2030, using a SLEUTH-based urban growth model. According to the results, Istanbul will continue growing northward. In the north of the city, 40% of forest areas and 83% of agriculture-urban open space will transform into settlement areas.

Keywords Urban GIS · Urban Growth · Cellular Automata · Simulation · Transportation

I. E. Ayazli (✉)
Department of Geomatic Engineering, Cumhuriyet University, Sivas 58140, Turkey
e-mail: ayazli@outlook.com

F. Kilic, · H. Demir
Yildiz Technical University, Department of Geomatic Engineering, Esenler 34220 Istanbul, Turkey
e-mail: fkilic@yildiz.edu.tr

H. Demir
e-mail: hudemir@yildiz.edu.tr

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_15,

1 Introduction

According to the European Environmental Agency (EEA), approximately 75% of European people live in urban areas. By 2020, this rate will reach 80% (EEA 2006). Increasing population causes land use changes around the cities. This land usually transitions from rural areas to urban areas. In metropolitan or rapidly growing cities, controlled urban growth is desired in order to increase quality of life and to make productive use of urban and environmental resources. Urban decision makers want to know the direction and speed of urban growth. Therefore, several urban growth models have been developed since the beginning of the twenty-first century (Baslik 2008; Candau 2002). Such models include the Monocentric City Model, the Tiebout Local Public Finance Model (Nechyba and Walsh 2004), the Concentric Zone Theory, the Sector Theory, and the Multiple Nuclei Theory (Yu and Ng 2007).

In the 1950s, the systematic approach was suggested for planning studies. Urban growth was accepted as a dynamic system, which was formed from a number of nonlinear, dynamic, and complex subsystems (Baslik 2008). Therefore, complex systems are used to model urban growth (Clarke et al. 1997; Cheng 2003). Recent developments in computer technology have enabled urban growth to be modeled with simulation techniques. Cellular automata (CA), which runs subdivided cells of a regular lattice, is one of the simulation methods (Clarke et al. 1997; Silva and Clarke 2002; Benenson and Torrens 2004; Batty 2007). The future state of each cell is determined by its adjacent cell's state. CA has five main elements: space, state, neighborhood, transition rules, and time. Each automaton is defined by a set of state, neighborhood of automaton, and transition rules. Various softwares based on a CA algorithm were created to simulate urban growth. SLEUTH is one of them, and it has been used in a number of projects (Silva and Clarke 2002; Jantz et al. 2003; Sevik 2006; Xibao et al. 2006; Watkiss 2008; Wu et al. 2009).

The name SLEUTH was derived from the first letters of the required input data: Slope, Land Use, Excluded, Urban, Transportation, and Hillshade. It was written in C programming language, works under UNIX, and has two components: an urban growth model (UGM) and a land cover deltatron model (LCD). The UGM uses the standard GNU C compiler (gcc), and the LCD is embedded in the code and driven by the UGM (URL 1).

SLEUTH is based on a CA algorithm and uses Monte Carlo iteration for the simulation, which is completed in three modes: test mode, calibration mode, and prediction mode. Data are tested in test mode to see whether or not they are ready for calibration. Calibration mode is completed in four steps: coarse, fine, final, and derive forecasting coefficients. SLEUTH uses brute force calibration methodology and determines the best-fit value for each coefficient at the end of the process (Candau 2002; Sevik 2006).

Table 1 Growth rules and coefficients (Ayazli et al. 2010)

Growth Rules	Growth Coefficients
Spontaneous Growth	Dispersion, Slope
New Spreading Center	Breed, Slope
Edge Growth	Spread, Slope
Road-Influenced Growth	Breed, Road Gravity, Slope, Spread

A simulation model is created by four growth rules and five growth coefficients (Table 1). Relations are shown between growth rules and growth coefficients. Each coefficient value must be between zero and 100 (URL 2; Sevik 2006).

Self-modification is associated with the second level of the growth rules. It "is prompted by an unusually high or low growth rate. The limits CRITICAL_HIGH and CRITICAL_LOW are defined in the scenario_file" (URL 3).

Because of SLEUTH's self-modification qualities, coefficient values that initialize the model for a date in the past (Start_Date) may be altered by the simulation end date (Stop_Date). "Therefore, for forecast run initialization, the coefficient values at the simulation end date are used to initialize a new simulation into a future date" (Candau 2002).

After computing, the best-fit values determined for each coefficient in forecasting mode/prediction mode start to create a simulation model.

This study aims to determine how the Bosphorus bridges, and their access roads, impact urbanization in Istanbul. Overcoming this problem may entail constructing a third bridge. Although zone plans that were made in the 1960s suggested an east-to-west urbanization pattern in Istanbul, the Bosphorus and Fatih Sultan Mehmet (FSM) bridges have catalyzed a northward urban growth. Most importantly, natural areas that are located along the north side of Istanbul will be under threat if the third bridge is constructed. The experiences of other Bosphorus bridges show that constructing bridges alone does not overcome transportation problems. On the contrary, each bridge creates its own traffic.

In this chapter, the third bridge's effect on urbanization was examined using satellite images. Moreover, this chapter has created a simulation model to predict urban growth for the year 2030 using CA. The following periods were selected: 1972, 1987, 2002, and 2009. Landsat imagery for Istanbul were classified during these years and created land use classes. Using digital elevation model (DEM) data for the year 2005, transportation networks for the year 1997 and 2009, and the third bridge route were obtained from Istanbul Metropolitan Municipality. The motorways that were labeled 1972 and 1987 were digitized manually from Landsat imagery. Change detection analysis was carried out in order to ascertain land transition from a natural area to an urban area between the years 2009 and 2030.

Input data were arranged and analyses were made using ArcGIS and Erdas. SLEUTH3.0_beta was used for creating a simulation model.

As the classification results suggest, forest area covered 712.35 km^2 in 2009, whereas they only covered 1,289.36 km^2 in 1972. Approximately half of forest areas and green lands have transformed into settlement areas in 27 years. If the third bridge is built on the Bosphorus strait—which is the most important of all natural areas and is located along the north side of Istanbul—the community will find itself under an urbanization threat.

2 Materials and Methodology

2.1 Study Area

Istanbul, which is located in northwestern Turkey, has been the attractive center of Turkey because of its natural, cultural, historical, social, and economic potential. At the beginning of the second half of the twentieth century, Istanbul's population began increasing rapidly. The population was 3 billion in 1973 and exceeded 13 billion in 2011 (URL 4). Its area is 5,461 km^2 (URL 5). Population growth has caused an enormous transportation problem. To overcome this, two bridges were constructed on the Bosphorus strait in 1973 and 1988. However, they were not an adequate solution for Istanbul's chaotic traffic problem. On the contrary, they created their own traffic and created new problems. The politicians now want to build a third Bosphorus bridge, which they believe will be a solution for Istanbul's transportation problem. The uppermost route (Fig. 1) was announced as the third bridge route in 2010 by the Ministry of Transport and Communications. At the beginning of this study, the third bridge route could not be obtained from the relevant institutions. Therefore, the study area had to be restricted, as shown in Fig. 1.

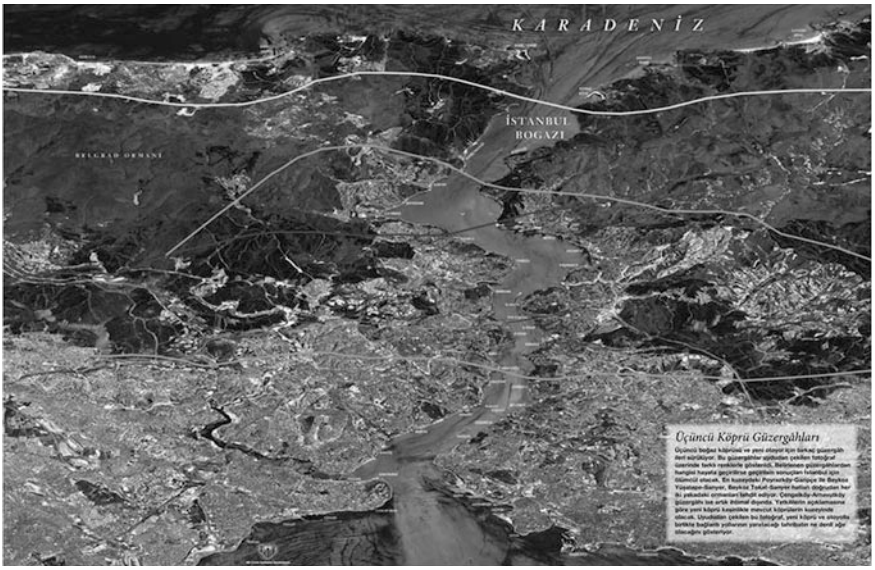


Fig. 1 Third bridge routes (Kabadayi 2008)

2.2 Data

Several data are used for monitoring urban growth, such as satellite imagery, land use/cover maps, DEM, digital terrain model (DTM), administrative boundary, topographic maps, aerial photos, ortophoto, geological maps, socioeconomic data, etc. (Cheng 2003; Celikoyan 2004). However, the main purpose of this study is only to ascertain the effects of the third bridge on urbanization in Istanbul. Thus, some factors should be eliminated, such as socioeconomic factors, inner city problems, housing preferences, etc. For this reason, four main data were used for this chapter, as shown in Table 2.

Land use maps were generated from satellite imagery for the years 1972, 1987, 2002, and 2009. The maximum likelihood algorithm was used to classify Landsat images for the year 1972. Other types of imagery were classified using the Iterative Self-Organizing Data Analysis Technique (ISODATA) algorithms. Classified satellite imagery and master plans were used to create urban and excluded areas. Water surfaces, military zones, and airports were selected as excluded areas. Forest areas were not included in the excluded area dataset because they can transform to settlement areas. DEM data obtained from Istanbul Metropolitan Municipality were utilized in order to create Hillshade and Slope maps. Landsat data for the years 1972 and 1987 were manually digitalized for producing road maps. The motorways were obtained from Istanbul Metropolitan Municipality for the years 1997 and 2009. The third bridge route is shown in the uppermost in Fig. 1 and was integrated with the motorways for the year 2009.

Input data, which are required for running the simulation software, must be grayscale GIF images that have the same projection, same map extent, same resolution, and the required naming format: “istanbul.urban.1972.gif” (URL 6).

Table 2 Input date and data source

Data	Years	Data type	Data class	Data source
Satellite imagery	1972	Raster	Urban, excluded	Landsat MSS
	1987	Raster	Urban, excluded	Landsat TM
	2002	Raster	Urban, excluded	Landsat ETM
	2009	Raster	Urban, excluded	Landsat TM
Digital elevation model	2005	Raster	Slope, hill shade	SPOT
Plans	1980	Vector	Excluded	Master plan
	1995	Vector	Excluded	Master plan
	2009	Vector	Excluded	Master plan
Route and motorways	1972	Vector	Transportation	Landsat MSS
	1987	Vector	Transportation	Landsat TM
	1997	Vector	Transportation	Istanbul metropolitan municipality
	2009	Vector	Transportation	Istanbul metropolitan municipality

2.3 Software and Hardware

All input data are arranged using ArcGIS and Erdas software in the Acer Aspire 5920G notebook, which has an Intel Core 2 Duo processor 2.2 GHz and 2 GB DDR2 RAM.

2.4 Creating a CA-Based Urban Growth Model

CA a simulation method that runs subdivided cells of a regular lattice. The future state of each cell is determined by its adjacent cell's state. CA has five main elements: space, state, neighborhood, transition rules, and time (Fig. 2). Each automaton is defined by a set of state, neighborhood of automaton, and transition rules. Information may be exchanged through neighborhoods during *t* time steps (Fig. 3; Benenson and Torrens 2004).

SLEUTH is based on a CA algorithm and uses Monte Carlo iteration for simulations, which are completed in three modes: test mode, calibration mode, and prediction mode. For each step, required parameters, such as iteration number, model start-stop date, input–output data paths, and coefficient value, are set into a scenario file, which is SLEUTH's execution file.

SLEUTH has 13 metrics that were used to evaluate the best fit value of growth coefficients for using the prediction mode. These are: Product, Compare, Pop, Edges, Clusters, Cluster Size, Lee-Sallee, Slope, Percent-Urban, X-Mean, Y-Mean, Rad, and F-Match. There is no consensus with regard to the metrics that should be used during the calibration mode. To compare, population and Lee-Sallee statistics were used in the Washington-Baltimore metropolitan area (Jantz et al. 2003); Dietzel and Clarke (2007) used the Optimal SLEUTH Metric (OSM) technique. Furthermore, Sevik (2006), Oguz et al. (2007), and Silva and Clarke (2002) only used the Lee-Sallee metric in their applications. The Lee-Sallee metric, which is "the ratio of the intersection and the union of the simulated and actual urban areas" (Dietzel and Clarke 2006), was used in this study.

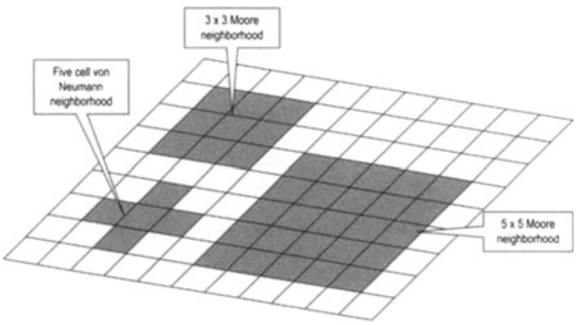


Fig. 2 Cellular automata neighborhoods (Benenson and Torrens 2004)

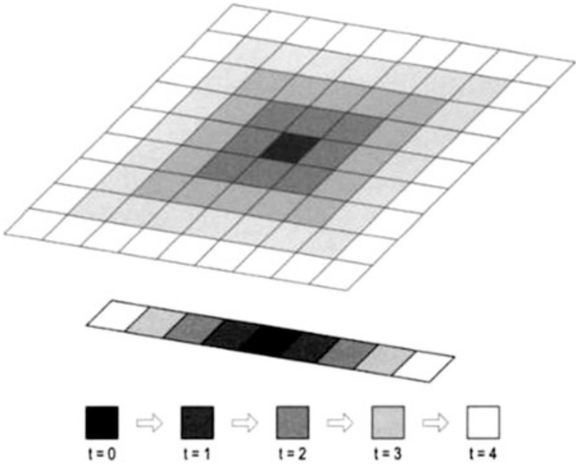


Fig. 3 Information exchange in cellular automata (Benenson and Torrens 2004)

The simulation model is created by using four growth rules: spontaneous growth, new spreading center growth, edge growth, and road-influenced growth. They are related to five growth coefficients: dispersion, breed, spread, road gravity, and slope (URL 2; Sevik 2006).

The *spontaneous growth rule*, which is controlled by dispersion and slope coefficients, defines the occurrence of random urbanization of land (Fig. 4) (URL 7).

The second growth rule, *the new spreading center growth rule*, is controlled by breed and slope coefficients. This rule can help determined whether any of the new, spontaneously urbanized cells will become new urban spreading centers (Fig. 5) (URL 8).

The *edge growth rule* is a function of spread and slope coefficients. It propagates both the new centers that are generated and the more established centers. If a

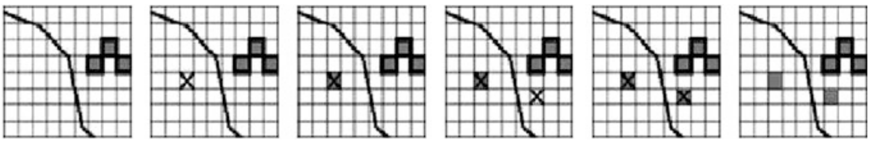


Fig. 4 Illustration of spontaneous urban growth rule (URL 7)

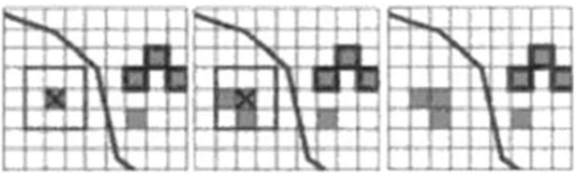


Fig. 5 Illustration of new spreading center growth rule (URL 8)

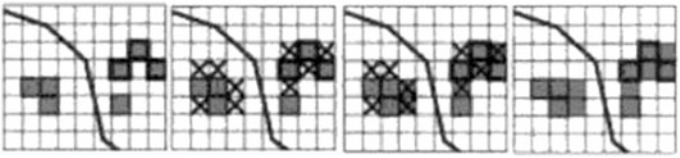


Fig. 6 Illustration of edge growth rule (URL 9)

nonurban cell has at least three urbanized neighboring cells, with an inclusive slope value that is less than 22%, it will be urban (Fig. 6) (URL 9).

The final growth rule is the *road influenced growth rule*, which is used to determine the effect of transportation networks on urbanization. This is done through three steps, which are controlled by all of the growth coefficients. First, newly urbanized cells are selected, and the existence of a road is sought out in particular neighborhoods. If a road is found within a given maximal radius of the selected cell, a temporary urban cell is placed at the point on the road that is closest to the selected cell. Second, this temporary urban cell conducts a random walk along the road if a cell that neighbors the temporary urbanized cell (on the road) is available for urbanization. If two adjacent cells to this newly urbanized cell are also available for urbanization, it will be classified as urban (URL 10) (Fig. 7).

3 Results

An urban growth simulation model was created for the year 2030 using SLEUTH. This was done in three steps. Data are tested in test mode to see whether they are ready for calibration. In calibration mode, which was carried out in four steps (coarse, fine, final, and derive forecasting coefficients), the model start date and model stop date were set to 1972 and 2009, respectively, in the scenario file for Istanbul. After the calibration mode, the best-fit values of the growth coefficients were computed, as shown in Table 3.

In the prediction phase, the best-fit value and prediction date range were arranged in the scenario file. The prediction phase initiated in 2009 and was terminated in 2030. After the prediction step, a simulation model was created for Istanbul (Fig. 8).

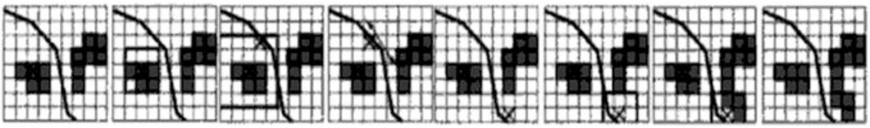


Fig. 7 Illustration of road-influenced growth rule (URL 10)

Table 3 Best-fit values of the growth coefficients

Growth coefficients	Best fit values
Dispersion	7
Breed	100
Spread	100
Slope	1
Road gravity	100

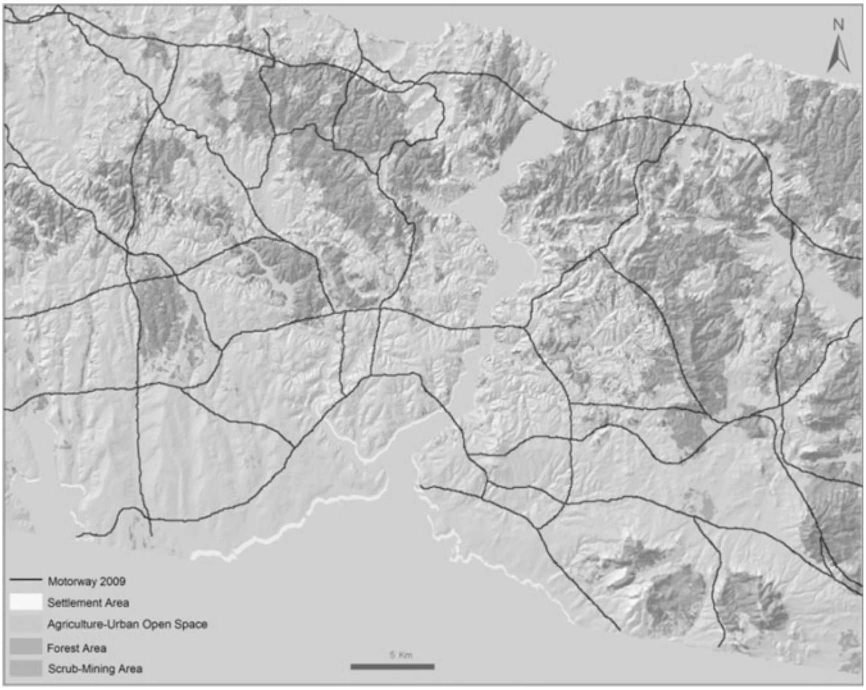


Fig. 8 Predicted land use map for the year 2030 (Ayazli 2011)

The postclassification technique was used to compare imagery for the year 2009 and 2030. It focused on change detection analysis for Istanbul and used ArcGIS 9.3 software. As a result, an image containing 126,608 pixels was obtained, which is approximately 20% of the study area. This represents what will become new urban areas. In other words, 83.17% of arable land/open spaces, 40.54% of forests-green lands, and 10.34% of water bodies will be urban spaces in 2030 (Fig. 9).

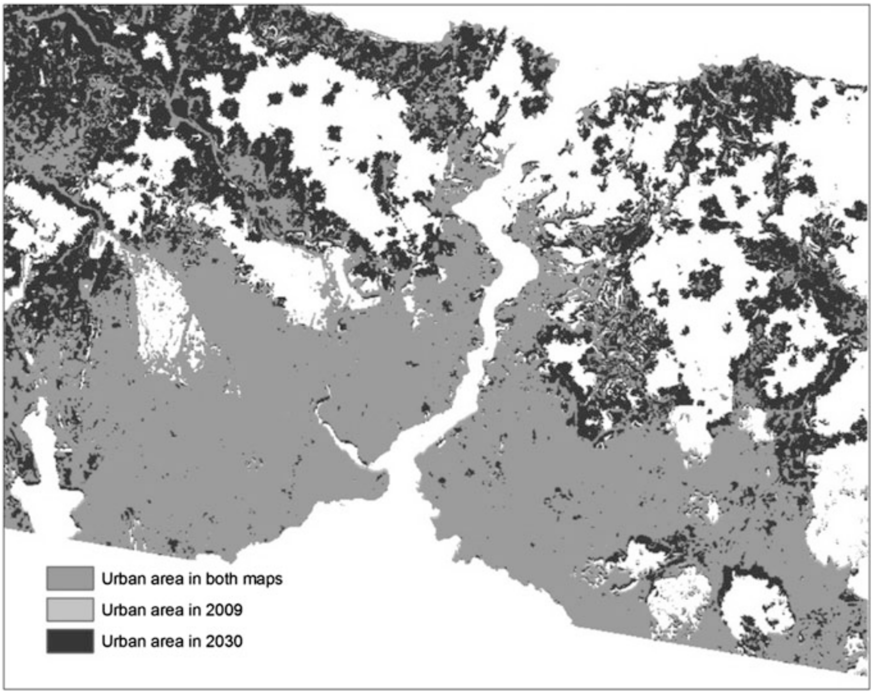


Fig. 9 Map comparison between the year 2009 and 2030

4 Conclusions

This study aimed to ascertain the effects of Istanbul's Bosphorus bridges on urbanization. It achieved this end by classifying satellite images into four time periods. It also created a simulation model for the year 2030. Landsat images, DEM, master plans, and motorways were used to determine lands that will transition from natural areas to urban areas.

As can be seen in Table 3, transportation networks have a great impact on urban growth in Istanbul. Therefore, new settlement areas tend to emerge along the highway routes. The urban areas gravitate toward these places. According to the results, Istanbul will continue growing northward. Approximately 40% of the forest areas and 83% of agriculture-urban open space in the north of the city will transform into settlement areas. The most important natural areas will be threatened by urbanization if the third bridge is built on the Bosphorus. This will lead to population growth and will exacerbate Istanbul's complex transportation problem. This is because the bridge will create its own traffic. To reach a permanent solution, primarily seaway and railway transportation must be supported.

The prediction and visualization of the potential land use change is the most important tool for raising social consciousness. For this purpose, various CA-based simulation software programs have been used for a number of projects worldwide. SLEUTH UGM is the most popular among them. Some project reports generated by using SLEUTH were published by media sources (Jantz et al. 2003, referring to Huslin 2002; URL 11). It has been proven that SLEUTH is an important tool for creating sensible reactions and raising public awareness about natural problems.

Acknowledgments The authors would like to thank Sivas Cumhuriyet University for supporting the study under CUBAP project M 355.

References

Ayazli IE (2011) Simulation Model of Urban Sprawl Driven by Transportation Networks: 3rd Bosphorus Bridge Example (in Turkish). Ph.D. thesis, Yildiz Technical University

Ayazli IE, Batuk F, Kleinschmit B (2010) Simulating landuse changes driven by a 3rd Bosphorus Bridge. ASPRS/CaGIS 2010 Fall Specialty Conference, Orlando, Florida, 15–19 Nov 2010

Baslik S (2008) Dynamic urban growth model: Logistic regression and cellular automata (Examples of Istanbul and Lisbon) (in Turkish). Ph.D. thesis, Mimar Sinan Fine Arts University

Batty M (2007) City and complexity: Understanding cities with cellular automata, agent-based models, and fractals. The MIT Press, Cambridge

Benenson I, Torrens PM (2004) Geosimulation automata-based modeling of urban phenomena. Wiley, West Sussex

Candau JT (2002) Temporal calibration sensitivity of the SLEUTH urban growth model. Master of Arts Thesis, University of California Santa Barbara

Celikoyan TM (2004) Monitoring and analysis of landuse changes in historical periods for the city of Istanbul by means of aerial photography and satellite imagery. Ph.D. thesis, Istanbul Technical University

Cheng J (2003) Modelling spatial and temporal urban growth. Ph.D. thesis, Utrecht University

Clarke KC, Hoppen S, Gaydos L (1997) A self-modifying cellular automaton model of historical urbanization in the San Francisco Bay Area. Env Plann B: Plann Des 24:247–261

Dietzel C, Clarke K (2007) Toward Optimal Calibration of the SLEUTH Land Use Change Model. Trans GIS 11(1): 29–45

EEA (2006) Urban sprawl in Europe, The ignored challenge. EEA Report No 10/2006

Jantz C, Goetz SJ, Shelley MK (2003) Using the SLEUTH urban growth model to simulate the impacts of future policy scenarios on urban land use in the Baltimore-Washington metropolitan area. Env Plann B: Plann Des 30:251–271

Kabadayi E (2008) Loyalty Istanbul, protect your Bosphorus (in Turkish). Atlas Magazine March 2008, İstanbul, p 180

Nechyba TJ, Walsh RP (2004) Urban Sprawl. J Econ Perspect 18(4):177–200

Oguz H, Klein AG, Srinivasan R (2007) Using the Sleuth urban growth model to simulate the impacts of future policy scenarios on urban land use in the Houston-Galveston-Brazoria CMSA. Res J Soc Sci 2:72–82

Sevik O (2006) Application of SLEUTH model in Antalya. Master of Science thesis, Middle East Technical University

Silva EA, Clarke KC (2002) Calibration of the SLEUTH urban growth model for Lisbon and Porto, Portugal. Comput Environ Urban Syst 26:525–552

SLEUTH Web Page, Overview http://www.ncgia.ucsb.edu/projects/gig/v2/About/bkOverview.htm

SLEUTH Web Page, Growth Rules (Accessed August 2010) http://www.ncgia.ucsb.edu/projects/gig/v2/About/gwRules.htm

SLEUTH Web Page, Self-modification (Accessed August 2010) http://www.ncgia.ucsb.edu/projects/gig/v2/About/gwSelfMod.htm

Turkish Statistical Institute Web Page (Accessed September 2011) http://www.tuik.gov.tr/

Istanbul Metropolitan Municipality Web Site (Accessed September 2011) http://www.ibb.gov.tr/tr-TR/Pages/Haber.aspx?NewsID=15773

SLEUTH Web Page, Naming Format (Accessed September 2012) http://www.ncgia.ucsb.edu/projects/gig/v2/Imp/imSetUp.htm#namingConvention

SLEUTH Web Page, Spontaneous Growth Rule (Accessed September 2012) http://www.ncgia.ucsb.edu/projects/gig/v2/About/gwSpontaneous.htm

SLEUTH Web Page, New Spreading Centre Growth Rule (Accessed September 2012) http://www.ncgia.ucsb.edu/projects/gig/v2/About/gwNewCenter.htm

SLEUTH Web Page, Edge Growth Rule (Accessed September 2012) http://www.ncgia.ucsb.edu/projects/gig/v2/About/gwEdge.htm

SLEUTH Web Page, Road Influenced Growth Rule (Accessed September 2012) http://www.ncgia.ucsb.edu/projects/gig/v2/About/gwRoadInflu.htm

Today's Zaman Newspaper Webpage, (16 February 2011) http://www.todayszaman.com/newsDetail_getNewsById.action?load=detay&newsId=235686&link=235686

Watkiss BM (2008) The SLEUTH urban growth model as forecasting and decision making tool. Master of Science thesis, at the University Stellenbosch

Wu X, Hu Y, He HS, Bu R, Onsted J, Xi F (2009) Performance evaluation of the SLEUTH model in the Shenyang Metropolitan Area of Northeastern China. Environ Model Assess 14:221–230

Xibao X, Feng Z, Jianming Z (2006) Modelling the impacts of different policy scenarios on urban growth in Lanzhou with remote sensing and cellular automata. Geoscience and Remote Sensing Symposium, IGARSS 2006, IEEE International Conference on July 31 2006–Aug. 4

Yu XJ, Ng CN (2007) Spatial and temporal dynamics of urban sprawl along two urban–rural transects: a case study of Guangzhou, China. Landscape Urban Plan 79:96–109

A New Method to Characterize Density Adapted to a Coarse City Model

Rachid Hamaina, Thomas Leduc, and Guillaume Moreau

Abstract Density is probably one of the most used indicators to characterize urban development. Even if it is a quantitative and properly defined measure, there are still difficulties in using it properly. This chapter proposes an updated approach to characterize urban density based on buildings' footprints. It can be applied to huge datasets and allows multilevel characterization of density. We first present an original partition of urban open space. This topology helps us to define a neighborhood function. We then adapt the ground space index and floor space index indices to the previously defined tessellation. The combination of the neighborhood function and the modified indices makes it possible to assess density iteratively. For each building, these values allow one to define the density profile, which is then used in a classification process. The results highlight spatial patterns and homogeneous areas. This transposable method is adapted to urban fabric characterization and surpasses old descriptive and low formalized classifications.

Keywords Density profile · Extended FSI (Floor Space Index) · Extended GSI (Ground Space Index) · Open space tessellation · Urban morphology

R. Hamaina (✉) · G. Moreau
LUNAM Université, Ecole Centrale de Nantes, CERMA UMR CNRS 1563, 1, rue de la Noë, Nantes, BP 92101 44321 Nantes Cedex 3, France
e-mail: rachid.hamaina@cerma.archi.fr
URL: http://www.cerma.archi.fr

G. Moreau
e-mail: guillaume.moreau@ec-nantes.fr

T. Leduc
LUNAM Université, CNRS, CERMA UMR CNRS 1563, 6, quai François Mitterrand, Nantes, BP 16202 44262 Nantes Cedex 2, France
e-mail: thomas.leduc@cerma.archi.fr

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_16,

1 Introduction

Three-dimensional (3D) city models are usually only available for the downtown area of cities, small particular areas, or individual urban landmarks. Furthermore, these 3D city models are usually only used for visualization purposes.

However, a coarse 3D city model can easily be derived from traditional topographic databases. It can be achieved by extruding building footprints by their corresponding heights. Such a coarse model is poor for visualization applications. It can nevertheless be used in a geographic information system (GIS) framework, thus offering wide capabilities for urban analysis. Indeed, coarse urban models have already been used to perform classification of building types (Henn et al. 2011) or to simulate urban blocks' densification (Perret et al. 2009).

Density is intensively used in urban geography, planning, and design. It is defined as a concentration measure of certain entities in a given area. The commonly used entities in urban analysis are population, dwellings, services, and jobs. Built-area (floor space) density is perhaps the most used to characterize urban development and its regulation.

Density is often confused with building type. For example, one assumes that detached houses are of lower density than attached housing types. Although this is generally true, it is not always the case. Also, a high-rise tower with large units set in a park-like area may be lower density than a set of detached houses on small lots (Forsyth 2003). In Berghauser, Pont, and Haupt (2007), a multivariable approach to density characterization is suggested. This approach uses four variables expressing intensity (floor space index, FSI), compactness (ground space index, GSI), pressure on nonbuilt space (open space ratio, OSR), and building height (layers, L).

However, because density is area dependant, the well-known modified area unit problem (Openshaw 1983) is unavoidable. Moreover, if the reference area is a spatial unit with an inadequate formalized definition, as are urban blocks and districts, the density measure can be of little relevance and can even be confusing.

When density indicators are used in descriptive (characterization) rather than prescriptive (normative) purposes, they are usually an aggregated measure over administrative areas (Smith and Crooks 2010) or a regular division of urban space (Koomen 2009).

An alternative characterization of urban form patterns is based on the use of fractals. Thomas et al. (2007) concluded that the use of the fractal dimension of built-up areas to characterize urban fabrics considers both morphology and internal structure, whereas density gives just a rough idea of the occupation of the surface. However, this fractal measure considers neither the buildings' 3D templates nor the neighboring open space morphology. Furthermore, it was pointed that the so-called fractal cities can be understood according to self-similarity in the sense of statistical analysis instead of pure theory and scaling invariance can be only defined within certain range of scales (Chen 2011).

Here, we propose a new formulation of urban density characterization by integrating open space morphological characteristics and a weighted neighborhood-aware density calculation. To achieve this, we associate a building's footprint data to a surrounding open space's geometry. This open space partition, based on building geometry, allows the measurement of density indicators by considering a reference area based on urban morphology and not on land ownership (as is traditionally done). In this way, urban space anisotropy is taken into account.

This open space partition is a well-formalized solution to identify the minimum area unit around a given urban building. Indeed, as a pure geometric method, it is reproducible independently from urban context. Moreover, it also overcomes not only potential lack of plot data availability, but also potential the spatial coverage problem between buildings and plot layers (frequent mismatch).

2 Methods

In this section, a three-step process is presented. The first one consists of defining a unique geometric item surrounding each building's footprint. This building's neighborhood, strongly related to the building item, makes it possible to partition the urban fabric. The partition method we present here has already been defined in Hamaina et al. (2012).

In a second step, intensity (FSI) and compactness (GSI) are revised and a definition based on the new spatial item (the building's neighborhood) is presented. Combining these extended GSI and FSI indices with the aforementioned neighborhood (at different distance levels) gives the possibility to define the density profile of a given building.

Finally, in a third step, building footprints of whole study area are classified into homogeneous subcategories based on profile density characteristics. The performed classification is a weighted neighborhood-aware classification because it takes into account GSI and FSI computed over several neighborhoods and weighted using a decreasing function.

2.1 Open Space Tessellation

2.1.1 Open Space Geometry

Placing a new spatial object in a space induces change in the configurational properties of this space. In the urban environment, every new building (as a new external morphological item introduced in its spatial context) induces in its immediate neighborhood a morphological change. It impacts the status with respect to several configurational parameters and physical phenomena (visibility, lighting, air circulation, movement, etc.) around the concerned building. The introduction of each new building corresponds to a spatial integration of a new attractor. By analogy, using a gravity model concept (i.e., the attraction that objects have for each other is set against the distance they are apart), this new building inserts a new item (cell) in the whole set of partitions. This new item can be delineated by a sort of skeleton through the open space between its neighboring buildings.

This space tessellation based on the Euclidean distance between neighboring buildings footprints is called a Voronoi diagram (see Fig. 1). Here, the tessellations represent the morphological influence of cells corresponding to individual buildings in the urban space.

Let us assume that the contours of this space tessellation (a sort of urban open space skeleton) are the edges of a street network. The same way, let us assume a pedestrian walks through the urban fabric using this (artificial) street network; in each of the tessellation nodes, the walker will face substantial morphological changes. More precisely, these punctual locations correspond to the most significant places in terms of morphological changes.

The results of this tessellation are three new spatial objects: the nodes (0D) in the Voronoi diagram intersections that are the locations of significant change on morphology, the skeleton edges (1D) that are developed artificial streets, and the

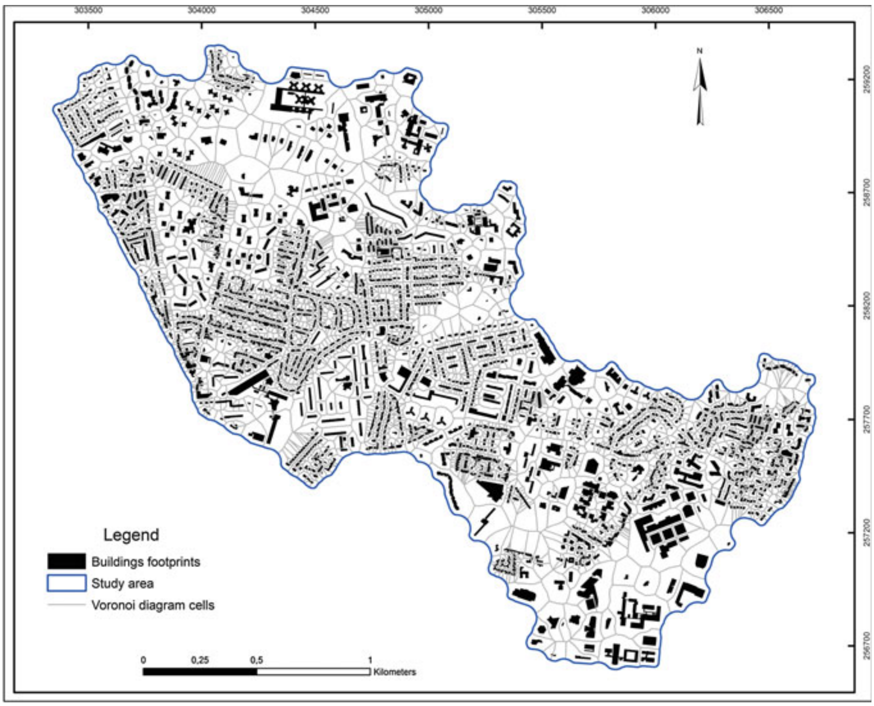


Fig. 1 Building footprints and corresponding Voronoi diagrams in a given urban area (northern districts of Nantes city, France)

Voronoi diagrams (2D) that are the morphological influence of cells. The latter are considered here as some vital space all around each building (required to supply air and light) and by analogy with ownership-based division, which are just a sort of morphological plot.

Urban environmental conditions strongly depend on urban space morphology. As an example, the amount of air (ventilation) and natural lighting of buildings depend on the urban space openness and consequently on surrounding open space geometry. We believe that these new spatial items based on morphological characteristics are more pertinent (for morphological issues, at least) than the traditional ones based on administrative, ownership, or other nonstandardized partitions.

2.1.2 Neighboring Properties

The Voronoi diagram partition topology allows one to define a neighboring function based on the shared borders (in the sense of a spatial intersection) of these polygons. The function allows one to define each building "B" several neighborhoods (from 1 to n) and several neighbors (buildings) for each neighborhood. This is based on spatial topologic relationships as follows.

Let $V = V_1, \ldots, V_n$ be the set of Voronoi diagram cells corresponding respectively to the set of buildings $B = B_1, \ldots, B_n$ in a given urban area. V is the space partitioning of this area.

Let $\mathrm{NH}_1(B_i)$ be the first neighborhood of a building "B_i" (the set of all adjacent cells) and $\mathrm{NB}_1(B_i)$ be its first neighbors (the set of all corresponding buildings).

$$\left.\begin{array}{c} \text{if } \; V_i \cap \{V_j, V_k, \ldots, V_m\} \neq \varnothing \text{ then} \\ \mathrm{NH}_1(B_i) = \bigcup\{V_i, V_j, V_k, \ldots, V_m\} \\ \text{and} \\ \mathrm{NB}_1(B_i) = \{B_j, B_k, \ldots, B_m\} \end{array}\right\} \tag{1}$$

The second neighborhood NH_2 and neighbors NB_2 of the building B_i are iteratively constructed from Eq. (1) by replacing V_i by NH_1. The process is iterative. At the n^{iest} iteration, NH_n and NB_n are constructed from Eq. (1) by replacing V_i with NH_{n-1}.

$$\left.\begin{array}{c} \text{if } \; \mathrm{NH}_{n-1} \cap \{V_j, V_k, \ldots, V_m\} \neq \varnothing \text{ then} \\ \mathrm{NH}_n(B_i) = \bigcup\{\mathrm{NH}_{n-1}, V_j, V_k, \ldots, V_m\} \\ \text{and} \\ \mathrm{NB}_n(B_i) = \{B_j, B_k, \ldots, B_m\} \end{array}\right\} \tag{2}$$

2.2 Density Characterization

2.2.1 Modified GSI and FSI

As mentioned, the aim of this subsection is to use a building's neighborhood (as a well-formalized spatial object) to revise FSI and GSI definitions. Thus, these two commonly used indicators in urban studies are transposed to the previously defined morphological plot. Even if these are traditional indicators, their calculation over artificial morphological plots (instead of administrative plots) gives them more relevance for urban morphology characterization purposes.

In our renewed approach, the GSI corresponds to the ratio between the building's footprint area and the corresponding Voronoi diagram's cell area. The FSI is the ratio between the building's volume and the corresponding Voronoi diagram's cell area (see Fig. 2).

2.2.2 Density Profiles

Density indicators (GSI and FSI) are computed for each building and at different distance levels (i.e., over successive i-neighborhoods from $i = 1$ to n). At level zero, the only building's neighborhood is the one of the morphological plot itself, and corresponding densities have already been defined previously. Densities of a building at the ith neighborhood are:

$$\text{Floor space index } (\text{FSI})_i(B) = \frac{\text{area}(B) + \sum_{j=1}^{i} \text{area}(\text{NB}_j)}{\text{area}(\text{NH}_i)} \tag{3}$$

$$\text{FSI}_i(B) = \frac{\text{volume}(B) + \sum_{j=1}^{i} \text{volume}(\text{NB}_j)}{\text{area}(\text{NH}_i)} \tag{4}$$

A density profile for a building B is constructed by the variation of GSI or/and FSI over several neighborhoods (see Fig. 3).

Density profiles offer a useful tool to characterize local urban density based on morphologic characteristics and perhaps associate some particular profiles to some physical behavior of the concerned buildings or group of buildings.

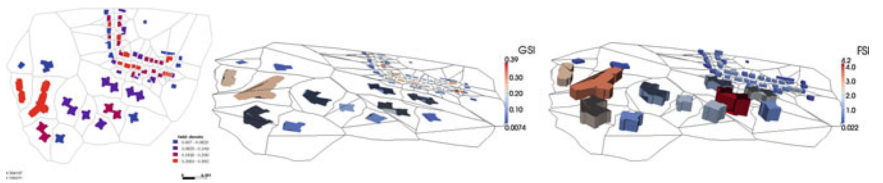


Fig. 2 Buildings density indicators: ground space index (*left* and *center*) and floor space index (*right*) in a small urban area

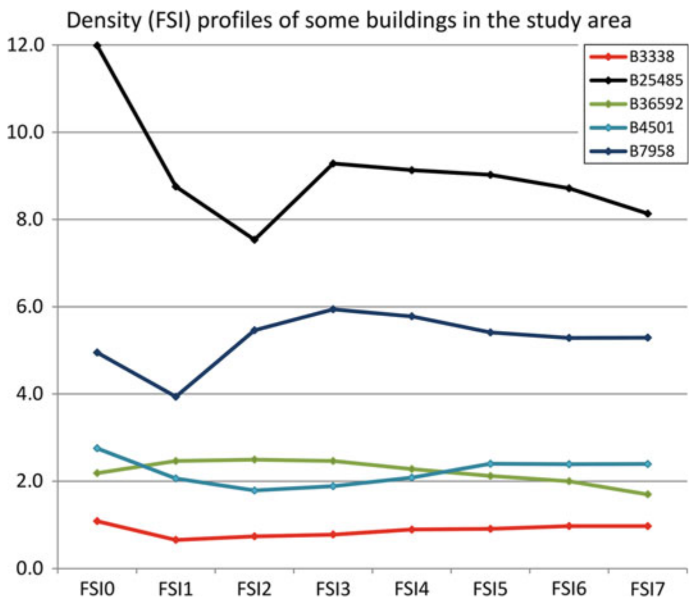


Fig. 3 Floor space index (FSI) profiles (over seven neighborhoods) of a sample of five buildings

Taking into account neighborhood geometry (morphologic cells) and neighboring building density is a key issue because each building influences its neighbors and is itself influenced by these. The characterization of individual buildings of an urban fabric is not sufficient to characterize the whole of the urban fabric itself.

2.3 *Urban Classification Based on Density Indicators*

For the classification of buildings, we use a traditional k-means process using GSI and FSI over several neighborhoods. Thus, the input data of the clustering process are a matrix of dimensions $(2n + 2,\ \ m)$, where n is the number of processed neighborhoods, $(2n + 2)$ is the number of variables, and m is the number of building footprints to classify.

GSI and FSI values are normalized and weighted following a decreasing function of the form $1/x$ before being used in the classification (see Fig. 4). The 0.3679 factor is approximately equal to $\frac{1}{\sum_{i=1}^{8} \frac{1}{i}}$. It is built in such a way as to "normalize" the weighting function. The objective of the weighting process is to assign less impact to neighboring buildings on density as one moves away from the concerned building.

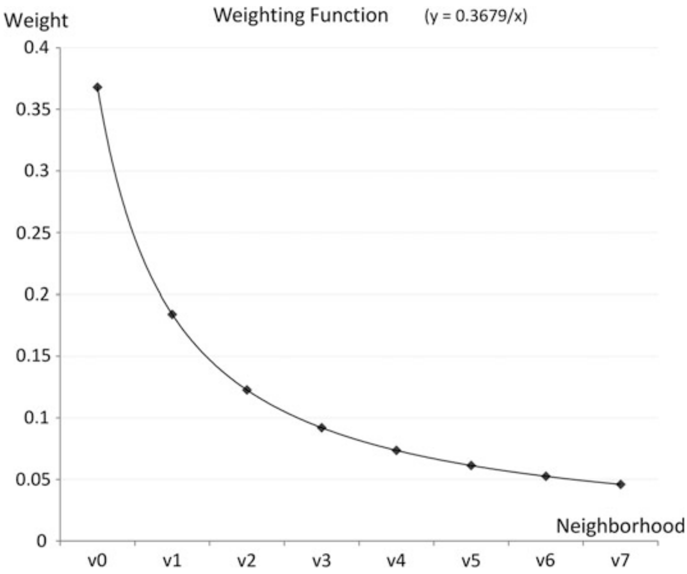


Fig. 4 The mathematical inverse function used for density weighting over seven successive neighborhoods

3 Results

3.1 Study Area

The study area is delineated by the beltway boulevard of Nantes city. Indeed, this freeway produces a physical (and functional) incision in the urban space and delimits a homogeneous urban area in its interior. The only input data used are building footprints. These spatial data are derived from the topographic database provided by the French Geographic Institute (the so-called BD TOPO).

3.2 Density Characterization

Seven neighborhoods were included. Figures 5 and 6 respectively represent the results of GSI and FSI over the zero (the cell of the concerned building itself) and these seven neighborhoods.

Density (both GSI and FSI) values in the zero's neighborhood are quite heterogeneous. This result is not surprising and it is expected because of the urban architecture and form diversity in cities in general and particularly in Nantes. At this local scale, it is difficult to highlight specific spatial patterns of density.

However, calculations over successive neighborhoods have a smoothing effect on local heterogeneity, so some spatial patterns can be recognized gradually. This

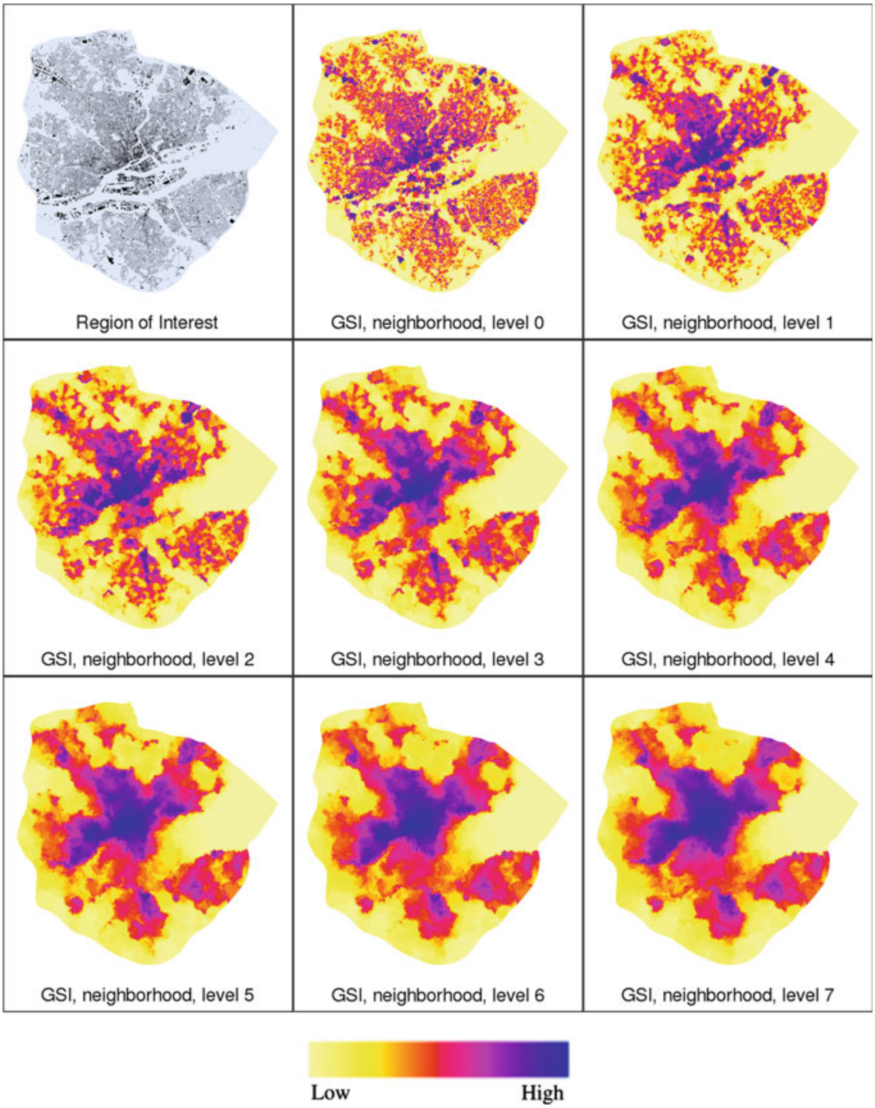


Fig. 5 Ground space index (GSI) results in study area over the "zero" and seven successive neighborhoods

pattern is quite similar to the traditional gradient center periphery, which gradually becomes clearer when moving from the zero to *n*th neighborhood, even if this is broken in many areas.

First, the rivers running in the study area and their watersheds break this gradient following four axes: an east–west axis (Loire River), a northern east axis (Erdre River), a southern axis (Sèvre River), and a northern west axis (Chézine River).

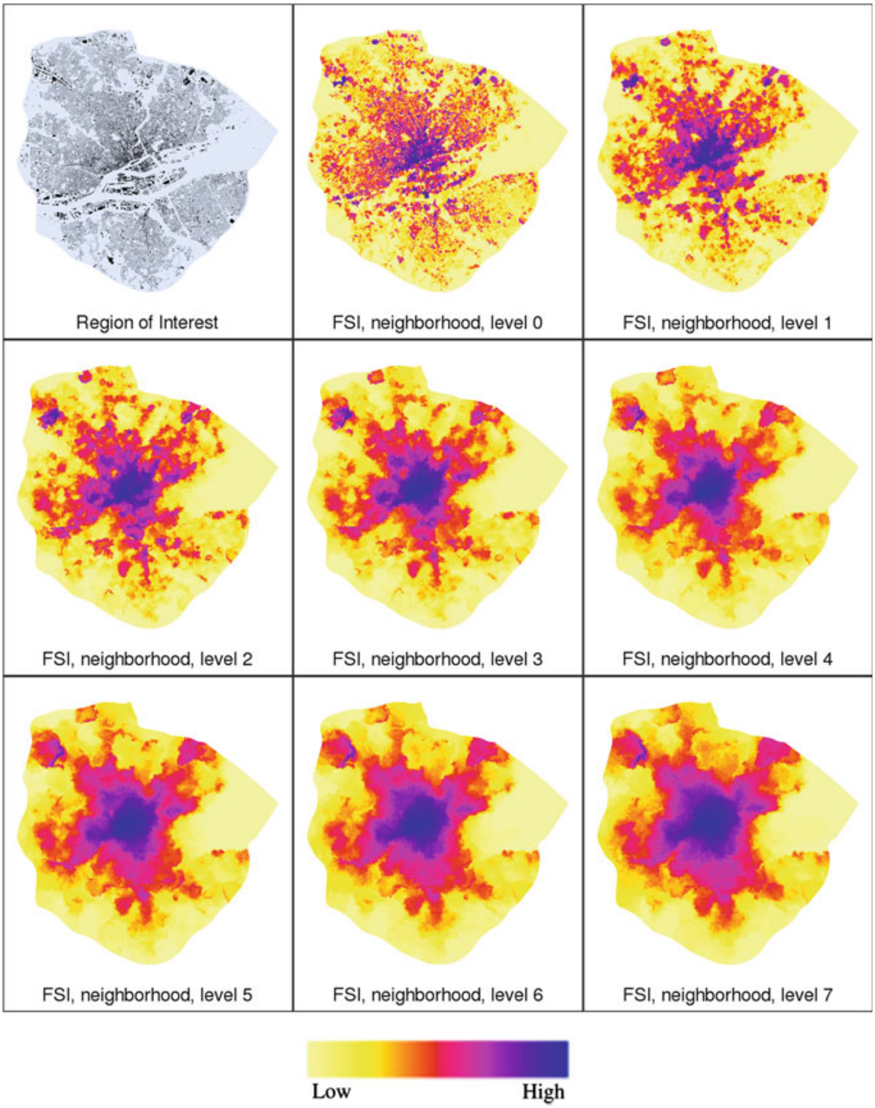


Fig. 6 Floor space index (FSI) results in study area over the "zero" and seven successive neighborhoods

Then, some natural nonbuilt areas, located especially in the southern west (nonurbanized area) and in the northern east (recreational area), exhibit the same areas of very low density.

Because GSI is a 2D indicator while FSI is a 3D one, they do not follow exactly the same spatial pattern, especially in the southeast and some western parts of the study area. The former corresponds mostly to low-rise housing areas (Saint-

Sébastien sur Loire), which is why it exhibits high GSI values and low FSI values. The latter area contains high-rise buildings (mostly large blocks), which is why it exhibits high values of FSI and low values of GSI.

3.3 Urban Classification

3.3.1 Urban Density Clusters

Figure 7 represents 10 clusters resulting from the classification of the study area's buildings. These are organized from higher density to lower density areas. Each cluster has different density characteristics from the others, as shown by density profiles. However, for better intelligibility of cluster typology, we group them into the traditional well-known and used urban typologies.

Within the same group, various clusters (if this is the case) express some gradient on density. We can distinguish each of them even if no formalized descriptive definition is associated with each one.

Therefore, the first group, composed of clusters 5 and 7, represents the downtown, which is a highly dense area (organic historic center). Cluster 7 is denser than cluster 5, because the latter includes areas that are adjacent to several open spaces, such as the *Cours des* 50 *Otages's* large avenue, and the Erdre and Loire Rivers. Very few industrial buildings are included in this group because they are enclosed by other large buildings, thus creating (for only these buildings) a similar morphology to that of the city center buildings.

The second group, including clusters 6, 8, and 9, is formed by downtown extensions until the first beltway of the city, some industrial buildings (*Ile de Nantes*), a few equipment buildings, linear facades (blocks) of some main (and thus dense) streets, and few ancient hamlets' cores. As in the first group, density is decreasing, moving respectively away from the clusters 8, 6, and 9.

The third group, formed by cluster 0, corresponds to medium-density areas. These are mostly midrise areas (collective housing blocks), a few equipment buildings, and terraced individual houses.

The fourth group, composed of cluster 4, corresponds to low-density areas. These include low-rise housing (mostly large individual houses) but also high-rise buildings set on park-like areas.

The last group, composed of clusters 1, 2, and 3, corresponds to very low density areas. These include mostly low-rise individual housing set on large open spaces.

3.3.2 Mean Density Profile of Clusters

For each cluster, a mean density (FSI and GSI) profile is processed. These are represented respectively in Figs. 8 and 9.

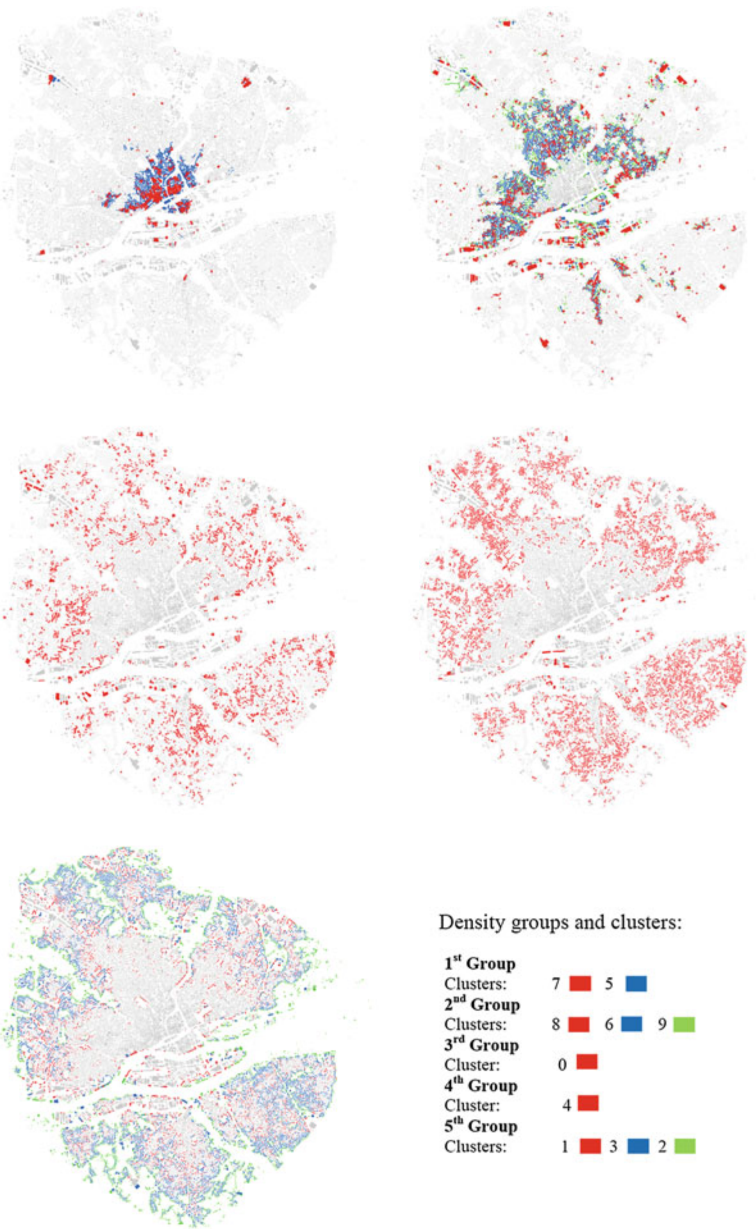


Fig. 7 Ten clusters grouped on five categories of urban density

The most interesting finding from these mean density profiles is that the smoothing effect of the neighborhood-aware calculation of density represents an intelligent method of aggregation and grouping individual buildings on homogeneous density areas because the urban space is anisotropic.

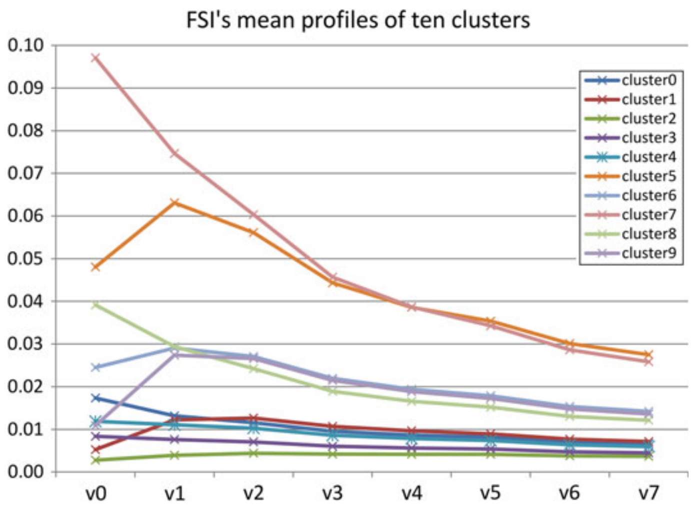


Fig. 8 Floor space index (FSI) mean profiles of the ten clusters processed in the study area

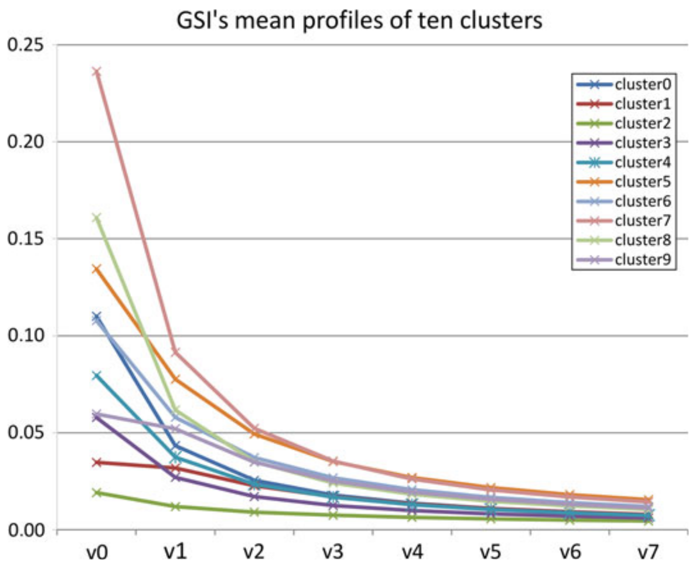


Fig. 9 Ground space index (GSI) mean profiles of the ten clusters processed in the study area

Moving through successive neighborhoods, several profiles tend toward each other to form some groups. Nevertheless, even on an advanced aggregation iteration (with several neighborhoods used), the main three density categories—high-density areas (downtown), high-to medium-density areas (downtown immediate extensions), and peripheral low-density housing areas—are still well distinguished. This is clearer in the FSI profile than in the GSI profile.

The first category is characterized by a fast-decreasing density profile over successive neighborhoods. The second one is characterized by a moderate (slow) decreasing profile, whereas the third category has a roughly constant profile.

4 Conclusion

This chapter formalized a modified GSI and FSI based on a morphologic tessellation of urban open space. The topology of this tessellation allows one to define a neighboring function, thus processing a density characterization by taking into account several successive neighborhoods.

Densities computed over several neighborhoods allow the construction of density profiles of each building. Then, these individual buildings are classified according to density (GSI and FSI) values weighted by a decreasing function over neighborhoods.

The results show that modified GSI and FSI profiles are useful tools for local characterization of density. Furthermore, the aggregation over neighborhoods as defined here is relevant for highlighting spatial patterns of density. Finally, the classification of individual buildings according to these modified density indicators highlights traditional urban fabric categories used in urban studies and allows one to perform even more detailed classification, going beyond descriptive and low formalized urban classifications.

The density analysis developed here uses only a building's footprint layer. It offers both local and aggregate density characterizations and can be performed on huge datasets in a GIS framework.

Regarding the legitimate domain, this method is relevant for any kind of urban area. One of its main goals is thus to delineate and classify urban fabrics.

Finally, it would be worth while to compare numerically the resulting clusters with the results obtained using classical methods based on local averaging. Therefore, GSI and FSI values should be studied using other partitioning techniques, such as regular grid, building blocks, or statistical units (all methods that are less adaptative to local sample density).

References

Berghauser Pont M, Haupt P (2007) The relation between urban form and density. Urban Morphol 11(1):62–66

Chen Y (2011) Derivation of the functional relations between fractal dimension of and shape indices of urban form. Comput, Environ Urban Syst 35(6):442–451, (November 2011)

Forsyth A (2003) Measuring density: working definitions for residential density and building intensity. Design Brief, 8:8

Hamaina R, Leduc T, Moreau G (2012) Towards urban fabrics characterization based on buildings footprints. In: J Gensel , D Josselin, D Vandenbroucke (eds). Bridging the geographic information sciences, lecture notes in geoinformation and cartography, Springer, Berlin, Heidelberg, pp 327–346

Henn A, Römer C, Gröger G, Plümer L (July 2011) Automatic classification of building types in 3D city models. GeoInformatica 16(2):281–306

Koomen E, Rietveld P, Bacao F (2009) The third dimension in urban geography: the urban-volume approach. Environ Plann B: Plann Des, 36(6):1008–1025

Openshaw S (1983) The modifiable areal unit problem. CATMOG - concepts and techniques in modern geography 38:41

Perret J, Boffet Mas A, and Ruas A (2009) Understanding urban dynamics: the use of vector topographic databases and the creation of spatio-temporal databases. In 24th international cartography conference (ICC09), Santiago de Chile

Smith DA, Crooks AT (2010) From buildings to cities: techniques for the multi-scale analysis of urban form and function. Technical report 155, centre for advanced spatial analysis (UCL), London

Thomas I, Frankhauser P, De Keersmaecker ML (2007) Fractal dimension versus density of built-up surfaces in the periphery of Brussels. Papers in regional science 86(2):287–308, (June 2007)

Part VII
Marine and Coastal GIS (for the Arctic Region)

The Skipper's Advisor: An Expert System for Coastal Navigation

Ruslan Sorokin

Abstract Now is the time to revive the idea of the widespread use of expert systems in various fields using modern hardware and software technology. Navigation in littoral waters is an appropriate application for expert systems because it requires many different kinds of knowledge and expertise. The availability of information on the Internet, including data suitable for machine processing, may facilitate the creation of such a system. However, one must conduct an in-depth study of the issue to reach an effective solution to this challenge. This chapter describes the ideas embodied in a draft of the expert system, a research prototype of the system, and the experiments using it.

Keywords Expert system · Coastal navigation · Semantic Web · Speech synthesis and recognition · Natural language processing · 3D imaging

1 Introduction

The peak of popularity for expert systems was in the 1990s. The creation of expert systems has been attempted in all fields of application. However, they only managed in be effective for real problems in a limited number of application areas. One reason for this was the level of computing and information technology at the time. Since then, computers and software have made a giant leap. Now is the time to revive the idea of the widespread use of expert systems in various fields using modern hardware and software technology.

Navigation in littoral waters is an appropriate application for expert systems. In addition to general knowledge about navigation, this application requires

R. Sorokin (✉)
Research Laboratory of Object-Oriented Geo-Information Systems, St.Petersburg Institute for Informatics and Automation of RAS, Galerny 5, Saint-Petersburg 199188, Russia
e-mail: sorokin@oogis.rushal@oogis.ru

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_17, © Springer-Verlag Berlin Heidelberg 2014

knowledge of the local pilotage, local meteorology, local laws and regulations, etc.—that is, extensive heterogeneous knowledge with tough local binding. Of course, a local pilot is an expert in littoral navigation, but this pilot is very expensive. An expert system can be used to replace the pilot.

Due to the semantic Web, an increasing amount of data that is structured and suitable for machine processing is available on the Internet. Further development in recent years has occurred in speech synthesis and speech recognition, as well as in natural language processing. A modern expert system, which serves as an advisor to the skipper, could be built on these technologies, along with 3D-imaging technology. This chapter presents the ideas inherent in a research prototype of such a system, as well as the details of its implementation and areas for improvement. The prototype is open to the trials and use.

2 Basic Ideas

2.1 Moving Cloud of Data

In an expert system for navigation in coastal waters, all available data of possible interest for navigation should be collected by an on-board communication equipment within a certain radius around the moving ship. Because the ship is moving, over time, some of the data with local binding lose relevance and can be erased from the memory. Thus, the ship seems like is moving within a cloud of data that are relevant in a given time and in a given place. The size of the cloud is quite limited, primarily by the high-speed capabilities of the ship, and therefore may be supported by modern computing. Data for the cloud can be obtained in real time from the Internet in areas covered by mobile networks, as well as from the databases on storage devices. Data from the on-board navigation system, if any, is added to this information. However, if only known the current position (from systems like GPS or GLONASS) informative enough cloud data allows us to replace the navigation system, as will be explained in the next section. In extreme cases, even the absence of external coordinate information can be replaced by the expert system

Data from the cloud are processed by the expert system and are used to analyze the situation, make recommendations, and respond to various user requests.

2.2 Virtual Situation

Next, in an expert system for navigation in coastal waters, the virtual situation around the moving vessel should be constructed based information from the cloud data. The virtual situation is a set of virtual objects cloud data by expert rules that bind data with objects and objects with each other. Such objects may be, for example, islands, rocky shoals and shallows, and other obstacles; routes for safe passage between them; and navigational landmarks, including non traditional ones, such as populated places on the coast and near the coast, which give a distinct glow to overcast sky, especially at night. An expert system can identify objects such as areas of active navigation, direction of way in them, types of ships (high speed, large-capacity, sailing, etc.), fishing and set nets areas, and zones of different administrative and legal regulation. Virtual objects are used along with the data in the rules of the expert system to make recommendations and answer user questions.

2.3 Speech Interface

The expert system for navigation in coastal waters may also use a speech interface between the expert system and the user. Car navigators have successfully used speech output of commands, which allows the driver to focus on the road. This factor is no less important in the handling of high-speed boats in the cramped conditions of coastal waters, especially at night and in bad weather. The addition of speech input to speech output increases speed and security of communication between a skipper and the expert system because the input of complex queries to the system using conventional input, such as keyboard, mouse, or joystick, is extremely difficult in waterproof clothing on the tossing sea and at night. In modern expert systems, speech output can include not only standard phrases, but also synthesis of arbitrary speech. Similarly, speech input can include not only fixed commands but also a recognition of questions on limited natural language.

2.4 Three-Dimensional Model of Situation

By using the cloud of data along with sophisticated methods of data processing, the expert system can synthesize a three-dimensional (3D) visual model of the situation and present it to the user in real time in the fog, at night, or during bad weather when visibility is poor, for example. This synthesized 3D-display can be invaluable to help the skipper make quick and correct decisions. On this display, one can remove objects or make them transparent when they are restricting visibility, such as an island or a large vessel moving parallel, to see what is going on behind the ship. Also, it is possible on an accelerated time scale to predict the development of a situation and the relative position of competitors and obstacles to movement in arbitrary moments of time. A synthesized 3D-picture combined a voice description allows the skipper to visualize an adequate visual model of the situation beyond the actual visibility in real-time, preparing him/her in advance to make the right decisions.

3 Related Works

In recent years, literature has again emerged on the application of artificial intelligence techniques and expert systems in the domain of maritime navigation. They integrate well with the modern concept of e-navigation, which consists of the harmonized collection, integration, exchange, presentation, and analysis of marine information on-board and ashore by electronic means to enhance berth-to-berth navigation and related services for safety and security at sea, along with protection of the marine environment[1]. Maritime navigation expert systems based on the classic shell CLIPS were described in Calfee and Rowe (2002). More and more researchers have drawn attention to the problem of representing marine knowledge in modern paradigms, in particular with the help of ontologies. The works of Malyankar (1999a, 1999b) studied the sources of ontological knowledge, including geodata standards, governing documents, manuals and guides, tabular data of tides and weather, and capture of commonsense knowledge. They are also considered as the problems of building based on these knowledge ontologies and types of queries for retrieval of information from knowledge bases. Malyankar (2002) introduced an ontology of electronic nautical charts using the S-57 standard and the marine information markup language (MIML). Banas and Breitsprecher (2011) studied the problem of knowledge representation from the main document for seafarers COLREGS-72 and good seamanship knowledge; they also described a typical expert system as a part of a navigational decision support system (NDSS). Approaches to create an NDSS based on expert systems were also considered by Nikitakos and Fikaris (2009) and Pietrzykowski and Uriasz (2010).

At the present stage of development of methods and means of artificial intelligence, the main feature is the Internet, which in the heyday of artificial intelligence was absent. All the necessary knowledge for the expert system had to be present inside it. Now, however, expert systems can use knowledge distributed throughout the whole Web. Free-for-all network services such as Google Maps can be considered for the general purposes of the geographic information system (GIS). Even more exciting are projects such as openstreetmap (OSM), which combines the features of GIS and Wikipedia. Already in many parts of the globe, OSM provides more detailed maps than Google Maps, full of useful information for the traveler. Codescu et al. (2011) used OSM to develop a system of information support for travelers that is focused on different activities. It uses an approach based on mapping the ontology of spatial activities into the ontology of OSM tags, which is used by voluntary authors of OSM. The OpenSeaMap project is a child of the OSM project in the area of marine spatial information. The authors hope to soon turn it into an all-in-one workstation for skippers (Bärlocher 2012).

Another important feature of modern information technology is the use of 3D-visualization and animation as an external representation of the virtual reality. Of

[1] e-Navigation. Frequently Asked Questions (Version 1.6 dated September 2011) http://www.iala-aism.org.

special interest for navigation is 3D-cartography. A comprehensive analysis of the current state of the topic and specific technical solutions for the implementation in a modern expert system for marine navigation are offered in the work of Goralski et al. (2011). Ray et al. (2011) presented a 3D-virtual environment and a distributed system for collecting and processing information. It performed a visualization for the general public of a sailing regatta held in the Brest (France) coastal area in May 2010, in real time, and subsequent in-depth analysis and evaluation of this experiment by experts.

The emergence and development of the Internet has influenced all directions of artificial intelligence, including natural language processing (NLP). While the semantic Web promotes the idea of structuring information on the Internet for machine processing by the annotation of web pages with the help of ontology, the NLP takes the idea of direct extraction of structured data from the pages written by humans. The current state of research in the field of NLP is presented in the monograph by Daniel et al. (2008). Advances in this field are used in open-source speech recognition (Huang et al. 1992) and speech synthesis (Schröder and Trouvain 2003) systems that achieved professional quality. They can form the basis of a speech interface for the navigation expert systems. This task is simplified by using an interface such as the Standardized Maritime Language (Katarzynska 2009) as a professional subset of natural language.

4 Research Prototype of the Expert System

To verify the correctness of the ideas presented in here, *OOGIS of SPIIRAS* developed a research prototype of an expert system for navigation in coastal waters. The system is based on a visual environment for simulating complex spatial processes of the same laboratory, as well as open-source, well-known software (Sorokin 2011). The prototype itself is also an open-source product that can be downloaded from the laboratory's website for free use and experimentation. The prototype is in active development, with the ongoing addition of new data sources, methods, and systems for their processing. Currently, it includes the following components:

- visual environment for simulating complex spatial processes—*DroolsTabScen*,
- system of knowledge representation based on ontology—*Protege*[2],
- library of GIS—*OpenMap*[3],
- shell of expert systems—*Drools Expert*[4],
- speech synthesis system—*Mary* (Schröder and Trouvain 2003),

[2] ProtegeOntology editor and knowledge-based framework Protege, http://protege.stanford.edu.

[3] Open systems mapping technology OpenMap, http://openmap.bbn.com.

[4] Declarative rule based environment Drools Expert, http://www.jboss.org/drools/drools-expert.html.

- Web service *Google* for speech recognition,
- Web services for natural language processing (*OpenCalais*[5], *Yahoo terms*[6], *ReVerb*[7]).

The data sources used in the system are freely available geodata and data of semantic Web (*OpenStreetMap*[8], *GeoNames*[9], *Wikipedia*[10], *DBPedia*[11], *Linked-GeoData*[12], *GeoRSS*[13]). The system is developed using a language for the visual design of scenarios, DroolsTabScen; the expert system shell, Drools Rule Language; and the programming language, *Clojure*.

5 Implementation Details of the System Prototype

The ontology editor Protege established a domain ontology of navigation in coastal waters. It includes classes of objects and concepts in the domain and their properties. The ontology is open and is constantly updated and refined with the development and improvement of the system.

Based on the ontology and expert knowledge, sets of rules were developed for the expert system and simulation scenarios. With the help of an integrated simulation system, objects can be moved in geographical space according to scenarios in a given time scale, such as the vessel on which the skipper and the expert system are situated.

Periodically, as the ship moves, the system downloads all available data from OpenStreetMap within a certain radius around the current location of the vessel; these data describe objects on a map, such as islands, pieces of the coastline, lakes, rivers, populated places, etc. Simultaneously, geographical names in the area of navigation are downloaded from the website GeoNames. This imitates the process of data cloud formation. After that, the system performs data fusion in the cloud and creates new objects in the knowledge base in accordance with the ontology, edits changed properties of existing objects, and removes objects that have lost relevance (left astern). Thus, the virtual situation is established and kept up to date.

---

5 Semantic information extraction service OpenCalais, http://www.opencalais.com.

6 Yahoo content analysis web service, http://developer.yahoo.com/search/content/V2/contentAnalysis.html.

7 Open information extraction software, http://reverb.cs.washington.edu.

8 Free editable map of the world OpenStreetMap, http://www.openstreetmap.org.

9 Geographical database GeoNames, http://www.geonames.org.

10 Free encyclopedia that anyone can edit Wikipedia, http://www.wikipedia.org.

11 Knowledge base of structured information extracted from Wikipedia—DBPedia, http://dbpedia.org.

12 Linked geographical data knowledge base, http://linkedgeodata.org.

13 Format used to publish frequently updated content with geographical location binding GeoRSS, http://georss.org.

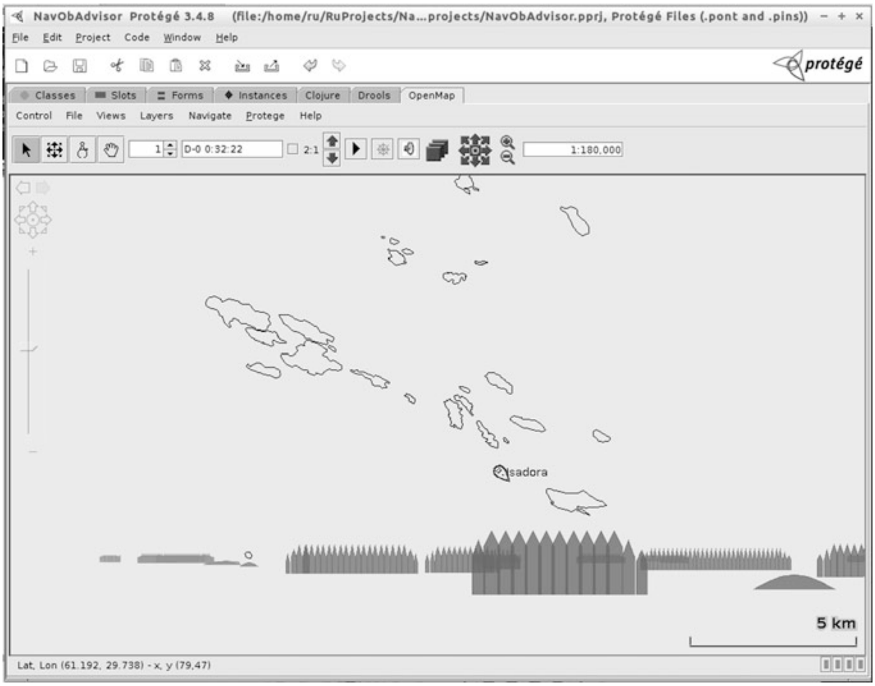


Fig. 1 View astern of a yacht Isadora on the synthesized 3D-display (HUD)

The "virtual radar" then begins to work, scanning the surrounding virtual space at a certain frequency to determine the relative positions of objects and define various relationships between them and the ship, such as "behind the island," "straight ahead," "on the left abeam," "between the islands," etc. Virtual radar is implemented as a set of rules of the expert system.

A separate set of rules is responsible for issuing recommendations to the skipper. At certain times, such as when changing course as the relative positions of the objects are changing, the expert system asks the skipper what is now straight ahead or on abeam. In addition, the system responds to user questions; for example, when asked the question "Where is the island of Stormy?", the system can answer "On the left abeam behind the islands of Boiler and Hare" based on the current relationship between the objects in the knowledge base in this moment of time. To answer user questions, also used other Web services are also used, such as Wikipedia, GeoRSS, etc., to get extended information about nearby objects, such as attractions, food service, medical facilities, etc. Verbal communication with the user is implemented using the speech synthesis system "Mary" and speech recognition by "Google Speech API".

The system implements a prototype of head-up display (HUD), which displays a 3D-picture of the virtual situation over a map, that, if desired, can be disabled. This prototype used a fairly simple mathematical 3D-model that creates a quite clear depiction of the real situation. In particular, (Fig. 1) shows how the features in the knowledge base are displayed as covered with fir trees and bare rocky islands. The islands on the 3D-display are shown using transparent color, allowing one to see which islands are behind the other islands, and to control an accuracy of the speech description of the situation (Fig. 1).

6 Discussion

The first studies of the expert system prototype were fruitful and demonstrated viability of the ideas embodied in it. The studies were conducted by simulating small-sized vessel (yacht, motor boat) voyages in different parts of the globe. Difficult navigation areas in littoral waters and lakes were selected, including the north western region of Lake Ladoga, Vyborg Bay in Russia, Lake Saimaa and Abo Åland archipelago in Finland, Stockholm archipelago in Sweden, Orkney Islands in the UK, etc. Researchers were allowed to collect a wealth of material from the public domain that is useful for navigation information. The volume and nature of the data vary considerably for different parts of sailing.

Accomplished investigations revealed a number of important new features of the system. First, the material collected during these "virtual" voyages can be stored for a real voyage, thus, making the system autonomous from external data sources. Second, the timing of passages can be made more precise during the "virtual" voyages, therefore clarifying the original travel plans. Third, during the "virtual" voyage, with the help of continuous 3D-modeling of situations and voice guidance, the system can train navigators and prepare them for the upcoming real voyage and unexpected situations they may encounter.

A survey helped to identify areas for further improvement of the research prototype and the studies. They include:

- adding knowledge of meteorology,
- further expansion and improvement of the ontology of navigation in coastal waters,
- further accumulation of expert knowledge governing this area, in sets of rules for the expert system,
- expansion of the list and systematization of data sources for the system,
- use of effective storage for collected data with advanced query logic [variants: triple storage such as AllegroGraph[14] or system Datomic[15]],
- connection of natural language processing systems to create a better and more flexible interface between the user and the system [variant: Stanford NLP Group software[16]],

[14] Graph database AllegroGraph, http://www.franz.com/agraph/allegrograph.

[15] Database of time-based facts Datomic, http://www.datomic.com.

[16] The Stanford Natural Language Processing Group, http://nlp.stanford.edu.

- connection of autonomous speech recognition system [variant: CMU Sphinx Huang et al. (1992)],
- connection of more sophisticated 3D-visualization tools [variant: software of Goralski et al. (2011)].

The work on the expert system for coastal navigation is in its early stage, so this article is more about the possibilities for the future research than a report about accomplished work. The ultimate scientific goal of this project is accumulation that is sufficient for structuring domain knowledge of coastal navigation. The ultimate practical goal is the creation of a commercial expert system for coastal navigation and a sail simulator for training small craft navigators.

References

Banas P, Breitsprecher M (2011) Knowledge base in the interpretation process of the collision regulations at sea. Int J Mar Navig Saf Sea Transp 5(3):359–364

Bärlocher M (2012) OpenSeaMap—die freie seekarte. Hydrographische Nachr 91:10–13

Calfee S, Rowe NC (2002) An expert system and tutor for maritime navigation rules. Naval Postgraduate School, Computer Science Department, Monterey

Codescu M, Horsinka G, Kutz O, Mossakowski T, Rau R (2011) DO-ROAM: activity-oriented search and navigation with openstreetmap. In: Claramunt C, Levashkin S, Bertolotto M (eds) GeoSpatial semantics. 4th international conference, GeoS 2011, Brest, France, May 2011, Proc., Springer, pp 88–107

Goralski R, Ray C, Gold C (2011) Applications and benefits for the development of cartographic 3D visualization systems in support of maritime safety. Int J Mar Navig Saf Sea Transp 5(4):423–431

Huang X, Alleva F, Hon H-W, Hwang M-Y, Rosenfeld R (1992) The SPHINX-II speech recognition system: an overview. computer, speech and language, vol 7, pp. 137–148

Jurafsky D, Martin JH (2008) Speech and language processing: an introduction to natural language processing, speech recognition, and computational linguistics, 2nd edn. Prentice-Hall, NJ

Katarzynska B (2009) Towards standardized maritime language for communication at sea. Int J Mar Navig Saf Sea Transp 3(3):339–341

Malyankar RM (1999a) Creating a navigation ontology, workshop on ontology management, AAAI-99, Orlando, FL In: Tech. Rep. WS-99-13, AAAI, Menlo Park, CA

Malyankar RM (1999b) An intelligent information system for maritime navigation. Symposium on question-answering systems, AAAI fall symposium series

Malyankar RM (2002) Vocabulary development for markup languages—a case study with maritime information. Eleventh international world wide web conference, Honolulu

Nikitakos N, Fikaris G (2009) Marine navigation using expert system. Int J Mar Navig Saf Sea Transp 3(3):251–259

Pietrzykowski Z, Uriasz J (2010) Knowledge representation in a ship's navigational decision support system. Int J Mar Navig Saf Sea Transp 4(3):265–270

Ray C, Goralski R, Claramunt C, Gold C (2011) Real-time 3D monitoring of marine navigation. In: Popovich V, Claramunt C, Devogele T, Schrenk M, Korolenko (eds) Information fusion and geographic information systems. Towards the digital ocean. Springer, New York

Schröder M, Trouvain J (2003) The german text-to-speech synthesis system MARY: a tool for research, development and teaching. Int J Speech Technol 6:365–377

Sorokin R (2011) Advantages of intelligent geographic information system research prototype. In: Popovich V, Claramunt C, Devogele T, Schrenk M, Korolenko (eds) Information fusion and geographic information systems. Towards the digital ocean. Springer, New York

Modeling of Noise and Acoustic Field Calculations in the Limited Water Area of Beibu Gulf Using Geospatial Data

Viktor Ermolaev and Truong Giang Bui

Abstract The noise situation and acoustic conditions of a water area have a significant effect on the efficiency of sonar's functioning. Noise situation analysis and modeling are require large amounts of geospatial data processing. The task becomes complicated if the forecast for the noise situation is made for non discrete functioning sonars in the specified water area during a long period of time. This challenge can be address using the interaction of geoinformation systems (GIS), databases, and calculating models. This chapter presents approaches to noise situation modeling using a hydroacoustic calculation system based on GIS, as well as examples of noise situation modeling and acoustic field calculations for the specified water area of Beibu Gulf.

Keywords Beibu Gulf · Sonar · Acoustic noise · Acoustic field · Geospatial data · Geoinformation system

1 Introduction

The Beibu Gulf region, which is located in the South China Sea, plays an important role in political and economic activities in Vietnam. Geographically, the South China Sea is a semi-enclosed sea off the coast of South east Asia, located between the Indochina peninsula, the islands of Kalimantan, Palawan, Luzon, and

V. Ermolaev (✉) · T. G. Bui
St. Petersburg Institute for Informatics and Automation of RAS, 39, 14 Liniya, St. Petersburg 199178, Russia
e-mail: ermolaev@oogis.ru

T. G. Bui
Saint Petersburg State Electrotechnical University "LETI", Professor Popov Street 5, St. Petersburg, Russia
e-mail: truonggiang2008@mail.ru

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_18,

Taiwan, and is a part of the marine basins of the Australasian Mediterranean Sea, between the Pacific and Indian oceans.[1]

Economically, the South China Sea plays an important role for Vietnam because it is an important transportation artery and a source of biological and energy resources. The South China Sea has big reserves of fuel, minerals, metals, and construction materials. This sea is considered to be one of the world's largest pools containing oil and gas.[2, 3]

The important geographical position and economic role of the South China Sea has lead to differences between the activities of the coastal populous states, particularly concerning the boundaries of exclusive economic zones and areas of developed biological and energy resources. This, in turn, created an urgency to protect economically important marine objects in Vietnam, which are located in territorial waters and exclusive economic zone, from capture or damage. Protection of these objects involves organizing observations in the underwater environment, detecting dangerous objects in the distance and allowing researchers to take the appropriate measures. Because the area of Vietnam's territorial waters and exclusive economic zone is rather big, a complete solution to this problem is not possible at the present time. One way to address the problem is to place separately coordinate-fixed sonar into regions with the greatest concentrations of economically important marine objects, such as the Beibu Gulf region (Fig. 1).

Here, the possible region for sonar installation has a complicated configuration and covers a large area. Placement and effective functioning of sonar in this region depends on knowledge of characteristics of the environment that influencing sonar's capabilities. For this purpose, an environment model should be formed, the components of which are:

- Hydroacoustic condition modeling of the area;
- Geoacoustic characteristic modeling of the sea floor;
- Sea-floor relief modeling;
- Sea-surface modeling;
- Noise modeling.

The models listed here are sources of basic data for the acoustic field model, characterizing the distribution of acoustic signals with set parameters in specific conditions of the area.

An effective tool for environmental and acoustic field modeling are sonar calculation systems, which integrate database and calculation models into the

[1] South China Sea (in Russian). Internet directory: Wildlife (http://www.infonature.ru/geo/water0026.html).

[2] Tran truong thuan. The territorial conflicts in the South China Sea (in Russian) (http://vnsea.net/tabid/149/ArticleID/489/language/en-US/Default.aspx).

[3] The convention of the United Nations on a marine law. Article 57 ≪ Width of an exclusive economic zone ≫ (in Russian) (http://www.un.org/ru/documents/decl_conv/conventions/lawsea.shtml).

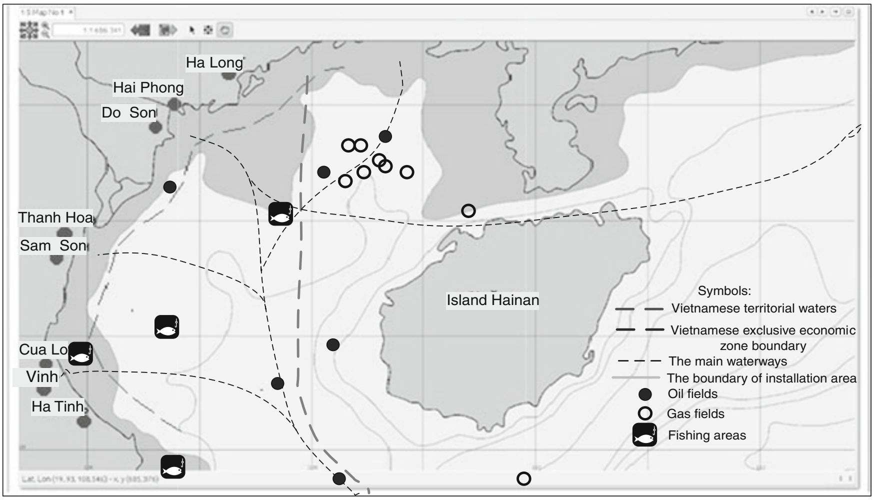


Fig. 1 The possible region for sonar installation

structure of the geographic information system (GIS) (Popovich et al. 2012a, b).[4] In this case, the backbone element is the GIS system, which provides:

- The formation of the initial data modeling;
- Binding to geographical coordinates of sonar source and receiver;
- Access to geospatial databases of environmental parameters;
- Visual representation of geospatial data;
- Geospatial data integration for the acoustic field calculation;
- Management and control of the modeling process;
- Output of simulation results in a user-friendly form on the map base;
- Storage of the modeling and reporting results.

To model the environment and the acoustic field, the hydroacoustic calculation system is used; its main user interface is shown in Fig. 2.

In this chapter, the capabilities of a hydroacoustic calculation system for modeling environment and acoustic fields in the Beibu Gulf region are shown.

[4] System of hydroacoustic calculations. Research laboratory of object-oriented geo-information systems, St. Petersburg Institute for Informatics and Automation of RAS (http://oogis.ru/content/view/24/42/lang,ru/).

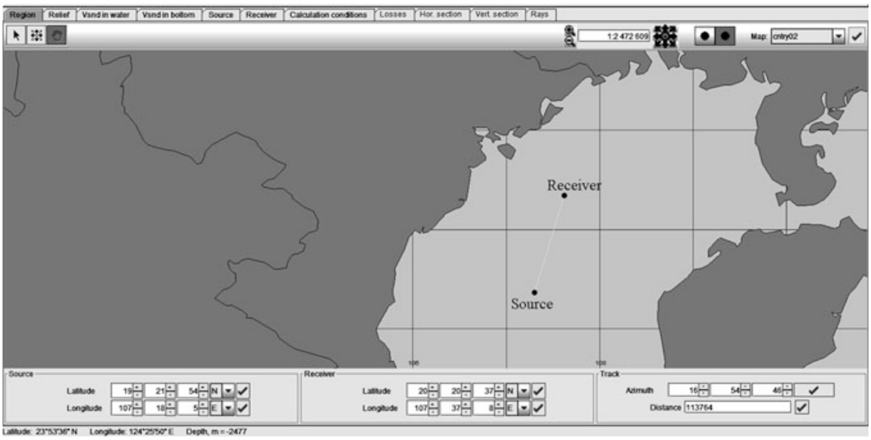


Fig. 2 Hydroacoustic calculation system

2 Modeling the Sea-Floor Relief and Its Geoacoustic Characteristics

A model of the sea-floor in this region was created using the ETOPO-1 database connected to the GIS. In the process of modeling, the GIS allows us to construct vertical sections of the water column (signal propagation path) in the directions crossing through the area's center with resolution of 5°. Examples of these sections at the parallel 19°00′ N and meridian 107°30′ E are shown in Fig. 3 (sea-floor relief in the north–south direction) and Fig. 4 (sea-floor relief in the west–east direction).

Analyzing data on the sea-floor relief suggests that the considered water area satisfies the conditions of shallow sea (of which the acoustic dimensionless parameter $kh \leq 10$, where k is the horizontal wave number and h is the depth of the considered place). Actually, geographic depths in this area do not exceed 200 m (Evtyutov and Mitko 1988). The minimum depth of the region is 25 m, whereas the maximum depth is 86 m. The sea-floor relief is rather smooth, with average depth differences of 0.5–1.5 m/km.

Distribution of sediment in the South China Sea as a whole corresponds to the general regularities of precipitate distribution in the world ocean. When moving away from the coast and increasing the depth, a reduction of sediment graininess is observed. Near the islands and rocky shore areas, the soil component is rock and sometimes shell. On the banks and in the island areas, corals and coral sands often meet. In this regard, this area can be attributed to the second geoacoustic model of shallow water (Zaraysky 2003; Hamilton 1977), characterized by the sound speed in water on the water—sea-floor interface equal to 1,850 m/s in the sand layer with the thickness of 10–15 m, and the sound speed more than 3,000 m/s in the layer of rocky grounds.

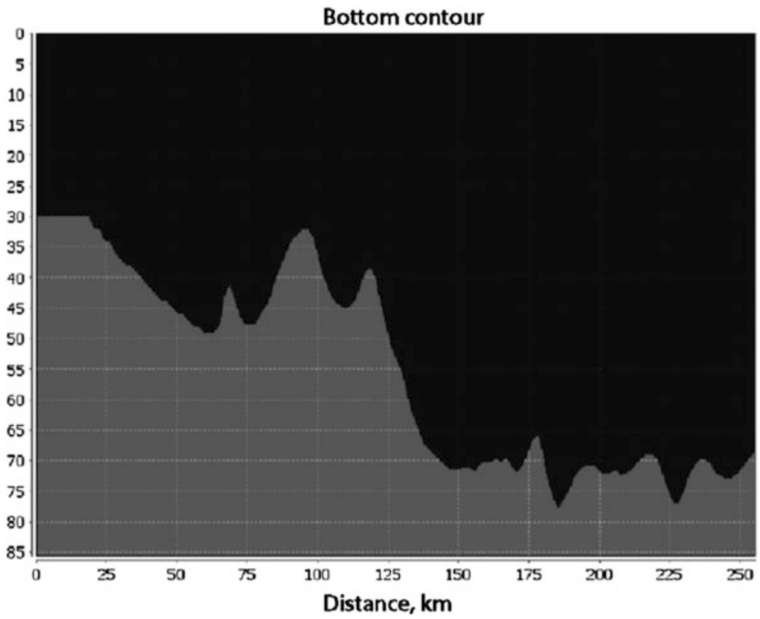


Fig. 3 Sea-floor relief in the north–south direction

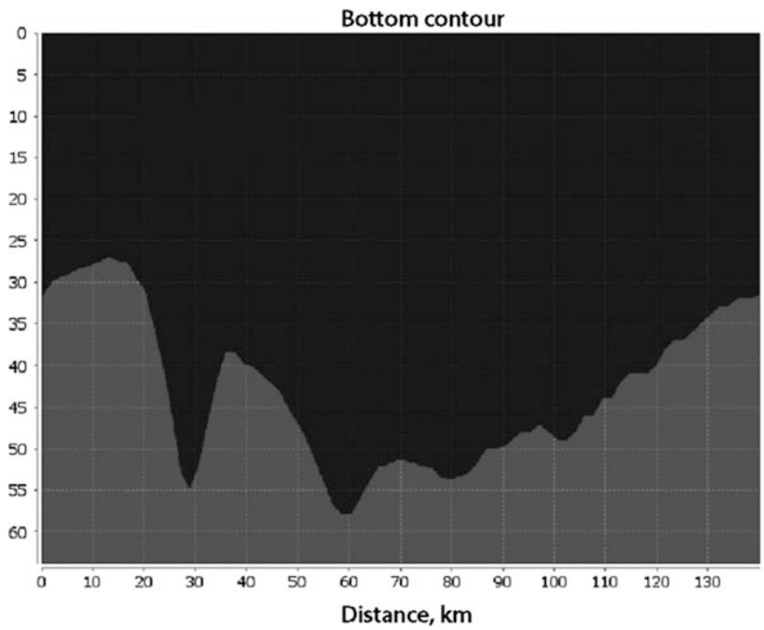


Fig. 4 Sea-floor relief in the west–east direction

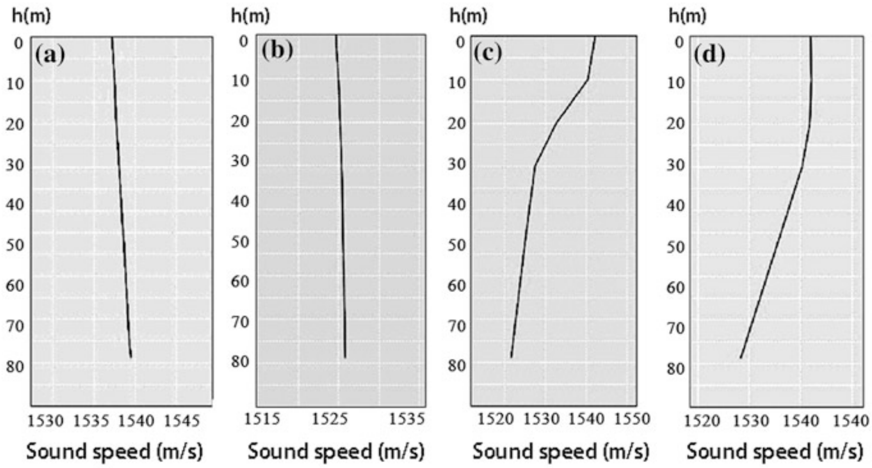


Fig. 5 Expected seasonal distributions of sound speed: (**a**) winter, (**b**) spring, (**c**) summer, (**d**) autumn.

3 Hydroacoustic Condition Modeling

The model of hydroacoustic conditions in this area was created by using the GDEM database integrated into the hydroacoustic calculation system.

The hydrological mode of the observation area develops under the influence of climatic conditions, monsoon circulation of the atmosphere, and water exchange with the adjacent seas of the Pacific Ocean.

The area is located in the equatorial and tropical zones. Air temperature at the sea surface can change from 28°C (June–September) to 8°–12°C (January–February). Depending on the air temperature change, water surface temperature drop within a year can be up to 14°C. The waters of the Beibu Gulf region are characterized by high salinity of 32–34% with small annual fluctuations.[5]

GIS capabilities allow one to obtain data on hydrological-acoustic condition in different points of the water area, as well as to construct the average curves characterizing vertical distribution of sound speed depending on season on this base (Fig. 5).

Analysis of data received from the hydroacoustic calculations system leads to the following conclusions:

- Hydrological-acoustic conditions in the area are characterized by the existence of underwater near-surface channels (UNSC), the absence of underwater acoustic channels, the far zones of acoustic insonification, and comparatively good sea-floor reflecting characteristics;

[5] The atlas on oceanography of the South China Sea//Oceanography and a status of the marine environment of the Far East region of Russia (in Russian) (http://pacificinfo.ru/data/cdrom/9/text.html#wind).

- In winter and spring, positive refraction dominates in this area. It defines a high probability (up to 94–96%) of UNSC generating. In the warm season (April–October), the probability of UNSC can reach 50%;
- The phenomenon of negative refraction at the surface layer during the cold period of year seldom occurs: repeatability in November–March can reach up to 4%; during the warm period of year, the probability of the negative refraction increases to 48–49% in April–October.

4 Noise Modeling

The environmental noise model in the selected area is formed by using algorithms, which are integrated into the hydroacoustic calculation system and provide power spectral density calculations for each noise source in the sea, as well as their integrated values in standard conditions. Basic data for the implementation of these algorithms are created by the GIS. For this purpose, a set of cartographical layers, on which zone division of noise sources and their characteristics are produced, are formed.

The main noise sources are as follows (Zaraysky 2003; Furduyev 1974; Stashkevich 1966; Dugan 1982):

- Shipping traffic noise;
- Dynamic noise of the sea;
- Seismic noise;
- Thermal noise.

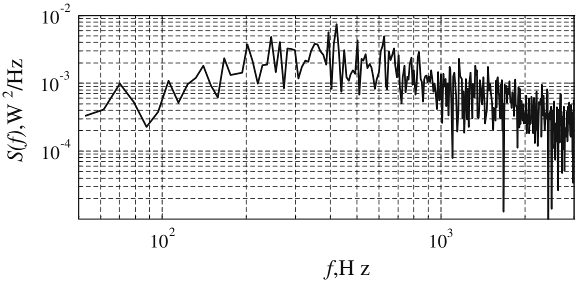
The basic data created by the GIS are used for noise modeling in spectral and time domains. In Figs. 6 and 7, the results of dynamic noise modeling are shown at the set coordinates.

In Fig. 8, spectral characteristics of total noise are given in the considered region.

The noise spectral model in this sea leads to following conclusions:

- The main sources of sonar interference in the Beibu Gulf region are dynamic noise caused by wind waves and shipping traffic noise (shown in Fig. 8);

Fig. 6 Spectral results of dynamic noise modeling in the set point of the region



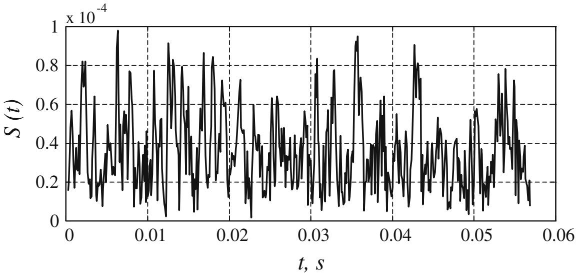


Fig. 7 Results of dynamic noise modeling in the time domain in the set point of the region

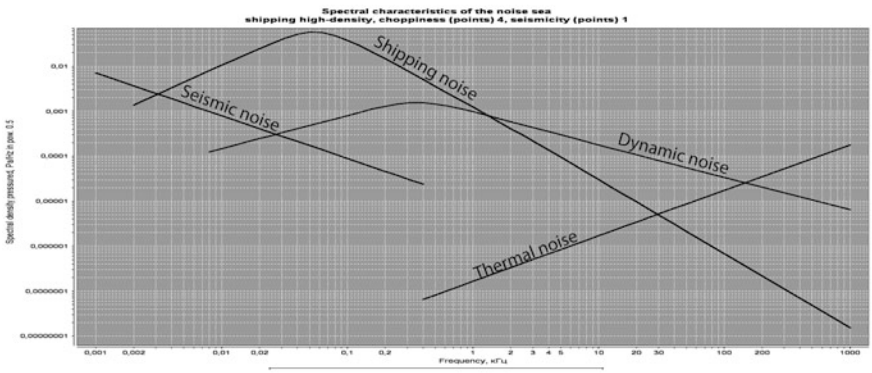


Fig. 8 Spectral characteristics of noise

- Seasonal changes in dynamic noise are determined by the variation of wind waves and noise of heavy rains occuring under the influence of monsoon and tropical cyclones (Zaraysky 2003);
- Distant shipping noise has an effect on the characteristics of total noise. In addition, a ship's movement on shipping routes located in close proximity to area of possible placement of sonar equipment protecting maritime economic activities will be a source of local noise;
- Noise of biological origin has local spatial and temporal characteristics. For this reason, they should be taken into account only in the dynamics of the direct operation of the sonar equipment. In the process of modeling, this type of noise is not taken into account;
- Other sources of noise (thermal noise, seismic noise) have little effect on the whole picture of noise in this frequency range. Consequently, they can be neglected.

Simulation showed that the noise field on an acoustic antenna input of sonar equipment in this water area represents an additive compound of independent components: dynamic noise of the sea and distant shipping noise. In addition, it is necessary to expect the existence of local noise in the survey zone of the sonar

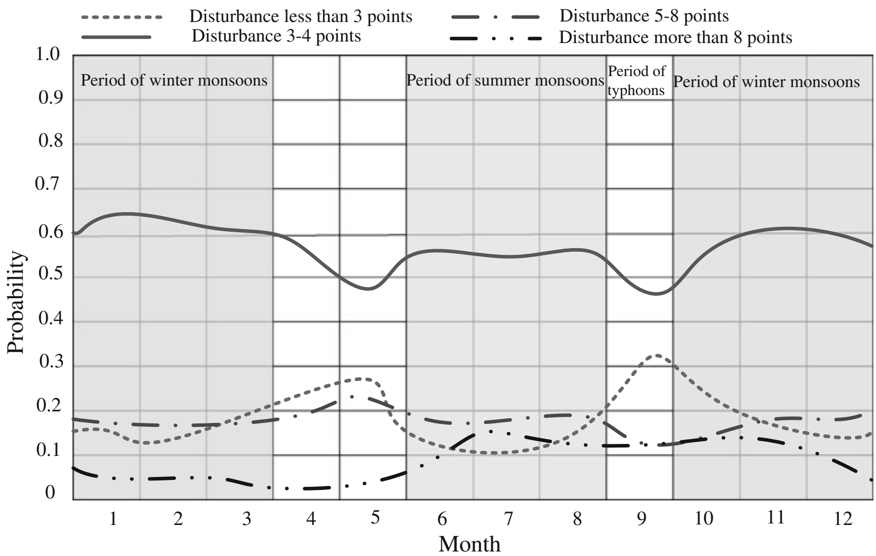


Fig. 9 Seasonal changes of sea disturbance

equipment. Its sources are ships passing on shipping routes, and also noise of biological origin.

The availability of instruments enabling evaluation of noise components and their integral characteristics in hydroacoustic calculation systems allows us to construct the model, reflecting the dynamics of noise change within a year.

In Fig. 9, the time-dependent diagram of probability values of sea disturbance is shown.The estimates showed that noise dispersion is within 0.9×10^{-6} to 1.3×10^{-6} Pa, depending on season (month). In calculations, the probability of rainstorm occurence in period of typhoons is taken into account. In the simulation, the values of distant shipping noise within 3×10^{-4} to 8×10^{-4} Pa are taken into account, not depending on season.

5 Acoustic Field Modeling

Modeling of acoustic fields in the area is performed with the use of two mathematical models integrated into the hydroacoustic calculation system: wave model and ray model. The basic data of these models are environmental parameters described above, as well as receiver parameters.

The basis of the wave model is the solution of the pseudodifferential parabolic equation (Avilov 1992, 1995). The ray approximation model provides the calculation of field parameters by presenting the propagation of acoustic energy in the water in the form of rays (Evtyutov and Mitko 1988; Zaraysky 2003; Stashkevich 1966).

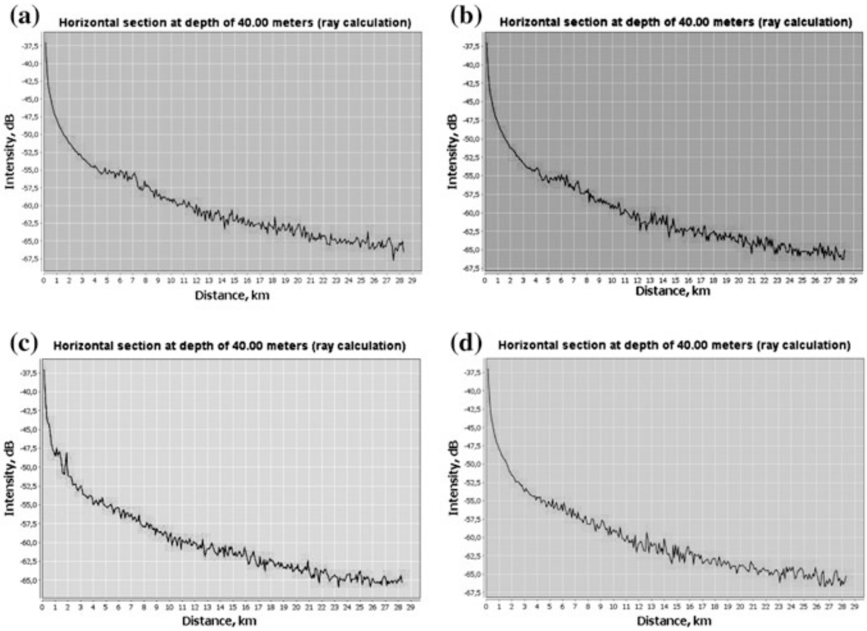


Fig. 10 Horizontal section of acoustic field at the depth of 40 m: (**a**) winter, (**b**) spring, (**c**) summer, (**d**) autumn.

For acoustic field modeling in the area, a point source set to a depth of 27 m and working at a frequency of 1 kHz was chosen. In Fig. 10, the section of acoustic field on the depth of 20 m on the route for various seasons is given. The calculation is performed with the use of the ray approximation method.

Analysis of acoustic field modeling results shows that the decline of sound energy with distance is about 2 dB/km in winter and spring, increasing to 3 dB/km in summer and autumn. Propagation losses are quite great in summer and autumn; in these seasons, the phenomenon of negative refraction occurs and increases energy loss of rays when these rays re-scatter from the bottom.

The results obtained in acoustic field modeling indicate that the effective distance of sonar equipment protecting maritime economic activities in the Beibu Gulf area will greatly vary based on the season. The ratio of the maximum and minimum effective distance can reach values of 1.35–1.93.

6 Conclusion

Noise modeling and acoustic field calculations in the water area of the Beibu Gulf showed that the hydroacoustic calculation system is an effective tool for analyzing the environment in different regions of the world ocean. Such systems are combined in a uniform structure of GIS, database, mathematical models, and display systems of modeling results. These systems can provide access and operation with geospatial data and expand their uses in design, verification of placement optimal coordinates, efficiency assessment, and definition of sonar system operating procedures.

References

Avilov CV (1992) An effective numerical solution of guided wave equations. J de Physique IV, Collock C1, Supplement au J de Physique III 3:C1–1023

Avilov CV (1995) The pseudodifferential parabolic equations of sound propagation in the ocean, smoothly inhomogeneous on horizonts and their numerical solution (in Russian). Publishers "Акустический журнал"

Dugan JP (1982) Oceanology in underwater acoustics (in Russian). In book: Acoustics of the ocean. Publishers "Наука"

Evtyutov AP, Mitko BP (1988) Engineering calculations in hydroacoustics (in Russian). Publishers "Судостроение"

Furduyev AV (1974) Ocean noise (in Russian). In book: Acoustics of the ocean. Publishers "Наука"

Hamilton EL (1977) Geoacoustic model of the seabed//Acoustics of marine sediments (in Russian). Publishers "Мир"

Popovich V, Aviloff C, Ermolaev V, Zinyakov Y, Leontev Y, Korolenko K, Popov O (2012a) Sound propagation modeling on intelligent GIS basic. St. Petersburg Institute for Informatics and Automation of RAS. Institute for Systems Analysis of RAS

Popovich V, Ermolaev V, Leontev Y (2012b) System of hydroacoustic calculations on the basis of intellectual geoinformation system (in Russian). St. Petersburg Institute for Informatics and Automation of RAS. Institute for Systems Analysis of RAS

Stashkevich AP (1966) Acoustics of the sea (in Russian). Publishers "Судостроение"

Zaraysky VA (2003) Acoustics of the ocean (in Russian). Publishers Naval academy

Anomaly Detection for the Security of Cargo Shipments

Muriel Pellissier, Evangelos Kotsakis, and Hervé Martin

Abstract Detecting anomalies in the maritime domain is nowadays essential as the number of goods transported by maritime containers keeps increasing. An anomaly can be described in several ways depending on the application domain. For cargo shipments, an anomaly can be defined as an unexpected relationship between ports. This chapter describes a new approach for detecting anomalies in the sequential data used to describe cargo shipments. The technique is divided in two steps. First, we find the normal itineraries with a regular expression technique. Then, we compare a given itinerary with a normal itinerary using a distance-based method in order to classify the given itinerary as normal or suspicious. The first results of this method are very promising, and it can be further improved when integrated with time-based information. This chapter presents both the methodology and some results obtained using real-world data representing container movements.

Keywords Anomaly detection · Maritime security · Sequence · Regular expression · Distance

---

M. Pellissier (✉) · E. Kotsakis
European Commission, Joint research Centre, Institute for the Protection and Security of the Citizen, Ispra, Italy
e-mail: pellissier.muriel@gmail.com

E. Kotsakis
e-mail: evangelos.kotsakis@jrc.ec.europa.eu

H. Martin
Laboratory of Informatics, University Joseph Fourier, Grenoble, France
e-mail: herve.martin@imag.fr

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_19,

1 Introduction

Today, most goods are transported by maritime containers all over the world, and the number of containers travelling every day is really significant. For example, around 15 million containers arrive into ports every year only in the US. It is necessary to control the container movements in order to detect illegal activities such as fraud, quota-related, illegal products, hidden activities, or drugs and arms smuggling. But only 1–2 % of the containers can be physically checked by customs agents because the cost and the human resources needed to check a container are quite significant. As the number of containers travelling every day in the world increases, it is necessary to improve the targeting in a way that reduces as much as possible the need for a physical check of random containers. Performing physical checks only on those containers that seems to be suspicious will improve both targeting and efficiency. Therefore, some computational tools are needed to help the customs agents to focus their attention on suspicious containers.

In this study, suspicious containers are defined as containers behaving as close as possible as normal containers in order not to attract the attention of customs agents. In order to detect such anomalies, we will use the container itineraries. An itinerary represents all the events of a container from its departure (loading) to its arrival (discharge), including the transshipments. The proposed method does not require the geographical information of a container at every moment during its trip. It is sufficient to know when a container is in a port, the name of the port, and what action is made on the container at that specific port. The different actions are loading, discharging, and transshipment. In Fig. 1, we provide an example of an itinerary with one transshipment port: the container leaves from Singapore, goes to Chiwan, and ends its trip in Rotterdam.

An itinerary is a sequence of ports: the first port corresponds to the departure port and the last port is the arrival port. If there are more than two ports in an itinerary, the ports between the departure port and the arrival port are called transshipment ports.

With this technique, we are able to analyze the relationships between the ports. We are interested in rare relationships between ports. We define the anomalies as unexpected relationships between ports where the relationships are close but do not match exactly the common relationships. Our technique is divided in two parts. The first part is to detect the common itineraries. Common itineraries are itineraries, or part of itineraries, that are frequently used in the transport of the

Fig. 1 Itinerary from Singapore to Rotterdam passing through Chiwan

containers. We detect the common itineraries using regular expressions. A regular expression allows us to find frequent patterns using wildcards. The second part is to calculate an anomaly degree between two itineraries. To calculate the anomaly degree, we measure the distance between a given itinerary and a common one. The given itinerary is classified as normal or abnormal (suspicious) depending on its anomaly degree value.

In the next section of this chapter, we will describe existing anomaly detection techniques. In Sect. 3, we explain the anomaly detection technique we have developed. Sect. 4 presents some testing of this anomaly detection technique and their corresponding results. We then conclude the chapter by proposing some directions for future research.

2 Related Work

This research is linked to the detection of anomalies in sequences. We will describe in this section some anomaly detection techniques for sequences and also some techniques applied to the maritime domain.

Chandola et al. (2012) presents a survey on an anomaly detection technique for sequences. The chapter provides a summary of the different techniques that currently exist for detecting anomalies in sequence data. Even though most of the anomaly detection techniques share the same purpose, the techniques and the data can be really different depending on the domain and/or on the application. For example, some well-known applications of anomaly detection of sequences are the system calls that can detect, for example, an attack on a user session or the biology domain in order to detect anomalies in DNA sequences. This survey is the only one that is not oriented on any application domain. Sequences are defined as series of ordered events. The events are data that can only take certain values. A subsequence is a part of a sequence.

The anomaly detection research can be divided in three different categories. The first group represents the *sequence-based anomaly detection techniques*: the abnormal sequences are identified thanks to some training data (normal sequences). Four techniques are part of this group. The *similarity-based techniques* deal with the similarities between a given sequence and a normal sequence in order to obtain an anomaly score. The *window-based techniques* use small parts of the whole sequences in order to detect the anomalies. The *Markovian techniques* use probabilistic models in order to evaluate the probability of a symbol knowing the previous symbols. The *Hidden Markov model-based techniques* add wildcards into the sequence in order to detect the anomalies.

The second group represents *contiguous subsequence-based anomaly detection techniques*: the anomalies are subsequences of longer sequences. These techniques use windows of sequences (subsequences). Each window is compared with all the windows obtained from the sequences using a distance technique. A subsequence is considered as an anomaly depending on the distance value.

The third group represents *pattern frequency-based anomaly detection techniques*: These techniques compare the expected frequency of a pattern, which is estimated from the training data set, with the frequency of the same pattern in the data set. When the expected frequency and the actual frequency are too different, the pattern is classified as abnormal.

Eberle et al. (2012) describes some anomaly detection techniques applied to the specific domain of the cargo security. Several directions can be taken to detect suspicious containers. Some studies use images of the cargo (Cardoso 2008; Swaney and Gianoulis 2008). The anomalies are identified by processing the images. Agovic et al. (2009) proposes a technique using the weight of the containers to detect anomalies. Ling et al. (2009) presents a technique to track and monitor automatically the movement of the barges in real-time in order to detect suspicious movements. Other techniques detect spatial anomalies comparing the distance between spatial information (Sun and Chawla 2004; Lu et al. 2008; and Kou et al. 2006). Eberle et al. (2012) describes a graph-based anomaly detection technique. This technique uses an unsupervised approach and does not need to have training data. They detect the anomalies by analyzing the relationships between the data using a graph-based representation. A graph represents a cargo and all its information (contents, departure port, arrival port, transshipment ports, shipment company information, vessel number etc.). By analyzing the graphs, it is possible to detect the normal behaviors and the deviations of those normal behaviors. Eberle and Holder (2007) define three types of graph-based anomalies: modifications, insertions, and deletions. This technique detects the best substructure(s) of the graphs (Cook and Holder 2000) using a minimum description length (MDL) heuristic (Rissanen 1989). The detection of anomalies is performed utilizing three different algorithms (one algorithm for each type of anomaly) by finding the patterns that are close to the best substructure(s) (Eberle and Holder 2007).

All these sequence-mining approaches focus on detecting anomalies in sequences. For the sequence-mining approaches, the order of the elements is important. For our application, the order is important but the absolute position of the elements in the sequence is equally important.

Those techniques that focus on the maritime domain require much more information for targeting anomalies, such as geographical information, pictures of the container, weight of the container, content of the container, or information about the shipment company. This information is not always available and this may cause certain problems when applying such techniques. The advantage of using our approach is that it requires much less information about the shipments and it can be easily applied.

In the next section, we will explain the different steps of the anomaly detection technique that we propose.

3 Anomaly Detection Technique

The proposed anomaly detection technique uses an unsupervised approach. Therefore, this approach does not need to have training data to define the normal data. The user does not need to have any knowledge on the data to use this technique. This technique is used to find anomalies in structured data. We apply this technique to the maritime domain, but it can also be used in other domains as well. For our application, maritime surveillance, we are interested in finding hidden information. Hidden information could be missing information, incomplete information, or information that has been well hidden intentionally or unintentionally. We define these anomalies as unexpected relationships between ports where the relationships are close but not exactly identical to normal relationships. In other words, we are looking for sequences of ports that are close to the most frequent ones but with some differences. A sequence of ports is called an itinerary.

Let $E = \{e_1, e_2, \ldots, e_m\}$ be a set of m distinct items. In this application, the items are ports. Let $S = \{S_1, S_2, \ldots, S_n\}$ be a set of n sequences of m items with m random variable changing from one sequence to another one. A subsequence of the sequence S_y is $S'_y = \{e_a, {}^*, e_b, {}^*\}$ with * unknown items (wildcard). A subsequence is a part of an itinerary.

This anomaly detection technique is divided into two steps:

1. We detect sequences or subsequences that are common in our data set. We use a regular expression technique in order to detect the common itineraries using all the data (normal and abnormal).
2. We compare a given sequence with the common sequences/subsequences and we calculate an anomaly degree with a distance technique. Using the anomaly degree, we can classify the given sequence as normal or abnormal.

With this technique, we can identify the anomalies and present their details to the user. The user can have some controls on the results with several thresholds in order to better adapt the results to his needs. The first threshold is to select the common itineraries. With a high threshold, the user can select only the main itineraries; with a lower threshold, the user can select more common itineraries that are less frequent than the main ones. The second threshold is the maximum anomaly degree threshold, which is the level defining an itinerary as normal or abnormal. Moreover, the first part of this technique could also be used alone because it can detect common routes or a frequent couple of ports. This information can be useful for maritime surveillance.

3.1 Common Sequences/Subsequences

To detect the common sequences (itineraries), we need to compare all the sequences of the data set and to calculate their frequency. We are also interested in finding common subsequences. A subsequence is a part of a sequence and not the whole sequence. We use wildcard in order to define the subsequences. The wildcard can substitute any port.

The frequency is directly obtained by counting how many times each sequence appears in the data set. The frequency of a subsequence is more difficult to calculate. In the following section, we will discuss how to calculate the frequency of a subsequence. As we do not have any knowledge about the data set, we do not know the interesting subsequences. We need to create all the possible subsequences and to calculate how many times they appear in the data set. In order to create the subsequences, we need to add wildcards to the sequences. If an itinerary has only two ports, we will not create a subsequence. The subsequence of an itinerary of two ports will be composed by only a port and a wildcard. We are not interested in a single port but only in relationships between ports. For every itinerary of more than two ports, we will create a subsequence of that itinerary by replacing only one port by a wildcard. For one itinerary, we can create several subsequences by replacing every port by a wildcard at a time. The number of subsequences created depends on how many ports an itinerary has. For example, we can create four subsequences with an itinerary that has four ports. Figure 2 shows an example of an itinerary from port $X1$ to port $X3$ passing through $X2$ and all the possible subsequences of this itinerary. In the subsequences, the character "*" is a wildcard and it can correspond to any port. As we see in Fig. 2, with this itinerary we can create three different subsequences.

The regular expression is a powerful method to identify strings in a text. It is used to detect specific characters, words, and patterns within a text. With regular expressions, it is possible to detect easily a specific sentence in a text. It is also easy to find sentences that are not exactly identical using wildcards. For example, it is possible to detect all the words of a sentence, to create a new sentence by replacing one of the words with a wildcard, and to detect all the sentences in a text that are similar to the new sentence.

A sequence can be seen as an ordered sentence where the positions of the words have an importance. A group of sequences can be seen as a text composed by many sentences. For our application, an itinerary is considered as a sequence that can be a sentence for the regular expression. The data set is a list of itineraries that can be seen as a text (list of sequences) for the regular expression.

We need to create for every itinerary of our data set all the possible subsequences as we have seen in Fig. 2. Then, we need to calculate the frequency of all the subsequences. The algorithm is shown in Fig. 3.

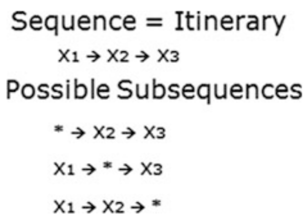


Fig. 2 An itinerary and all its possible subsequences

Fig. 3 Common subsequences algorithm

```
Algorithm Common_subsequences
    open dataFile
    for each line in dataFile do
        lineNo ← lineNo + 1
        arrayOfWords ← split(' ',line)
            for each item in arrayOfWords do
                subLine ← line.replace(item, '.+?')
                frequency(line, lineNo)
    write outputData in frequencyDataFile
```

```
Algorithm frequency(line, lineNo)
    open dataFile
    frequency ← 0, subLineNo ← 0
    for each subLine in dataFile do
        subLineNo ← subLineNo + 1
        if line = subLine and lineNo <= subLineNo then
            frequency ← frequency + 1
        outputData ← frequency
        outputData ← subLine
    return outputData
```

1. We read every line of the data set file. Every line of the file is an itinerary.
2. We detect all the ports (words) of a line. All the ports are separated to each other by a space character (' ').
3. We create subsequences by replacing every time a different port with a wildcard. With regular expression, the combination of characters ". + ?" is used for a wildcard.
4. We count how often the subsequences are present in the data set. With a minimum frequency threshold, we can select which lines (itineraries) are frequent enough in the data set to be considered as common itineraries.

In our database, each port is represented by a code. In a location database, we have for each port all the details of the location: the name of the country, the name of the city, and the geographic coordinates. The codes used for the ports are inconsecutive numbers and the length of the port names is not standardized. As there is no important difference for the algorithm, we have decided not to change the port names because it is easier for the user to have a link between the data records in the different databases.

A wildcard "*" replaces a port in an itinerary and it can match any port. With regular expression, the wildcard is the symbol ". + ?". The symbol "." represents every character except the new-line character. As every line is a different itinerary, it is important to keep the new-line character to know where an itinerary ends. The "+" is used to match one or more characters. But because all the characters except the new-line character will match ". +", we need to limit it. If we do not limit it, all the characters until the new-line character will always be a match. In order to control the comparison, we use the symbol "?", which makes the operation lazy. This means that every time we compare two characters, the character after ". + ?" is also checked. As soon as the character after ". + ?" is found in the other line, the characters following will not match ". + ?" anymore.

With this first part, we are able to detect the common itineraries; we need the second part to detect the suspicious itineraries.

3.2 Anomaly Degree

The purpose of our technique is to detect anomalous itineraries and to spot the anomalies. We identify the anomalies by comparing a given itinerary with the common ones. We calculate an anomaly degree between an itinerary and a common one. Depending on the value of the anomaly degree, we will classify the given itinerary as normal or abnormal. The anomaly degree is calculated with a distance technique.

Let S_{data} be a given sequence of the data set S. S_{nor} is a common sequence or subsequence discovered with the regular expression technique as described previously. Dist(S_{data}, S_{nor}) is the distance between two sequences: a given sequence S_{data} and a common one S_{nor}. For example, in Fig. 4, we have two itineraries. One itinerary is a given itinerary from $X1$ to $X4$ passing through $X2$ and $X3$. The second itinerary is a common itinerary $X1$ to $X4$ with a wildcard for the transshipment port.

The dist(S_{data}, S_{nor}) is the number of actual changes divided by the maximum number of possible changes between two sequences. We calculate the distance using two factors: the structure of the itinerary and the port names. The *structure* corresponds to the number of events and their type. The *port names* are the actual

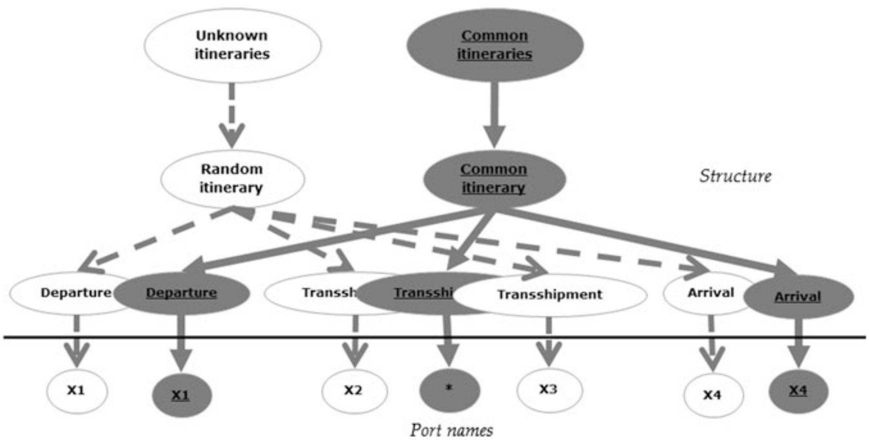


Fig. 4 Two itineraries: a given itinerary ($X1$, $X2$, $X3$, $X4$) and a common itinerary ($X1$, *, $X4$)

ports visited during an itinerary. As shown in Fig. 4, the structure is the part above the black horizontal line and the port names are the part below the black horizontal line.

To calculate the *structure* distance, we use the events of the itineraries. The events are the departure, the arrival, and the transshipment(s). We calculate the absolute difference between the number of events of the sequence S_{data} and the number of events of the sequence S_{nor}:

$$|S_{\text{data}}.\text{length} - S_{\text{nor}}.\text{length}|$$

The maximum possible changes between two structures are the maximum number of events between S_{data} and S_{nor}:

$$\max(S_{\text{data}}.\text{length} \vee S_{\text{nor}}.\text{length})$$

We also need to take into account the *port names* and the event linked with the port. There are three possibilities: (1) A port is identical in both sequences S_{data} and S_{nor}. It means that the name of the port and the event related to it are similar in both itineraries. For example, in Fig. 4, $X1$ is for both itineraries the port of departure. (2) A port of S_{data} is also present in S_{nor} but the port has not the same event in both itineraries—for example, if in S_{data} port $X5$ is the port of departure and in S_{nor} port $X5$ is the port of transshipment. (3) A port of S_{data} is not present in S_{nor} or the opposite.

The event related to the port is linked in our data set to the position of the port in the sequence. The first port is the port of departure, the last port is the port of arrival, and the ports between are the ports of transshipment.

Therefore, we calculate the difference between two sequences as:

1. If a port is in both itineraries at the same position, we attribute the value 0 to the port. But as two itineraries might not have the same number of ports, it is not straightforward to compare the position of two ports. In order to compare the position of a port, we compare the two itineraries from the beginning and starting from the end of the itineraries. If the position is similar in one of the two comparisons, the two ports will be considered as having the same position and we will attribute the value 0 to the ports. For example, consider the common itinerary $X3$ $X5$ * and the given itinerary $X10$ $X3$ $X5$ $X6$. If we compare them only from the beginning, they will have a really high anomaly degree because none of the ports have the same position as in the other itinerary. But if we compare the two itineraries from the end, they appear to be quite similar.
2. If one port is present in both but with a different position, as defined previously, we attribute the value 1.
3. If a port is present in only one of the two itineraries, we attribute the value 2 to the port.
4. We need to add the case when the common itinerary contains a wildcard "*". As we have seen before, the character "*" implies that it can be any port. We will not put any weight on the port "*" and we will not consider it as the same port as any other port. Thus, we know that if an itinerary is common because of

several itineraries when we compare one of these several itineraries with the common one, the anomaly degree will not be 0. For example, we have in our data set the itineraries *a b c*, *a b d*, *a b e*, *a b f*, *a b g,* and *h j k*. The itinerary *a b* * will be a common itinerary. If we compare *a b c* with the common itinerary *a b* *, even if *a b c* is actually included in *a b* *, the anomaly degree will be 0.2222. The ports *a* and *b* take the value 0 but the port *c* takes the value 2. We will define a minimum anomaly degree equal to 0.23. The itineraries that have an anomaly degree inferior to the minimum degree will not appear as abnormal itineraries.

The maximum number of changes that we can have with the port names is the number of distinct ports between the two itineraries. We do not consider the port "*" as one of the distinct port for the maximum number of changes. As we have seen, we can attribute to a port the value 0, 1, or 2. The maximum value attributed to a port is 2, so if two itineraries are completely different, the maximum possible value will be the number of all the distinct ports multiplied by 2. Thus, the number of maximum changes is the number of distinct ports multiplied by 2.

The anomaly degree is:

$$\mathrm{dist}(S_{\mathrm{data}},\ S_{\mathrm{nor}}) = \frac{|S_{\mathrm{data}}.\mathrm{lenght} - S_{\mathrm{nor}}.\mathrm{lenght}\ | + \Sigma\,\mathrm{port\_value}(0,\ 1,\ 2)}{\max(S_{\mathrm{data}}.\mathrm{lenght} \vee S_{\mathrm{nor}}.\mathrm{lenght}) + (\mathrm{num\_of\_distinct\_port}) * 2}$$

A maximum anomaly degree threshold is defined by the user. Depending on the threshold, the itinerary S_{data} will be classified as normal or abnormal. If the anomaly degree is lower than the threshold, the itinerary will be detected as anomalous.

The algorithm is shown in Fig. 5.

1. We reduce the data file. The data file contains all the itineraries of our data set. As a consequence, the same itinerary might be present several times. In order to avoid comparing several times a common itinerary with the same itinerary, we create a new file where each itinerary is present only once. This new file is the reducedDataFile.
2. We read every line of the reducedDataFile.
3. We read every line of the frequencyDataFile that we have created previously with the common subsequences algorithm described in Fig. 3. The file contains the frequency of every sequence/subsequence.
4. A sequence/subsequence is considered to be a common itinerary if its frequency is superior to the minimum frequency threshold. If the itinerary, is a common itinerary, we compare the two lines. The first line is a given itinerary. The second line is the sequence/subsequence found with the common subsequences algorithm described in Fig. 3.

We attribute values to the ports: the value 0 if a port is in the other itinerary and at the same position or if a port is "*", the value 1 if a port is in the other itinerary but at a different position, and the value 2 if a port is not present in the other itinerary.

```
Algorithm anomaly
   reduce_data()
   open reducedDataFile
   for each reducedLine in reducedDataFile do
      open frequencyDataFile
      for each frequencyLine in frequencyDataFile do
         if frequency >= FREQ_MIN then
            reducedArray ← split(' ', reducedLine)
            frequencyArray ← split(' ',frequencyLine)
            for each item in reducedArray do
               if reducedArray[item] not exists in frequencyArray then
                  value ← value + 2
               if reducedArray[item] exists in frequencyArray &&
                reducedArray[item].index != frequencyArray[item].index then
                  value ← value + 1
            for each item in frequencyArray do
               if frequencyArray[item] != ".+?" && frequencyArray[item]  not exists in
                reducedArray then
                  value ← value + 2
            portNo ← calculate distinct number of ports between reducedArray and
             frequencyArray
            anomaly_degree ← abs(reducedArray.size –  frequencyArray.size) + value /
             max(reducedArray.size v frequencyArray.size) + portNo*2
            if MIN <= anomaly_degree <= MAX then
               anomaly ← reducedLine
               anomaly ← frequencyLine
               anomaly ← anomaly_degree
   write anomaly in anomalyFile
```

```
Algorithm reduce_data()
   open dataFile
   for each line in dataFile do
      compare line with previous lines
      if line != previous lines then
         reduceData  ← line
   write reducedData in reducedDataFile
```

Fig. 5 Anomaly algorithm

5. We calculate the anomaly degree between the two itineraries.
6. If the anomaly degree is between the minimum anomaly degree threshold and the maximum anomaly degree threshold, then we put in an array the given itinerary, the common itinerary, and the anomaly degree value.
7. We write all the anomalies found in a text file: anomaly File.

For example, if we compare the two itineraries of Fig. 4:

- Structure: actual changes: $|4-3|$ (4 events S_{data} and 3 events S_{nor}) and maximum changes: 4.
- Name ports: actual changes: 4 ($X1$ compared with $X1 = 0$, $X2$ compared with * or $X1 = 2$, $X3$ compared with * or $X4 = 2$, $X4$ compared with $X4 = 0$) and maximum changes: 4*2 (4 different ports: $X1$, $X2$, $X3$, $X4$).
- Anomaly degree: $\text{dist}(S_{\text{data}}, S_{\text{nor}}) = (1+4)/(4+4*2) = 0.41.$

If the user defines a maximum anomaly degree threshold that is higher than the distance, the itinerary will be detected as anomalous. The normal behavior is to have only one transshipment port, but the itinerary S_{data} has two transshipment ports.

4 Experiment

We collect our data on container trips from different public sources. In order to have a coherent data set, we clean the data. The cleaning is mostly linked with text string errors for geographic locations and container events. The data set contains container events information. For each event, we know the container number, the geographical localization of the container, and what action is done on that container in that specific port. Currently, the data set contains more than 900 million events and about 12 million containers. Because our data contains anomalies, first we tried this technique with some experimental data created based on the real world data. Once the results with experimental data were satisfied, we tried this technique with the real-world data.

4.1 Experimental Data

We have created some experimental data sets based on real-world data. With some tests done on our data set, we know that for a number x of itineraries, we have y number of ports with $y = x * 6\ \%$. We also know, as shown in Fig. 6, that most of the itineraries have no transshipment port.

We created experimental data with 200 itineraries. In all, 60 % of the itineraries have zero transshipment, 30 % have one transshipment, and 10 % have two transshipments. We use twelve distinct ports. With a random function, we created the itineraries. We add four different itineraries ten times in order to be considered as frequent itineraries. We add ten suspicious itineraries.

For this experimentation, the minimum frequency will be 5.40, the maximum anomaly degree will be 0.49, and the minimum anomaly degree is 0.23. The minimum frequency value is calculated with the frequencyDataFile of the common subsequences algorithm. The average frequency is calculated with the frequency of all the itineraries. Another frequency is calculated with the itineraries that have a frequency above the frequency average. The second frequency value is the value used for the minimum frequency threshold.

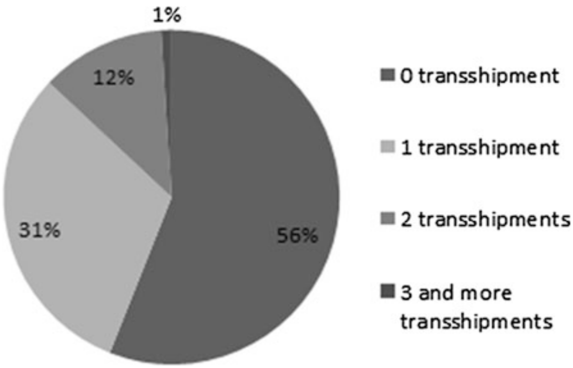


Fig. 6 Percentage of transshipments in real-world data

With this technique from a data set of 210 itineraries, we detected 42 itineraries (20 %) that could be anomalous. All the anomalous itineraries inserted are detected as anomalous; other suspicious itineraries created randomly were also detected.

The results are written in a text file, as shown in Fig. 7. Every anomalous itinerary uses three lines on the text file. The first line is the anomalous itinerary, the second line is the common itineraries, and the third line is the anomaly degree values.

As shown in Fig. 7, one anomalous itinerary detected is the itinerary going from port 1 to port 5 passing through port 3 and 4. It is considered anomalous because of three different common itineraries. The first common itinerary used to define this itinerary as abnormal is the itinerary going from any port to port 4 passing through port 3. Because the departure port of the common itinerary is a wildcard, it can be any port; we consider the port of departure to be the port 1. We can see in that case that the two itineraries are really similar but an extra port is added (the arrival port), which might be a suspicious behavior.

```
1 3 4
1 0 3
0.333333333333333
1 2 3
1 3 0
0.333333333333333
1 3 4 5
0 3 4 / 1 0 4 / 1 3 0
0.416666666666667 / 0.416666666666667 / 0.416666666666667
4 2 3
0 3 4
0.444444444444444
1 5 6
1 5
0.333333333333333
```

Fig. 7 Anomalous itineraries using experimental data

Fig. 8 Anomalous itineraries using real-world data

```
509 277 124
277 124 0 / 509 277
0.444444444444444 / 0.333333333333333
142 175 407
142 175
0.333333333333333
193 606 158
193 606
0.333333333333333
175 277 955
0 175 277
0.444444444444444
187 19 963
187 19 / 0 187 19
0.333333333333333 / 0.444444444444444
```

4.2 Real-World Data

We tested this technique with real-world data as well. The data set had 135,798 itineraries but 22,831 distinct itineraries. The minimum frequency calculated with the frequency of the itineraries as described previously is 60. As for the experimental data, the maximum anomaly degree is 0.49 and the minimum anomaly degree is 0.23.

With these thresholds, we detect 4,773 anomalous itineraries. It is 3.5 % of the whole data set and 20.9 % of the distinct itineraries. Some examples of anomalous itineraries are shown in Fig. 8.

As we can see in the Fig. 8, we detect the modification, the deletion, or the insertion of information of the common itineraries.

5 Conclusion and Future Work

This chapter presents a method for identifying suspicious itineraries of maritime containers. This technique consists of two main parts. In the first part, we detect the common sequences and the most frequent one; in the second part, we calculate the distance between a given itinerary and a common one.

We have tested this technique with experimental data based on real-world data as well as with real-world data. The results are very promising and indicate that this method is satisfactory in identifying suspicious itineraries.

This technique can be further improved by integrating customs import declarations or potentially other types of intelligence. For example, knowing which port is sensitive or which combination of ports is suspicious could facilitate the user to make a decision in favor of some itineraries and identify directly suspicious cases.

In the future, we plan to integrate other types of information as well. For example, we could take into account also the time dimension in order to estimate unnecessary delays. If we know, for instance, the usual time it takes for a container to travel from one port to another, we would be in a position to identify itineraries suffering from long delays and target those itineraries for further inspection.

References

Agovic A, Banerjee A, Ganguly AR, Protopopescu V (2009) Anomaly detection using manifold embedding and its applications in transportation corridors. Intell Data Anal 13(3):435–455

Chandola V, Banerjee A, Kumar V (2012) Anomaly detection for discrete sequences: a survey. IEEE Trans Knowl Data Eng, pp 823–839

Cook DJ, Holder LB (2000) Graph-based data mining. IEEE Intell Syst 15(2):32–41

Eberle W, Holder L (2007) Anomaly detection in data represented as graphs. J Intell Data Anal 11(6):663–689

Eberle W, Holder L, Massengill B (2012) Graph-based anomaly detection applied to homeland security cargo screening. International conference of the florida artificial intelligence research society (FLAIRS)

Cardoso B (2008) Standalone Multiple Anomaly Recognition Technique—SMART. http://www.sbir.gov/sbirsearch/detail/137676

Kou Y, Lu C, Chen D (2006) Spatial weighted outlier detection. Proceedings of the sixth SIAM international conference on data mining, Bethesda, MD, USA

Ling Y, Jin M, Hilliard MR, Usher JM (2009) A study of real-time identification and monitoring of barge-carried hazardous commodities. 17th International conference on Geoinformatics, pp 1–4

Lu C, Chen D, Kou Y (2008) Detecting spatial outliers with multiple attributes. Fifth IEEE international conference on tools with artificial intelligence, pp 122

Rissanen J (1989) Stochastic complexity in statistical inquiry. World Scientific Publishing Company

Sun P, Chawla S (2004) On local spatial outliers. Fourth IEEE international conference on data mining, pp 209–216

Swaney RE, Gianoulis ER (2008) Cargo X-ray image anomaly detection using intelligent agents—FORELL. http://www.sbir.gov/sbirsearch/detail/168550

Atmosphere and Ocean Data Processing in a Decision-Making Support System for Arctic Exploration

Oksana Smirnova and Nataly Zhukova

Abstract Presently, attention must be paid to the conditions in the Arctic region. The area has experienced significant environmental changes in ice cover, the atmosphere, and the ocean in recent years. In addition, there has been active development of the Arctic region. Data from the Arctic region are used for ensuring safe navigation of passenger and freight traffic between Europe, Asia, and America, as well as by fishing companies and companies that research and extract mineral resources. Current data acquisition, processing, and analysis systems for the Arctic region are highly specialized systems that do not perform complex analyse of all sets of available data. Consequently, these systems do not provide products of analysis with the desired precision and efficiency in the defined conditions. The problem can be solved by developing a decision-making support system that will provide acquisition, processing, and analysis of all sets of available data and also monitor and investigate the environment in the Arctic region. In the chapter, the results of analyse of atmosphere and ocean data are presented and a solution for integrated data processing and analysis based on intelligent geoinformation system (IGIS) technologies is proposed. The main components of IGIS that are necessary to provide decision-making support for Arctic exploration are considered. A developed prototype of decision-making support system for Arctic exploration is described and results of data analyses are given.

Keywords Atmosphere data processing · Ocean data processing · Decision support system · Intelligent GIS

O. Smirnova (✉) · N. Zhukova
St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS), 39, 14 Linia VO, St Petersburg 199178, Russia
e-mail: sov@oogis.ru

N. Zhukova
e-mail: nazhukova@mail.ru

V. Popovich et al. (eds.), *Information Fusion and Geographic Information Systems (IF&GIS 2013)*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-642-31833-7_20,

1 Introduction

Major changes have occurred in the Arctic region over the last decades. The air temperature has increased and, as consequence, the frequency and the intensity of cyclones have grown and the width of the ice cover has decreased. Also, the climate changes caused important changes to the thermohaline structures of the Arctic Ocean waters.

In recent years, several expeditions to the Arctic region were organized, including Arctic-2007/08, NABOS-2007/08, BARKALAV-2007/08, PALEX-2007/08, and SHELF-2010. The analysis of the results of these expeditions showed that extremely high water temperatures (up to +3 degrees) were observed in surface layers of the eastern part of the Arctic Ocean. Due to the extremely high water temperature, the distribution of surface water salinity was also abnormal; in two areas, salinity varied from −4 % to −6 %. The first area was located in a southern part of the Wrangel abyssal plain and an adjoining part of the Siberian shelf, whereas the second area was located in the Canadian abyssal plain (Ashik 2012).

In summer 2008, the positive anomalies of temperature and the negative anomalies of salinity in the surface layer in the majority of areas were less than in summer of 2007. However, the absolute values of these anomalies were so essential that the situation of the summer 2008 can be put on the second place after 2007.

The Arctic region is also characterized by inclement weather and an extreme ice situation. During the cold period of the year, the air temperature in the Arctic region can decrease to −40°. The annual number of the storms reaches 22, with an average duration of 9 days. The average height of waves is about 4 m, with a maximum of 13 m. Around planned placements of objects for industrial infrastructure, steady ice cover lasts 7 months. The thickness of an ice cover can reach 2 m; hummocks of thickness up to 6–8 m can be formed (Alekseev et al. 2009).

At the present time, data processing and analysis in the Arctic region are, as a rule, performed using a set of specialized systems, which and takes significant amount of time. The products of data processing and analysis are provided approximately 6 months after they are acquired. In addition, data about atmosphere, ocean, and ice conditions are provided by different data centers and the end-user interface is oriented primarily to domain experts. Because the situation in the Arctic region constantly changes, it is required to provide operational data in a convenient form for different categories of users. To meet the requirements for data acquiring, systematization, and storage according to the standards of data representation, adaptive processing, and analysis of available data sets using statistical methods and methods of intelligent analysis in a decision-making support system, the tasks of monitoring and research of the environment and development of a unified user-friendly interface need to be solved. The development of this decision-making support system is supported by a grant from the Office of Naval Research.[1]

[1] GRANT # 62909-12-1-7013. Decision Making Support System for Arctic Exploration, Monitoring and Governance.

2 Description of the Decision-Making Support System Structure

The overall goal of the decision-making support system (DMSS) is to develop an intelligent geographical information system (IGIS)-based system that will synthesize ocean/ice/atmosphere observations and model-based products of their analysis to allow fast access to the available information on the Arctic environment. The prospective end-users of the DMSS (various groups of stakeholders) may not necessarily be experts in the specific knowledge area. They need environmental data as supportive information to solve their professional tasks. Therefore, the basic requirements for the system are: (1) coverage of all environmental domains; (2) acquiring, processing, and analyzing data in automation mode; (3) presentation of information in a form that can be understood by a non specialist in the Earth sciences (e.g. engineer, skipper, administrator); and (4) user-friendly interface.

Technically, the system consists of two major components: the IGIS engine, which is the program package for storage, manipulation, and visualization of various types of geo-referenced data, and the data itself. The novel feature of the proposed system, compared to the existing analogs (e.g., Ocean Data Viewer (ODV), http://odv.awi.de) is its multidisciplinary character and low requirements for end-user skills in the subject domain.

IGIS systems that include means of artificial intelligence (AI) are the base of the developed decision-making support system. Application of IGIS technologies provides end-users with data about analyzed entities, phenomena, environment, and various results of its processing that are informative and convenient to use. The use of AI technologies allows one to substantiate variants of complex problems by taking into account various factors, their interrelations, and mutual influence. The basic elements of IGIS are ontology of subject domain and knowledge base. Ontology of subject domain contains descriptions of main concepts, objects and their properties, and relationships between concepts. The knowledge base contains production rules that describe objects, behavior and their interactions in different conditions. For data processing, AI components such as expert systems, inference machines, and intellectual data analyses methods are used. Two aspects of data processing are considered: organization of data processing and a set of algorithms for ocean and atmosphere data analyses.

Organization, execution, and estimation of data processing and analysis are provided by two main components—scenario approach support and an inference machine. Scenario approach allows complex data processing algorithms, which include computation blocks, to be automatically visually executed in the form of block diagrams. Processes can be easily modified and corrected at both stages—building and executing. The expert system is used to design, visualize, and execute scenarios of data processing. Application of the scenario approach for processing organization is described in Sorokin et al. (2003).

The component of intellectual data analysis contains a structured library of algorithms with semantic descriptions that is used at the stage of process execution. Application of algorithms of intellectual data analysis provides automated data processing in the conditions of permanent changes of structure and quality of input data. Adaptation of data processing algorithms to input data is made at the expense of the use of a priori knowledge that is available in system, as well as the results of performance of exploratory analysis procedures which, as a rule, are represented as a set of statistical tests; these provide additional facts about data. Selection of processing algorithms is performed using conditions of algorithm applications that are defined for each algorithm and estimation of obtained results using one or several specified criteria. Various boundary values used in algorithms as well as admissible values of calculated criteria are defined using statistical estimation of historical data. The choice of the parameters of processing algorithms is realized in two steps. First, initial values are defined and then results are iteratively estimated for various sets of parameters. Parameters that allow one to obtain best results according to end-user requirements are used. In most cases, algorithms of data processing occur in stages. The first stage is the training stage, when a range of admissible parameters and preferred initial parameters are defined based on the results of statistical data processing; at the second stage, the algorithm is applied for processing data.

Processing and analysis of atmosphere and oceanographic data are performed using methods that are applied in combination with intellectual analysis.

3 Description of Atmosphere and Oceanographic Data

Initial data were provided by the Arctic and Antarctic Research Institute, as well as several other data acquiring and analysis centers. The data described and analyzed here refer to the area of the Barents Sea.

Meteorological data are of two types. The first type is raw observations that are made by World Meteorological Organization meteorological stations located around the Barents Sea. This type of data is collected over a standard 6-hour time interval. The second type of data is assimilated meteorological data that is provided in the form of regular grids ($2.5 \times 2.5°$). Regular grids are products of reanalysis by the National Center for Environmental Prediction (NCEP) and National Center for Atmospheric Research (NCAR). Station data is transmitted via the channel of Automatic Transmission System (ASPD) every 6 h (at 0, 6, 12, 18 GMT). Data is decoded and stored in text format. An example of atmosphere data is given in Table 1. Atmosphere data includes information on atmospheric pressure at sea level, air temperature and humidity at 2 m height, and wind speed and direction at 10 m height.

Gridded meteorological data is calculated by means of modeling/assimilation procedures in the framework developed within the NCEP/NCAR Reanalysis collaborative program. The overall aim of this project is to generate global digital

Table 1 Example of atmosphere data

Date	Time	Id	Wind direction	Wind speed	Temperature	Humidity	Pressure
01.01.2011	0	20,046	50	9	−33.4	71	1,024.3
01.01.2011	0	20,107	90	6	−17	55	1,021.7
01.01.2011	0	22,113	170	6	−11.3	75	994.3
01.01.2011	0	20,744	180	9	−5.7	87	1,004.3
01.01.2011	0	22,028	130	4	−12.3	80	995.4
01.01.2011	0	23,114	180	12	−6.1	89	1,011.2
01.01.2011	3	22,113	180	6	−11.4	85	994.3
01.01.2011	3	20,107	90	4	−17.8	55	1,023.1
01.01.2011	3	22,292	190	13	−5.7	90	1,003
01.01.2011	3	22,193	180	16	−4.6	93	1,002.5
01.01.2011	3	20,107	90	4	−17.8	55	1,023.1
01.01.2011	3	22,113	180	6	−11.4	85	994.3

arrays containing consistent meteorological information from 1948 untill the present time (NCEP/NCAR Reanalysis at PSD 2012). All available meteorological information worldwide is used for this task. Quality control of raw data and modeling/assimilation are fulfilled using the climate data assimilation system (CDAS). Regular grids are kept unchanged over the reanalysis period. The main efforts are concentrated on gathering data from all available sources located on land; on board ships, aircraft, and pilot-balloons; and received from satellites; tasks of quality control and data assimilation are solved by the CDAS. Gridded meteorological data also can be used in the DMSS.

Atmosphere data may be used to describe meteorological conditions and as the input data for calculation of derivative parameters (e.g., energy exchange at the surface, wind current, ice drift).

Oceanographic data include deep observations of water area parameters and results of their processing. Observed temperature and salinity profiles are considered, as recorded, from more than 135,000 various stations during the period from 1870 to present. In Fig. 1, the distribution of measurements made at different years and months is given. For executing measurements, such stations as CTD, XCTD-2, T-7, ITP, IMB, PPF, ARGOS, and ARGO buoys were used. In all, more than 2 million observations were received at latitudes from 65° to 85° S and longitudes from 16° to 65° E at depths down to 450 m. The distribution of measurements by geographic coordinates is given in Fig. 2. The distribution of measurements by depths and months is given in Fig. 3.

For each measurement, the following descriptions are provided: earth coordinates (latitude, longitude, and depth) where the measurement was made; time (year, month, day, and hour) of measurement; and values of temperature, salinity, and sound speed in water. An examples of ocean initial data given in Table 2. An example of profiles received from the ARGO buoy at points P1(70.081°N 3.677°E), P2(70.229°N 2.593°E), P3(70.014°N 1.787°E), P4(69.878°N 2.220°E), P5(69.765°N 2.353° E), P6(69.937°N 2.672°E), P7(69.937° N 3.429°E), P8(69.600°N 3.023°E) is given in Fig. 4.

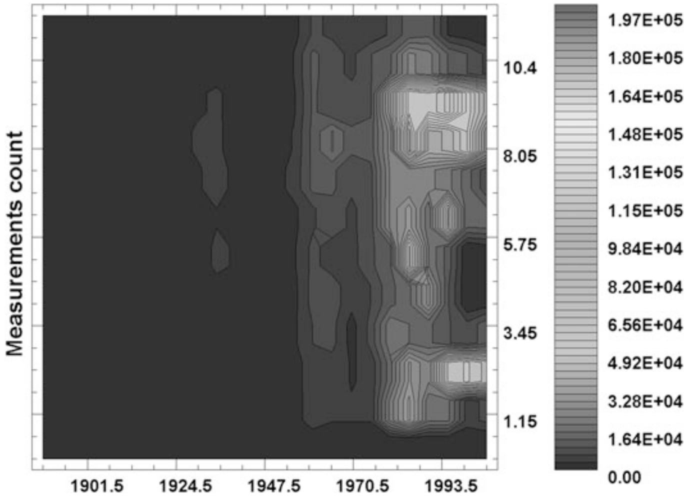


Fig. 1 Distribution of measurements by years and months

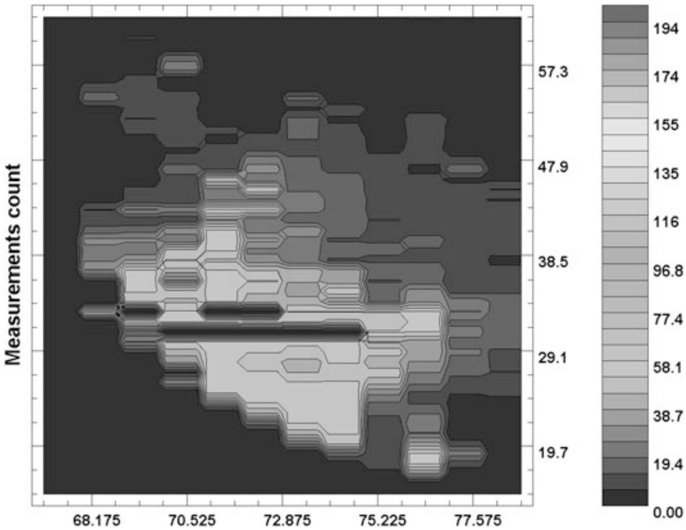


Fig. 2 Distribution of measurements by geographic coordinates

Oceanographic observations for the period from 1900 to 2011 that passed quality control were interpolated at standard depth levels and integrated monthly to build monthly observed fields. All monthly observed fields were objectively analyzed to generate horizontal gridded fields by means of data interpolation variation analysis (Boyer et al. 2005). Climatic fields have no time gaps. An example of a calculated gridded field of water temperature for January for latitude 71°N, longitude from 30 to 33°E, and depth from 0 m to 400 m is given in Fig. 5.

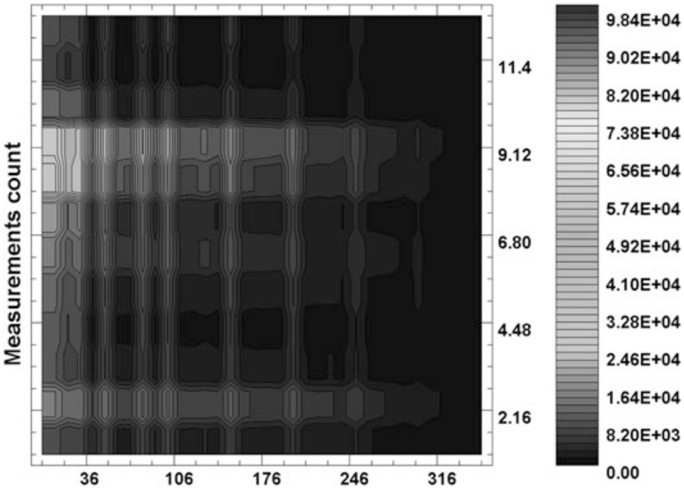


Fig. 3 Distribution of measurements by depths and months

Table 2 Example of ocean initial data

Latitude	Longitude	Year	Month	Day	Hour	Sound	d/p	Temparature	PSAL
68.4	41.83333	1870	7	22	0	−9	0	9	−9
68.4	41.83333	1870	7	22	0	−9	36	6.4	−9
68.4	41.83333	1870	7	22	0	−9	45	5.7	−9
68.4	41.83333	1870	7	22	0	−9	54	5.6	−9
68.65	43.41667	1870	7	22	0	−9	0	7	−9
68.65	43.41667	1870	7	22	0	−9	45	4.9	−9
68.65	43.41667	1870	7	22	0	−9	55	4.7	−9
69.1	37.8	1870	7	31	0	−9	0	9.5	−9
69.1	37.8	1870	7	31	0	−9	36	6.9	−9
69.11667	38.18333	1870	7	31	0	−9	0	10.6	−9
69.11667	38.18333	1870	7	31	0	−9	36	7.4	−9
69.16667	38.91667	1870	7	31	0	−9	0	11.5	−9
69.16667	38.91667	1870	7	31	0	−9	36	6.4	−9
69.25	39.33333	1870	7	31	0	−9	0	11.1	−9

Observational data sets are essential for monitoring and understanding changes in ocean climatology and for ocean–atmosphere model initialization and validation. Ocean water parameter fields with high resolution and spatial–temporal coverages are used for solving practical tasks.

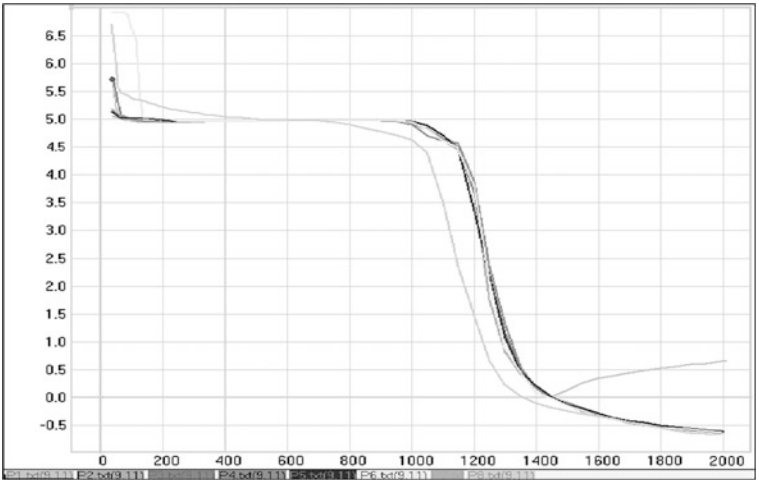


Fig. 4 Profiles of temperature parameters measured by ARGO buoys

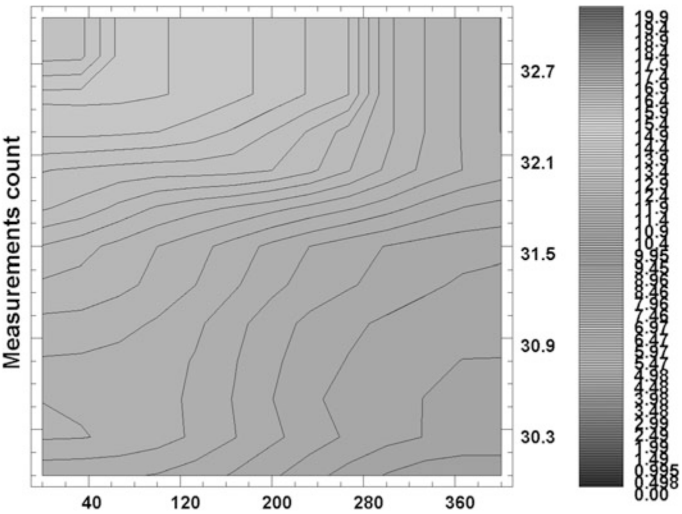


Fig. 5 Example of calculated gridded field of water area temperature

4 Atmosphere and Ocean Data Processing in Decision-Making Support System

An executed analysis of measurements showed that measurements are irregular both in time and coordinates, with the exception of data received from fixed stations. Data and its characteristics for different regions are far from being similar and need special solutions for their processing. Also, external factors strongly influence received measurements (e.g., seasonal phenomena, state of water environment of contiguous area). In addition, the data contain a considerable number of error values, such as noise, outliers, gaps, and also offsets and trends due to errors of measuring tools. All of these features are to be considered when processing and analyzing acquired measurements.

Atmosphere and ocean data processing are performed in two stages: the preprocessing stage (data verification stage) and the regularization stage.

Data verification stage. The main objectives of data verification are the gathering, storage, processing, and analysis of received data so that the data can be used for building regular grids. In the verification stage, the performance of the following tasks are to be solved:

1. data harmonization, which provides data conversion from various formats to the format used in the unified model of object domain (Popovich and Voronin 2005);
2. data integration, which is oriented on operational data quality control based on application of specialized sets of tests, detection, and exclusion of duplicating values, while removing noise and outliers. Data integration includes cluster analysis of measurements in time and space and definitions of boundary conditions for measurements in different regions;
3. data fusion, which has the purpose of building formalized statistical descriptions of measurements (Zhukova et al. 2013) and detailed statistical data quality control. Data fusion creates and improves of statistical models for various regions.

Data regularization stage. The main objectives of the regularization stage are building regular grids of parameters values using gathered measurements and estimation of data accuracy in knots of a regular grid. The stage of data regularization assumes solution of the following key tasks (Korablev et al. 2007): (1) data interpolation on standard depth levels; (2) data quality control of the interpolated values; (3) data interpolation to standard depth levels taking into account results of interpolated values, quality control; (4) interpolated values, quality control based on the standard deviations of parameter values for month periods; and (5) application of data objective analysis methods.

It is necessary to solve two main problems for the automation of oceanographic and atmospheric data processing and analysis:

1. At the verification stage, estimation of data quality is to be organized in the shortest possible time without involvement of experts of subject domain. Using the results of data estimation, a solution if measurements can be used for improvement of regular grids is to be made;
2. For each of the processing stages, depending on the structure of processed data, features of measurement instrument, and the region, where measurements were made, selection of the most suitable algorithm from the library of data processing and analysis algorithms is to be organized. This problem is solved using

the scenario approach (Sorokin et al. 2003) application to building and executing processes, which provides a selection of algorithms using a set of production rules that are interpreted and used by expert systems.

The algorithm for operative data processing (Algorithm 1) based on data mining algorithms is proposed for providing quality control of operative measurements.

Algorithm 1

Input: (1) array of historical data H that satisfies the specified requirements to quality of data; each element of data is a vector that describes one measurement in the following feature space: time (year, months, day) and earth coordinates (latitude, longitude, depth) of the measurement value; in some cases, it may be reasonable to expand feature space and consider measurements of several parameters simultaneously; (2) array of newly acquired data, the quality of which is estimated by $O = \{o_1, \ldots, o_n\}$, where n is the total number of measurements.

Output: vector of flags $Q = \{q_1, \ldots, q_n\}$ that determine the quality of each measurement; list of measurements D that are defined as duplicated; list of measurements that contain errors R.

Step 1 (training step)
1.1. $f_{cl}(H) = C, C_T = \{c_1, \ldots, c_k\}, H = \{h_{ij}\}, i \in (0, r), j \in (0, m)$, where f_{cl} is the algorithm of cluster analyses; C is the set of temporary clusters, k is the total number of clusters, r is the total number of measurements in historical data, and m is the number of features that describe measurement.
{analyses of temporary clusters}
1.2. **for** $c_i \in C_T$

{if measurement values within cluster are a compact cluster, it does not need further decomposition and is included into the result set of clusters}
if $|f_{cl}(c_i)| = 1$ **then** $C_R \leftarrow c_i$, where C_T is the set of output clusters
{cluster is not compact enough and needs further decomposition}
else $C_T \leftarrow f_{cl}(c_i)$
{calculation of result cluster characteristics}
1.3. **for** $c_i \in C_R$

calculate center of cluster and standard deviation of measurements values within the cluster
calculate boundary values B_i for measurements in the formed cluster
Step 2 (estimation step)
2.1. **define** cluster $c_k \in C_R$ that corresponds to the region, where analyzed measurement $X' = \{x_i\}_{i=0}^{i=m}$ was made
{procedure for determining border values}
2.2. **execute** standard procedure of quality control tests for measurement instruments using defined boundary values B_k;
fill vector of flags Q using results of quality control tests
{procedure for determining duplicated values}

2.3. **calculate** $d = \min(\{dist(X', X_j\}),\ j \in (0, v)$, where v is the total number of elements in cluster $c_k elementfromcluster, and X_j$ is the element from cluster c_k;
if ($d < d_{border}$) **then** $D \leftarrow X'$ is a duplicated value, where d_{border} is a priori defined border value;
{procedure for removing noise and outliers}
2.4. $f_{cl}(X' \cup \{X_j\}_{j=1}^{v}) = C^E$, where X_j is the element from cluster $c_k; and\, C^E$ is the set of obtained clusters
if ($\exists c_k \in C^E : |c_k| = 1$) **then** $R \leftarrow X'$; it is considered that the measurement value contains and error;
else the measurement value is considered to be correct;

Quality flags $q_i \in Q$ are presented in binary form (test was passed successfully/test was not passed) and additionally can contains information about the distance from the analyzed measurement to the closest border value. The list of duplicated values D as well as list of measurements that contain errors R can be also extended by information about distances between analyzed measurements and measurements that are considered to be duplicates.

For parameter profiles that are represented as the time series of measurements, it is important to analyze not only measurement values but also the dynamics of time series behavior (Algorithm 2). The algorithm assumes that the following stages are to be executed: building formalized description of profiles, building patterns that correspond to typical variants of time series behavior, and analysis of newly acquired profile match templates.

Algorithm 2

Input: (1) array of historical profiles that were processed by experts and satisfy specified requirements of data quality: $F = \{f_j\}_{j=0}^{n}$, $f_j = \{(t_i, v_i)\}_{i=0}^{m}$, where f is the profile of a parameter; t_i is the time and coordinates of i-th measurement; v_i is the value of i-th measurement, n is the number of historical profiles; m is the length of a profile; (2) array of newly acquired data that needs quality estimation $O = \{o_1, \ldots, o_s\}$, where s is the total number of profiles that are to be analyzed.

Output: vector of flags $Q = \{q_1, \ldots, q_s\}$ that contain estimations of measurement quality for each profile.

Step 1 (training step)
{procedure of building segments}
1.1. **for each** f_k **calculate** $f_{segm}(f_k) = S_k$, where f_{segm} is the algorithm of segmentation, $S_k = \{t_j, v_j\}_{j=1}^{r}$, r is the number of segments

for each $S_k, k = 0\ldots n$

for each $s_j \in S_k,\ j \in 1, \ldots, r, r$ is the number of segments
{procedure of building formalized description for segment}
$f_{forml}(s_j) = \bar{g}_j$, where f_{forml} is the algorithm for building formal description, $\bar{g}_j$ is the vector of segment characteristics and, $j = 0 \ldots h$, where h is the number of characteristics

{procedure of building formalized description for profile}
calculate $\bar{g}_k = \bigcup_j \bar{g}_j$
Step 2 (training step)
{procedure of building cluster of profiles with similar behavior}
$f_{cl}(\bigcup_k \vec{g}_k) = \{Z_l\}_{l=0}^{p}$, where f_{cl} is the algorithm of cluster analyses, Z is the cluster of profile descriptions, p is the total number of clusters
Step 3 (estimation step)
foreach $o_i \in O$

calculate $d = \min(\{dist(o_i, \{Z_l\}_{l=0}^{p})\}), X$ to Q^P, where Q^P- standard template
if $d < d_{border}$ **then** o_i has typical behavior
else o_i has unexpected behavior

Segmentation of profiles is made according to borders of water layers. For building formalized description of segments, it is reasonable to use cubic spline interpolation because measurement profiles are smooth functions. The proposed algorithm of cluster analyses is described below (Algorithm 3).

Algorithm 3

Input: formalized descriptions of profiles $\{\vec{g}_k\}_{k=0}^{n}$, n is the number of profiles; *MaxNCl* is the maximum number of clusters.

Output: a set of clusters $\{Z_l\}$; each cluster corresponds to one of the typical parameter's behavior.

Step 1 (algorithm initialization)
if ($\exists$ vectors with given prior information)
then take the first vector $\bar{g}_k$ of them
1.2. **form** the first cluster Z_1 from feature vector $\bar{g}_k = <g_{k1}, g_{k2}, \ldots, g_{kn}>$ with center $\bar{e}^{(1)} = <g_{k1}, g_{k2}, \ldots, g_{kn}>$, where top index $^{(1)}$ indicates cluster number;
define $N^{(1)} = 1$, which is the number of profiles in formed cluster;
define $N_{clust} = 1$, which is the number of clusters;
Step 2 (processing vectors)
for $j \in (2, \ldots, n)$

find cluster m where distance w is minimal:

$$w : \left\| \bar{g}_j - \bar{e}^{(m)} \right\| \to \min, \text{where } \rho_{jm} = \begin{cases} \infty, & \textit{if profiles are incompatible} \\ \sum_{k=1}^{h} (g_{kj} - g_{km})^2, & \textit{in other cases} \end{cases},$$

h is the total number of characteristics in formalized descriptions. Profiles are incompatible if, according to a priori information, they belong to different types;
if $(w < r^{(u)})$, where $r^{(u)} = \left(\frac{1}{2} \left(\sqrt{r^{(j)}} + \sqrt{r^{(m)}} + \sqrt{w(r^{(j)}, r^{(m)})} \right) \right)^2$ is the border value for distance
then add vector $\bar{g}_j$ to cluster m

calculate $\bar{e}^{(m)} = \frac{\bar{e}^{(m)} N^{(m)} + \bar{g}_j}{N^{(m)} + 1}$;

$$N^{(m)} = N^{(m)} + 1$$

else

if $(N_{clust} < MaxNCl)$
then find clusters k_1 and k_2 with minimal distance *dist*
if $(dist < w)$
then merge clusters k_1 and k_2
calculate $r^{(k1)} = \left(\frac{1}{2}\left(\sqrt{r^{(k1)}} + \sqrt{r^{(k2)}} + \sqrt{w(k_1, k_2)}\right)\right)^2$;
$\bar{e}^{(k1)} = \frac{\bar{e}^{(k1)} N^{(k1)} + \bar{e}^{(k2)} N^{(k2)}}{N^{(k1)} + N^{(k2)}}$;
$N^{(k1)} = N^{(k1)} + N^{(k2)}$, where $N^{(k1)}, N^{(k2)}$ are the number of vectors in clusters k_1 and k_2, respectively
create new cluster Z_j for profile j
calculate $N_{clust} = N_{clust} + 1$;

$$Z \leftarrow Z_j$$

create new cluster Z_j for profile j
calculate $N_{clust} = N_{clust} + 1$;

$$Z \leftarrow Z_j$$

For solving the task of cluster analysis, any other distance-based algorithm can be used that does not need to define a priori number of clusters.

5 Representation of Atmosphere and Ocean Data and Results of Their Processing in DMSS

Initial data and processed data are provided in the form of text tables (Fig. 6) and also in a graphical form on the map of the world (Fig. 7). DMSS supports various data views. Also, various statistical characteristics and distributions can be calculated, and dependence between various parameters can be determined (Fig. 8). The results of data processing are provided to the end user in the form of a hierarchy of grids with various permissions on time, depth, and coordinates.

For each of the grids, the following information can be displayed: values of the gridded parameter in knots of grids, estimation of reliability of the calculated

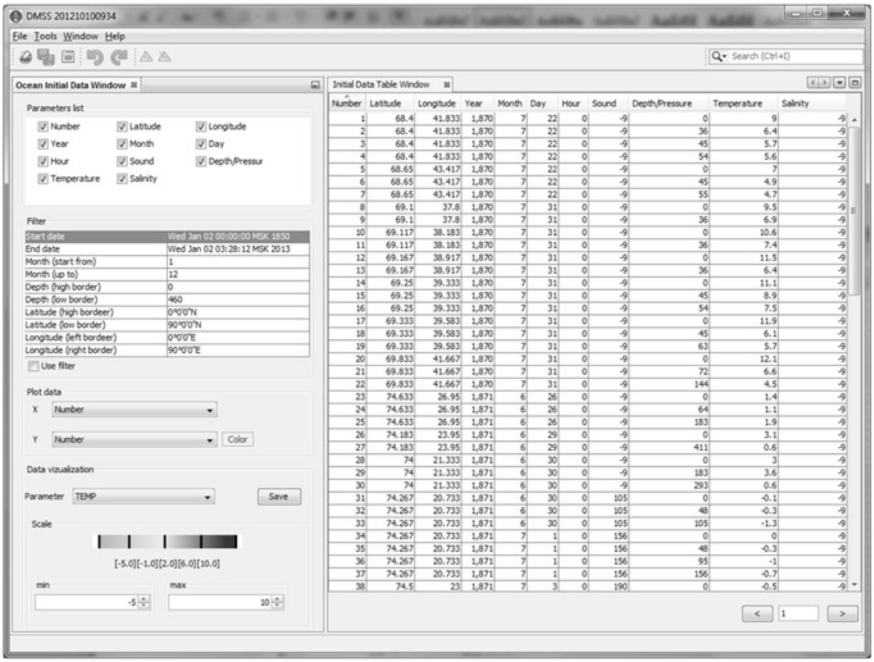


Fig. 6 Representation of initial oceanographic data. The output table (*on the right*) includes all measurements that match user queries (*on the left*)

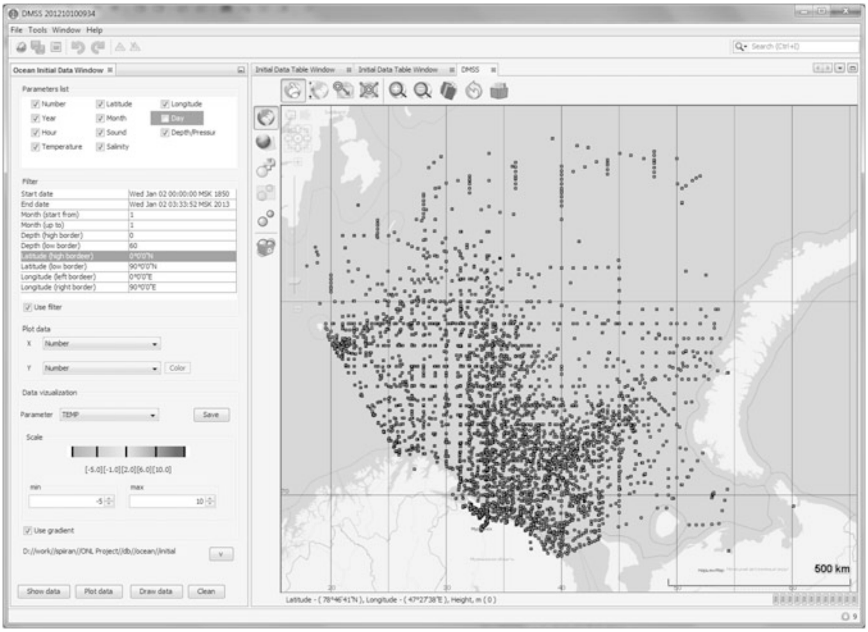


Fig. 7 Representation of water temperature initial measurements on the map of the world. The color of the points corresponds to a temperature value

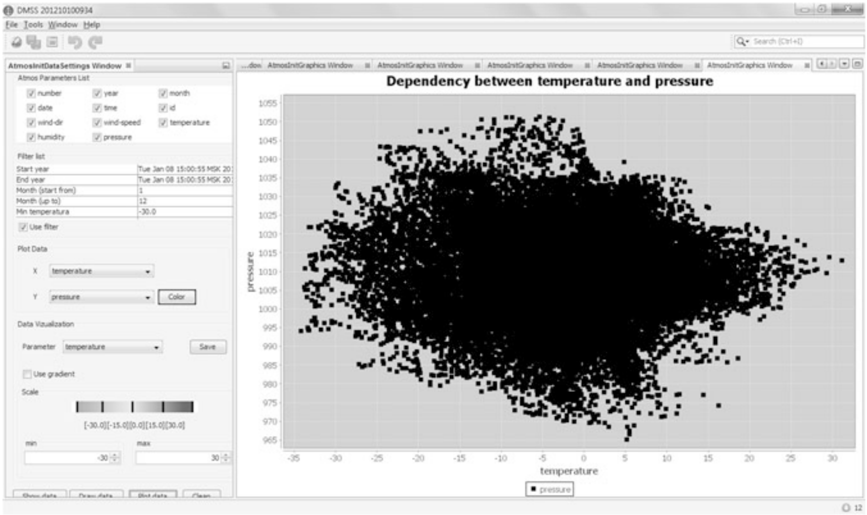


Fig. 8 Visualization of dependency between temperature and pressure for initial atmosphere data

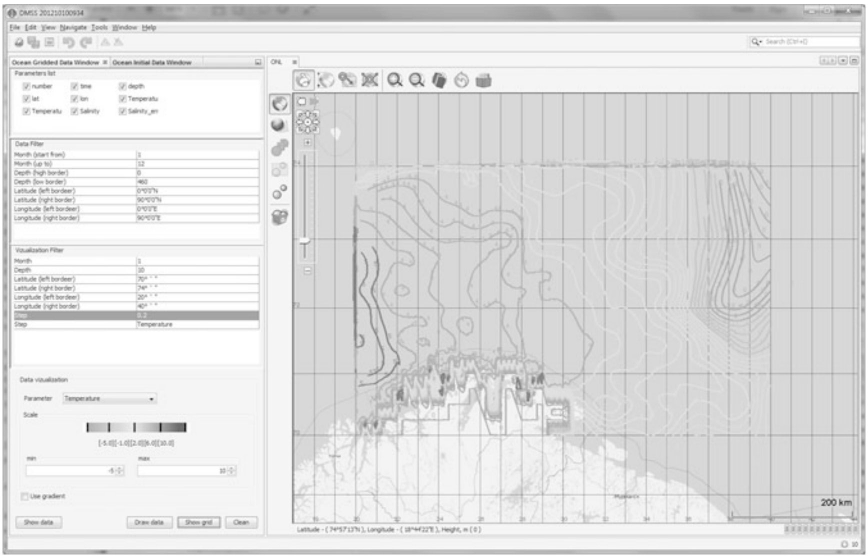


Fig. 9 Graphical representation of gridded temperatures on the map of the world

values, and also, if necessary, measurements that were used for calculation of values in grids knot. Figure 9 gives an example of temperature parameter regularization results.

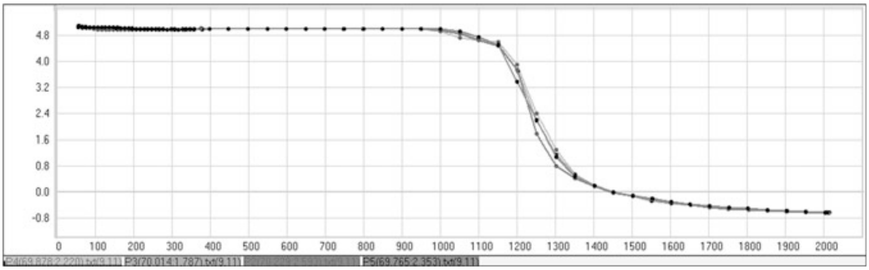


Fig. 10 Temperature profiles of Argo buoys in a graphical representation of gridded temperatures on the map of the world

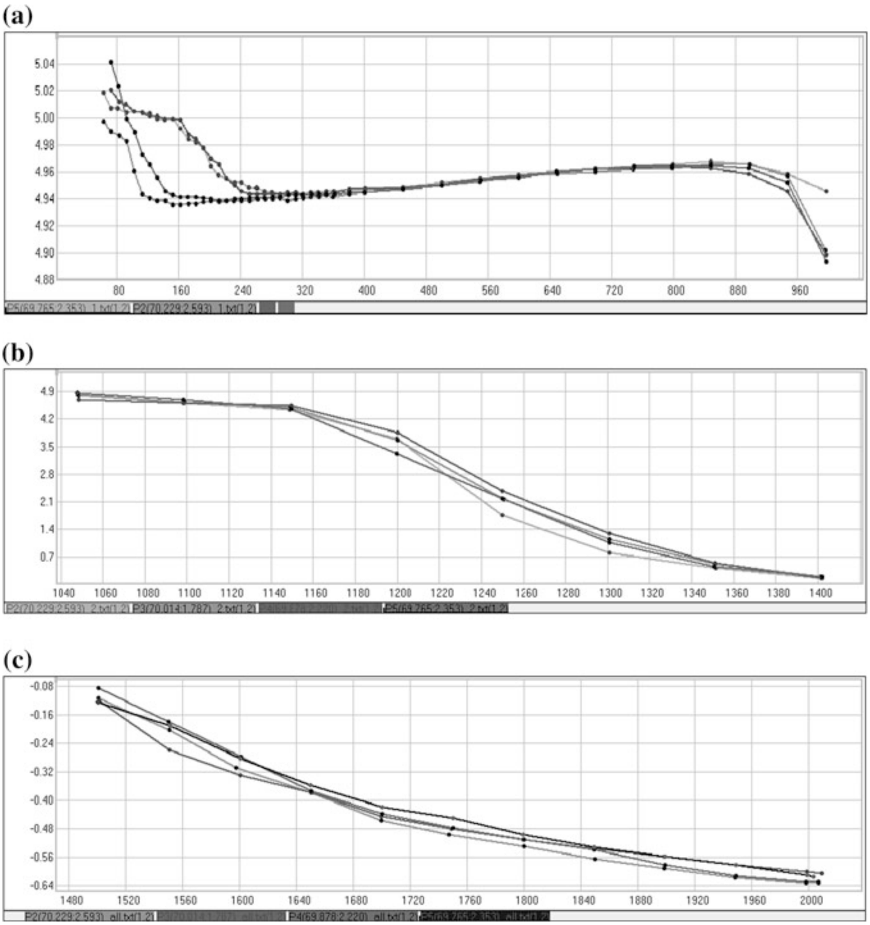


Fig. 11 Results of temperature profile segmentation. **a** First segment of temperature profiles (depth from 60 m to 1000 m). **b** Second segment of temperature profiles (depth from 1000 m to 1400 m). **c** Third segment of temperature profiles (depth from 1400 m to 2000 m)

Table 3 Formalized descriptions of temperature profiles

	Segment 1				Segment 2				Segment 3			
P2	59.7	999.4	4.903	4.997	1,050.3	1,400	0.172	4.805	1,500	2008.3	−0.634	−0.113
P3	60.4	1,000.2	4.899	5.02	1,050.2	1,399.8	0.207	4.828	1,500.4	2010.7	−0.606	−0.12
P4	60	1,000.2	4.894	5.061	1,050.5	1,400.5	0.175	4.691	1,499.6	2008.3	−0.63	−0.083
P5	60	1,000.1	4.943	5.018	1,050.1	1,400.6	0.215	4.878	1,499.6	2005.2	−0.614	−0.124

Fig. 12 Results of cluster analysis of temperature profiles

6 Case Study

The task of automation of data quality control was solved for a limited region (69.6°–70.08°N; 2.593°–3.023°E). Input data profiles of temperature from Argo buoys that were received in the defined region were considered. The first step of data quality control builds a formalized description of temperature profiles for the defined region using historical data. In Fig. 10, four temperature profiles from the processed set are given.

Each profile was segmented, taking into account borders of different water layers that are located at 60, 1000, and 1400 m. Results of segmentation are given in Fig. 11. Each segment was formally described using coefficients of cubic spline interpolation. A formal description of analyzed profiles is shown in Table 3. Using cluster analysis algorithm, SimpleKMeans vectors that describe profiles were decomposed in three clusters. Clusters are given in Fig. 12. Each cluster corresponds to one of the typical behaviors of temperature profiles for the analyzed region.

The second step of the quality control algorithm is processing new data. A test profile that was considered as new data is given in Fig. 13. The test profile was

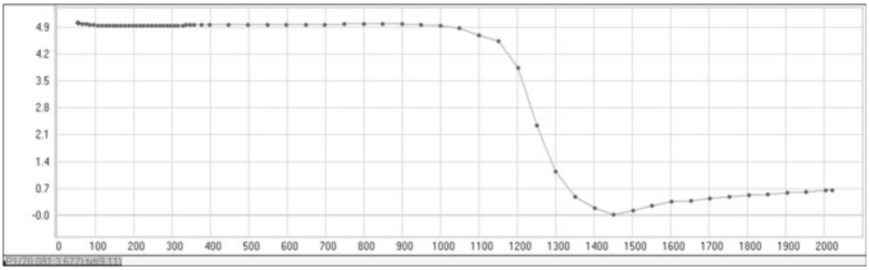


Fig. 13 Temperature profile considered as test profile

Table 4 Description of test temperature profile

Segment 1			
60.1	1000.3	4.939	4.99
Segment 2			
1,049.7	1,400.3	0.185	4.876
Segment 3			
1,500.4	2,016.6	0.125	0.655

Fig. 14 Results of cluster analysis of profiles corresponding to typical behavior of profiles and a test profile

segmented and formally described using the same approach that was described for processing historical data. A formalized description of the test profile is shown in Table 4.

Cluster analysis was applied to a data set in which descriptions of one profile from each of the defined clusters for historical data along with the test profile were included. The result set of clusters is given in Fig. 14. Definition of conformity profile by template.

A separate cluster is created for the test profile. That is because of the profile behavior at the range of depth from 1400 m down to 2000 m. It means that the test profile does not correspond to any of types of profile behavior and consequently does not pass quality control. Profiles that did not pass quality control are to be analyzed by experts manually.

7 Conclusion

In this chapter, our proposed approach for atmosphere and ocean data processing in a decision-making support system was presented, using Arctic exploration data for the Barents Sea acquired since 1870. Decisions about measurement quality proposed by the DMSS were compared to decisions that were made by experts in the subject domain in the same situations. The analysis showed that our proposed approach for automation processing reduced the time of data quality control by 20 % and also increased the accuracy of processing results at the expense of application of data mining algorithms. By processing oceanographic and atmosphere data using intelligent geoinformation technologies and intelligent data analysis technologies within one system, a wide range of users can obtain the necessary data on the Arctic region more efficiently.

Acknowledgments The project for developing a decision-making support system is supported by the Office of Naval Research (ONR) global, Project # 62909-12-1-7013, "Decision Making Support System for Arctic Region".

References

Ashik I (2012) Recent Russian marine research activities in the arctic ocean. Legal aspects of marine science in the arctic ocean. In: Wasum-Rainer S, Winkelmann I, Tiroch K (eds) vol 235. p 374

Alekseev GA, Pnyushkov AV, Ivanov HE, Ashik IM, Sokolov VT (2009) Complex assessment of climatic changes in the sea Arctic with use of data of MPG 2007/2008. Problems of the Arctic and Antarctic, vol 1(81). pp 7–14

Boyer T, Levitus S, Garcia H, Locarnini RA, Stephens C, Antonov J (2005) Objective analyses of annual, seasonal, and monthly temperature and salinity for the World Ocean on a 0.25° grid. Int J Climatol 25:931–945

Korablev AA, Pnyushkov AV, Smirnov AV (2007) Creation of an oceanographic database for climate monitoring in the North European basin of the Arctic. In: Trudy AANII. Issue 447. pp 85–108

NCEP/NCAR Reanalysis at PSD (2012). URL: http://www.esrl.noaa.gov/psd/data/reanalysis/reanalysis.shtml

Popovich VV, Voronin MN (2005) Data harmonization, Integration and fusion: three sources and three major components of Geoinformation Technologies. In: Proceedings of IF&GIS, September 25–27, St. Petersburg, Russia

Sorokin RP, Shaida SS, Smirnov AV (2003) Intelligent geoinformation systems for modeling and simulation. In: Proceedings of the international workshop on harbour, Maritime and multimodal logistics modelling and simulation. riga TU

Zhukova N, Ignatov D, Smirnova O (2013) Dynamic information model for oceanographic data representation. In: 20th international conference on conceptual structures, Mumbai, India