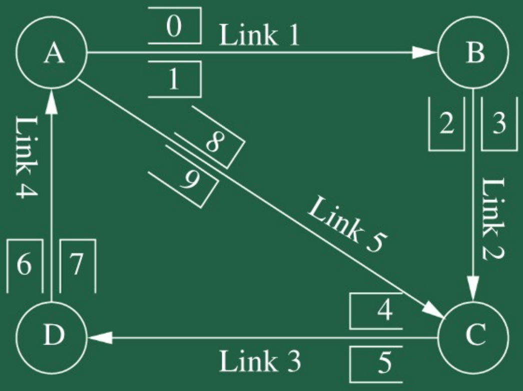
Systems & Control: Foundations & Applications

E.H. Abed
Editor

Advances in Control, Communication Networks, and Transportation Systems

In Honor of Pravin Varaiya



Birkhäuser

Systems and Control: Foundations & Applications

Advances in Control, Communication Networks, and Transportation Systems

In Honor of Pravin Varaiya

E. H. Abed
Editor

Birkhäuser
Boston • Basel • Berlin

E.H. Abed
University of Maryland
Institute for Systems Research and
Department of Electrical
and Computer Engineering
College Park, MD 20742
U.S.A.

Library of Congress Cataloging-in-Publication Data
Advances in control, communication networks, and transportation systems:
in honor of Pravin Varaiya / Eyad H. Abed, editor.
p. cm. – (Systems & control)
Essays in honor of Professor Pravin Varaiya; also presented at a meeting held in Berkeley, June 5-7, 2005
Includes bibliographical references.
ISBN 0-8176-4385-0 (alk. paper)
1. Feedback control systems. 2. Data transmission systems. 3. Computer networks. I. Abed, Eyad H. II. Varaiya, P. P. (Pravin Pratap) III. Series.

TJ217.A39 2005
629.8′3–dc22 2005048106

ISBN-10 0-8176-4385-0 e-ISBN 0-8176-4409-1 Printed on acid-free paper.
ISBN-13 978-0-8176-4385-0

Printed in the United States of America. (TXQ/SB)

9 8 7 6 5 4 3 2 1 SPIN 11381235

www.birkhauser.com

Dedicated to Pravin Varaiya on the occasion of his 65th birthday

Pravin Varaiya

Contents

Part I Hybrid Systems

Part II System Theory and Design

Part III Networks

Part IV Transportation

Preface

This volume was prepared in conjunction with a Symposium held in Berkeley June 5–7, 2005, as a tribute to Professor Pravin Varaiya. The contributions represent most of the lectures given at the meeting. The Symposium brought together former students, collaborators and friends from throughout the world to celebrate Pravin's career as he approached the memorable occasion of his 65th birthday. The authors, speakers, organizers, supporters and attendees of the Symposium are very pleased to dedicate this work to Pravin, to congratulate him on his many seminal contributions, and to thank him for his leadership in the fields of systems, control and networks over the past four decades.

Pravin Varaiya was born on October 29, 1940 in Bombay, India. He earned the B.E. degree in Electrical Engineering from the University of Bombay in 1960, and then began his graduate studies at the University of California, Berkeley. The early 1960s was an exciting time during which the foundations of systems and control were developed, and Berkeley contributed to this development through the research of Professors Arthur Bergen, Charles Desoer, Mac Hopkin, Eli Jury, Elijah Polak, Otto Smith, and Lotfi Zadeh. Professor Eugene Wong joined the faculty in 1963 and contributed to the understanding of stochastic systems. Berkeley attracted outstanding visiting faculty, including Moshe Zakai and Bill Root. The faculty trained and mentored a strong group of graduate students, including Mike Athans, Dick Mortensen, Jack Wing, Jim Eaton, Cesare Galtieri, Barry Whalen, and Pravin Varaiya.

Pravin was a Member of the Technical Staff of Bell Laboratories during 1962–1963, following completion of his M.S. at Berkeley. He and Ruth Kosh were married on June 30, 1963, while Pravin was at Bell Labs. Through the years, Ruth would be a constant supporter of Pravin and his work.

Pravin joined the faculty of the Department of Electrical Engineering and Computer Sciences at U.C. Berkeley upon completing his Ph.D. in 1966. From the beginning, his work was marked by creativity, rigor, timeliness and impact. These qualities alone are enough to launch an outstanding career in academia. However, Pravin demonstrated something more: an unusual breadth of interest and ability that extended beyond his original research area of control and optimization to include communication and information theory, stochastic processes, game theory, and circuit and network theory. In each area that he entered, Pravin made important and lasting contributions. He quickly moved through the academic ranks, gaining the rank of Professor in 1970, and became well known and highly regarded in the field at an early age.

His interests continued to expand, building on a solid foundation in system and control theory and mathematics. In the early 1970s, Pravin started his research in economics, focusing mainly on issues of urban economics, such as the design of rent control, urban land use, and the economics of home ownership, but also contributing to general economic theory. This entry into economic research was followed by his appointment as Professor of Economics at Berkeley in 1975. His teaching and research duties would be split between electrical engineering and economics until 1992, when he decided to again focus his full attention on engineering.

Later in the 1970s and into the 1980s, Pravin steadily increased his activities in the area of communications, with an emphasis on communication networks, and simultaneously began a research effort in the field of electric power systems, focusing on dynamics and control of nonlinear power system models. Under his direction, teams of research students and visitors made important contributions to the understanding of many important issues in these areas. In the 1980s he also began research efforts in discrete event systems and hybrid systems, as well as in pricing issues both for communication network services and for electric power. In the late 1980s he became involved in what was to become a major commitment with the Institute of Transportation Studies at Berkeley and the California PATH Program (Partners for Advanced Transit and Highways), a multi-university research program dedicated to the solution of California's transportation problems. He made seminal contributions to the design of intelligent vehicle highway systems, building on his past research in large scale, multilayer, and hybrid systems. From 1994 to 1997 he was Director of the PATH program.

Looking back over four decades, the contributions of Pravin Varaiya are difficult to summarize in a few pages of this book. He has published extensively, having authored or co-authored four books and more than 280 technical papers. His papers and books are lucid and have influenced many researchers, undergraduate and graduate students, and practitioners. He has also served on the Board of Directors of several technology companies, and has personally been involved in technology transfer through involvement in start-up companies. In addition to his research, teaching, and writing activities, Pravin has consistently managed to also maintain a significant level of public service by working for human rights causes around the world.

During leaves of absence from Berkeley, Pravin has held visiting appointments at the Federal University of Rio de Janeiro (Fall 1970) and MIT (January 1974–January 1975).

Pravin Varaiya has been recognized with many awards and distinctions. He has held a Guggenheim Fellowship (1972) and a Miller Research Professorship (July 1978–June 1979). He holds an Honorary Doctorate from l'Institut National Polytechnique de Toulouse. In 2002 he was awarded the IEEE Field Medal in Control Systems for "outstanding contributions to stochastic and adaptive control and the unification of concepts from control and computer science." He is a Fellow of the

IEEE and a member of the National Academy of Engineering. He is currently Nortel Networks Distinguished Professor in the Department of Electrical Engineering and Computer Sciences at the University of California, Berkeley. He has served on the editorial boards of many prominent journals, and is currently on the editorial boards of Discrete Event Dynamical Systems, Transportation Research—C, and the Journal of Economic Dynamics and Control.

As mentioned earlier, this volume represents most, but not all, of the presentations at the Symposium. The chapters are written by experts, and each is at the forefront of research or presents a careful review and assessment of an important topic. Since it was not feasible to cover all the areas in which Pravin has contributed over the years, emphasis was placed on the areas of his greatest current interest. The chapters are thus broadly classified into four categories: I. Hybrid Systems; II. System Theory and Design; III. Networks; and IV. Transportation. Connections with the work of Pravin Varaiya are evident in all of the contributions, and many authors chose to make these connections explicitly.

Part I consists of three chapters on hybrid systems. The first chapter, by Hwang, Stipanović and Tomlin, extends numerical methods for reachability analysis developed for linear systems to feedback linearizable systems, linear dynamic games, and norm-bounded nonlinear systems. The chapter by Kurzhanski solves a problem of measurement-based feedback control of systems with unknown but bounded uncertainties using an ellipsoidal approximation technique developed with Varaiya. The third chapter, by Piazza and Mishra, develops a concept of functional hybrid automata for use in biological system models, then proceeds to reduce these models to differential equation models and to study their stability.

Part II also contains three chapters. Davis gives a survey of representation of martingales as stochastic integrals, with applications to special classes of stochastic processes and to mathematical finance. The second chapter is by Lee, who discusses combining computing and engineering through redesign of the systems part of the undergraduate curriculum. The third chapter, by Deshpande, outlines some of the emerging themes in system design for network equipment, emphasizing systems for which throughput performance is the primary concern.

Part III consists of seven chapters on networks. Gastpar uses information-theoretic bounds to extend to sensor networks a result of Varaiya and Walrand on the effectiveness of feedback in a causal coding context. Liu and Goldsmith consider the joint design of a wireless network and networked controllers, and illustrate the framework they introduce through cross-layer optimization of the link layer, MAC layer, and sample period selection of a double inverted pendulum system. Garg, Borkar and Manjunath consider the pricing of network resources to control demand behavior during periods of congestion in the internet. Gupta and Walrand propose a novel backoff mechanism for ad-hoc networks and demonstrate its tendency to improve fairness. Baras and Jiang discuss a distributed cooperative game-theoretic framework for trust establishment in distributed networks such as mobile ad-hoc networks, sensor networks and ubiquitous computing systems. Johari and Tsitsiklis consider network settings such as communication networks and power systems, and study the design of market mechanisms that minimize efficiency loss and are robust to gaming behavior of market participants. Stoenescu and Teneketzis present a mechanism design theory view of decentralized network resource allocation.

Part IV contains four chapters on transportation systems. Shladover discusses automated highway systems research with a focus on Pravin Varaiya's influence on the field. Kotsialos and Papageorgiou use optimal control and simulation to study the potential of freeway network ramp metering control to reduce congestion. The third chapter, by Mahmassani and Zhou, develops a state-space model for real-time trip demand pattern estimation and prediction, along with optimal updating algorithms and application to real network data. The fourth chapter, by Horowitz, Muñoz and Sun, presents a new traffic-responsive on-ramp switched control strategy as well as test results obtained using a traffic simulator.

On behalf of all the contributors to this Festschrift volume and the participants in the associated Symposium, the members of the Organizing Committee would like to congratulate Pravin Varaiya on his uniquely successful career to-date, and to wish him many more years of continued success. Mostly, however, the contributors and organizers would simply like to wish him well on his 65th birthday, and to sincerely thank him for his leadership in research, and for being such a dedicated mentor, colleague and friend of the many students and research collaborators who have been privileged to work with him.

College Park, MD	*E.H. Abed*
Stanford, CA	*A. Goldsmith*
Berkeley, CA	*R. Horowitz*
Urbana, IL	*P.R. Kumar*
Berkeley, CA	*S.S. Sastry*
March 2005	Symposium Organizing Committee

Acknowledgments

We are grateful to the Defense Advanced Research Projects Agency for its support of the Symposium on Systems, Control and Networks. Dr. Sri Kumar of DARPA saw the importance of the Symposium and approved critical funding early in the planning stages of the meeting. We would also like to sincerely thank Dr. Kishan Baheti of the National Science Foundation for facilitating a very helpful NSF travel grant. Logistic and material support of the Institute for Systems Research, University of Maryland and the Department of Electrical Engineering and Computer Sciences of the University of California at Berkeley are gratefully acknowledged. Several staff members at these organizations provided essential administrative and planning support. We would like to in particular thank Karen Deal and Rebecca Copeland of the University of Maryland and Jasna Mrkic and Dana Dee Little of the University of California at Berkeley. We are very grateful to the Symposium's Outreach Committee, consisting of René Boel, Mustafa Ergen and Sonia Sachs, who worked to reach past students and colleagues of Pravin Varaiya in advance of the Symposium. The advice of colleagues and friends is also much appreciated, and in particular we would like to thank John Baras, Jean Paul Jacob, Alexander Kurzhanski, Ruth Varaiya and Jean Walrand for their advice at various stages of the project. The Editor is also very grateful to Munther Hassouneh of the University of Maryland for his assistance with Latex typesetting, and to Tom Grasso and his able staff at Birkhäuser Boston for their help and expert advice during the preparation of this volume.

Publications of Pravin Varaiya

A. Books

1. *Notes on Optimization,* Van Nostrand Reinhold, New York, 1972.
2. *Stochastic Systems: Estimation, Identification, and Adaptive Control,* (with P.R. Kumar), Prentice-Hall, 1986.
3. *High-Performance Communication Networks,* (with J. Walrand), Second Edition, Morgan-Kaufmann, 2000.
4. *Structure and Interpretation of Signals and Systems,* (with E.A. Lee), Addison Wesley, 2002.

B. Edited Volumes

5. P. Varaiya and A.B. Kurzhanski, Editors, *Discrete Event Systems: Models and Applications,* Springer-Verlag, Lecture Notes in Control and Information Sciences, No. 103, 1988.
6. P.R. Kumar and P.P. Varaiya, Editors, *Discrete Event Systems, Manufacturing Systems, and Communication Networks,* IMA Volumes in Mathematics and Its Applications, Vol. 73, Springer-Verlag, 1995.

C. Papers

7. "Saturation throughput analysis of IEEE 802.11 wireless LANs for a lossy channel," (with X. Dong), *IEEE Communications Letters,* Vol. 9, No. 2, February 2005.
8. "Ellipsoidal techniques for hybrid dynamics: The reachability problem," (with A.B. Kuzhanski), *16th Mathematical Theory of Networks and Systems Conference* (MTNS2004), July 5-9, 2004, Leuven, Belgium.
9. "Distributed control of teams of unmanned air vehicles," (with J. Sousa, T. Simsek), *16th Mathematical Theory of Networks and Systems Conference* (MTNS 2004), July 5-9, 2004, Leuven, Belgium.
10. "Traffic measurement and vehicle classification with a single magnetic sensor," (with S-Y. Cheung, S. Coleri, B. Dundar, S. Ganesh, C-W. Tan), *Transportation Research Board, 84th Annual Meeting,* Washington, D.C., January 2005.
11. "The congestion pie: Delay from collisions, potential ramp metering gain, and excess demand," (with J. Kwon), *Transportation Research Board, 84th Annual Meeting,* Washington, D.C., January 2005.
12. "Real-time estimation of O-D matrices with partial trajetories from ETC tag data," (with J. Kwon), *Transportation Research Board, 84th Annual Meeting,* Washington, D.C., January 2005. Also to appear in *Transportation Research Record,* 2005.

13. “WTRP–Wireless token ring protocol,” (with M. Ergen, D. Lee and R. Sengupta), *IEEE Trans. Vehicular Technology,* Vol. 53, No. 9, pp. 1863-1881, November 2004.
14. “An efficient incentive-compatible combinatorial market mechanism,” (with R. Jain), *Proc. 42nd Allerton Conference on Communiation, Control, and Computing,* University of Illinois, Urbana, IL, September 2004.
15. “Sensor networks for traffic monitoring,” (with S. Coleri and S. Cheung), *Proc. 42nd Allerton Conference on Communiation, Control, and Computing,* University of Illinois, Urbana, IL, September 2004.
16. “Fault tolerance and energy efficiency of data aggregation schemes for sensor networks,” (with S. Coleri), *Proc. IEEE Vehicle Technology Conference,* Los Angeles, September 2004.
17. “Hybrid algorithm for indoor positioning using wireless LAN,” (with J. Kwon, B. Dundar and M. Ergen), *Proc. IEEE Vehicular Technology Conference,* Los Angeles, September 2004.
18. “Combinatorial bandwidth exchange: Mechanism design and analysis,” (with R. Jain), *Communications in Information and Systems,* Vol. 3, No. 4, pp. 305-324, September 2004.
19. “Extensions of PAC learning for partially observable Markov decision processes,” (with R. Jain), *Proc. Conf. on Information Sciences and Systems (CSIS),* Princeton University, March 2004.
20. “Scalar estimation and control with noisy binary observations,” (with T. Simsek and R. Jain), *IEEE Trans. Automatic Control,* Vol. 49, No. 9, pp. 1598-1603, 2004.
21. “Systematic identification of freeway bottlenecks,” (with C. Chen and A. Skabardonis), *Transportation Research Board, 83rd Annual Meeting,* Washington, D.C., January 2004. Also *Transportation Research Record,* No. 1867, pp. 46-52, 2004.
22. “Statistical methods for detecting errors in spatial configuration information of traffic surveillance sensors,” (with J. Kwon, C. Chen), *Transportation Research Board, 83rd Annual Meeting,* Washington, D.C., January 2004. Also *Transportation Research Record,* o. 1870, pp. 124-132, 2004.
23. “A system for displaying travel times on changeable message signs,” (with C. Chen and A. Skabardonis), *Transportation Research Board, 83rd Annual Meeting,* Washington, D.C., January 2004.
24. “Measuring traffic,” (with P. Bickel, C. Chen, J.Kwon, J. Rice and E. van Zwet), *Statistical Science,* to appear.
25. “Admission control and throughput analysis of IEEE 802.11,” (with M. Ergen), *ACM Kluwer MONET Special Issue on WLAN Optimization at the MAC and Network Levels,* to appear.
26. “Grid discretization based method for anisotropic shortest path problem over continuous regions,” (with Z. Jia), *Proc. IEEE Conference on Decision and Control,* 2004.

27. “On some nonstandard dynamic programming problems of control theory,” (with A.B. Kurzhanski), in *Variational Analysis and Applications*, F. Giannessi and A. Maugeri, Eds., Kluwer Acad. Pubs., NY, 2004.
28. “Reachability under uncertainty and measurement noise,” (with A.B. Kurzhanski), accepted for publication in *Computation and Control,* 2004.
29. “On the problem of control synthesis through available measurements,” (with A.B. Kurzhanski), accepted for publication in *Applied Mathematics and Mechanics (PMM),* 2004.
30. “Control synthesis for state constrained systems and obstacle problems,” (with A.B. Kurzhanski and I. Mitchell), *Proc. of IFAC Symposium on Nonlinear Control Systems (NOLCOS),* 2004, Stuttgart.
31. “Improving the aggregate throughput of access points in IEEE 802.11 wireless LANs,” (with X. Dong, M. Ergen and A. Puri), *Proc. IEEE WLN,* Bonn, Germany, October 2003.
32. “Router scheduling configuration based on the maximization of benefit and carried best effort traffic,” (with E. Magana and D. Morato), *Telecommunication Systems,* Vol. 24, Nos. 2-4, pp. 275-292, 2003.
33. “Comparison of wireless token ring protocol with IEEE 802.11,” (with M. Ergen, D. Lee, R. Datta, J. Ko, A. Puri and R. Sengupta), *J. of Internet Technology,* Vol. 4, No. 4, 2003.
34. “Mechanism design for networking research,” (with J. Shu), *Information Systems Frontiers,* Vol. 5, No. 1, pp. 29-37, January 2003.
35. “QoS Aware adaptive resource allocation techniques for fair scheduling in OFDMA based broadband wireless access,” (with M. Ergen and S. Coleri), *IEEE Transactions on Broadcasting,* Vol. 49, No. 4, pp. 362-370, December 2003.
36. “Pricing network services,” (with J. Shu), *Proc. INFOCOM 2003, San Francisco,* Vol. 2, pp. 1221-1230, April 2003.
37. “Travel time reliability as a measure of service,” (with C. Chen, E. van Zwet, A. Skabardonis), *Transportation Research Board, 82nd Annual Meeting,* Washington, D.C., January 2003. To appear in *Transportation Research Record.*
38. “Detecting errors and imputing missing data for single loop surveillance systems,” (with C. Chen, J. Kwon, A. Skabardonis and J. Rice), *Transportation Research Board, 82nd Annual Meeting,* Washington, D.C., January 2003. Also *Transportation Research Record,* No. 1855, pp. 160-167, 2003.
39. “Measuring recurrent and nonrecurrent traffic congestion,” (with A. Skabardonis), *Transportation Research Board, 83rd Annual Meeting,* Washington, D.C., January 2003. Also *Transportation Research Record,* No. 1856, pp. 118-124, 2003.
40. “Estimation of truck volume from single loop detector using lane-to-lane speed correlation,” (with J. Kwon and A. Skabardonis), *Transportation Research Board, 83rd Annual Meeting,* Washington, D.C., January 2003. Also *Transportation Research Record,* No. 1856, pp. 106-117, 2003.
41. “Tuning the weights in WFQ schedulers for the maximization of carried best effort traffic,” (with E. Magana and D. Morato), *10th International Conference on Telecommunications,* ICT’2003, Tahiti, February 2003.

42. “Reachability under uncertainty,” (with A. Kurzhanski), *Proc. IEEE Conf. on Decision and Control,* Las Vegas, pp. 1982-1987, December 2002.
43. “Control under communication constraints,” (with T. Simsek), *Proc. IEEE Conf. on Decision and Control,* Las Vegas, pp. 3209-3216, December 2002.
44. “Freeway performance measurement system (PeMS): An operational analysis tool,” (with T. Choe and A. Skabardonis), *TRB 81st Annual Meeting,* Washington, D.C., January 2002.
45. “Reachability under state constraints – the ellipsoidal technique,” (with A.B. Kurzhanski), in *Proc. of the IFAC World Congress*, Barcelona, Spain, 2002.
46. “Optimization methods for target problems of control,” (with A.B. Kurzhanksi), Proc. MTNS-2002, Notre Dame, Indiana, 2002.
47. “On the existence of competitive equilibria in bandwidth markets,” (with R. Jain and A. Dimakis), *40th Annual Allerton Conference on Communiation, Control, and Computing,* Urbana, IL, October 2002.
48. “On reachability under uncertainty,” (with A.B. Kurzhanski), *SIAM J. Control and Optimization,* Vol. 41, No. 1, pp. 181-216, 2002.
49. “Application of GPS to mobile IP and routing in wireless networks,” (with M. Ergen, S. Coleri, B. Dundar, R. Jain and A. Puri), Proc. IEEEE Vehicular Technology Conference, pp. 1115 - 1119, 24-28 September 2002.
50. “Elliposoidal techniques for reachability, Part I,” (with A.B. Kurzhanski), *Optimization: Methods and Software,* Vol. 17, No. 2, pp. 177-206, 2002.
51. “Elliposoidal techniques for reachability, Part II,” (with A.B. Kurzhanski), *Optimization: Methods and Software,* Vol. 17, No. 2, pp. 207-237, 2002.
52. “California’s performance measurement system: Improving freeway efficiency through transportation intelligence,” *TR News,* No. 218, pp. 18-24, Jan.-Feb. 2002.
53. “A game-theoretic model for generation expansion planning: problem formulation and numercial comparisons,” (with A. Chuang and F. Wu), *IEEE Trans. Power Systems,* Vol. 16, No. 4, pp. 885-891, November 2001.
54. “Causes and cures of highway congestion,” (with C. Chao and Z. Jia), *IEEE Control Systems Magazine,* Vol. 21, No. 4, pp. 26-33, Dec. 2001.
55. “The PeMS algorithm for accurate, real-time estimates of g-factors and speeds from single-loop detectors,” (with Z. Jia, C. Chen, B. Coifman), *4th IEEE ITSC Conference,* Oakland, CA, August 2001.
56. “Space division multiple access (SDMA) for robust ad hoc vehicle communication networks,” (with S.V. Bana), *4th IEEE ITSC Conference,* Oakland, CA, August 2001.
57. “Real-time vehicle location with desired accuracy,” (with S.V. Bana), *4th IEEE ITSC Conference,* Oakland, CA, August 2001.
58. “Freeway performance measurement system: mining loop detector data,” (with C. Chen, K. Petty et al), *Transportation Research Board, 80th Annual Meeting,* January 2001, Washington D.C. Also, *Transportation Research Record,* No. 1748, 96-102, 2001.
59. “Dynamic optimization for reachability problems,” (with A.B. Kurzhanski), *J. Optimization Theory and Applications,* Vol. 108, No. 2, pp. 227-251, 2001.

60. "Ellipsoidal techniques for reachability analysis," (with A.B. Kurzhanski), *Hybrid Systems: Computation and Control,*N. Lynch and B.H. Krogh Eds., Springer-Verlag, LNCS Vol. 1790, pp. 202-214, 2000.
61. "On the reachability problem under persistent disturbances," (with A.B. Kurzhanski), *Doklady Mathematics,* Vol. 61, No. 3, pp. 3809-384, 2000.
62. "Reachability under disturbances," (with A.B. Kurzhanski), Proc. 21st Allerton Conference on Communiation, Control, and Computing, Univ. of Illinois at Urbana-Champaign, October 2000.
63. "Ellipsoidal techniques for reachability analysis: Internal approximation," (with A.B. Kurzhanski), *Systems and Control Letters,* Vol. 41, pp. 201-211, 2000.
64. "Control design of an automated highway system," (with R. Horowitz), *Proc. of the IEEE,* Vol. 88, No. 7, pp. 913-925, July 2000.
65. "Quality matters: Some remarks on Internet service provisioning and tariff," (with J. Altmann and B. Rupp), *Telektronik,* Vol. 96, No. 2, pp. 20-25, 2000.
66. "A question about hierarchical systems," in T.E. Djaferis and I.C. Schick, Eds., *System Theory: Modeling, Analysis and Control,* SESC 158, Kluwer, 2000.
67. "Coordinated multilateral trades for electric power markets: Theory and implementation," (with F.F. Wu), *Electrical Power and Energy Systems,* Vol. 21, pp. 75-102, 1999.
68. "Internet demand under different pricing schemes," (with J. Altmann and B. Rupp), *Proc. of the ACM Conference on Electronic Commerce (E-COMMERCE '99),*Denver, 1999.
69. "Managing usage-based pricing in a future telecommunication market," (with B. Altmann), *PAAM'99, Fourth International Conference on Practical Application of Intelligent Agents and Multi-Agents,* London, April 1999.
70. "Towards an integrated services Internet: The demand side of things," (with B. Rupp and R. Edell), *Proc. of the 1998 Conf. of the International Telecommunications Society,* 1998.
71. "Providing Internet access: What we learn from INDEX," (with R. Edell), *IEEE Network,* Vol. 13, No. 4, 1999.
72. "SmartAHS: A simulation framework for Automated Vehicles and Highway Systems," (with A. Gollu), *Mathematical and Computer Modelling,* Vol. 27, Nos. 9-11, pp. 103-128, 1998.
73. "Bifurcations associated with sub-synchronous resonance," (with Y. Mitani, K. Tsuji, M. Varghese and F.F. Wu), 1997 Summer IEEE Power Meeting, Berlin; also *IEEE Trans. on Power Systems,* Vol. 13, No. 1, pp. 139-144, February 1998.
74. "Capacity of fading channels with side information," (with A. Goldsmith), *IEEE Trans. Information Theory,* Vol. 43, No. 6, pp. 1986-1992, November 1997.
75. "I-880 Field experiment: Analysis of incident data," (with A. Skabardonis, K. Petty, R.L. Bertini, H. Noeimi and D. Rydzewski), *Transportation Research Record,* Vol. 1603, pp. 72-79, 1997.
76. "Capacity, mutual information, and coding for finite-state Markov channels," (with A. Goldsmith), *IEEE Transactions on Information Theory,* Vol. 42, No. 3, pp. 468-486, May 1996.

77. “The freeway service patrol evaluation project: Database support programs and accessibility,” (with K. Petty, H. Noeimi, K. Sanwal, D. Rydzewski, A. Skabardonis and H. Al-Deek), *Transportation Research – C,* Vol. 4, No. 2, pp. 71-85, 1996.
78. “Folk theorems on transmission access: proofs and counterexamples,” (with F. Wu, P. Spiller and S. Oren), *Journal of Regulatory Economics,* Vol. 10, No. 1, pp. 5-23, July 1996.
79. “Decidable hybrid systems,” (with A. Puri), *Mathl. Comput. Modelling,* Vol. 23, Nos. 11/12, pp. 191-202, 1996.
80. “What’s decidable about hybrid automata?,” (with T.A. Henzinger, P.W. Kopke and A. Puri), in *Proceedings of the 27th Annual ACM Symposium on Theory of Computing (STOC 1995),* pp. 373-382, Las Vegas, 29 May-1 June 1995.
81. “What’s decidable about hybrid automata?,” (with T.A. Henzinger, P.W. Kopke and A. Puri), *Journal of Computer and System Sciences,* Vol. 57, pp. 94-124, 1998.
82. “Stability of hybrid systems,” (with M. Kourjanski), in *Hybrid Systems III: Verification and Control,* R. Alur, T.A. Henzinger and E.D. Sontag, Eds., Lecture Notes in Computer Science, Vol. 1066, Springer-Verlag, Berlin, pp. 413-423, 1996.
83. “Design and evaluation tools for automated highway systems,” (with A. Deshpande, D. Godbole, A. Gollu), in *Hybrid Systems III: Verification and Control,* R. Alur, T.A. Henzinger and E.D. Sontag, Eds., Lecture Notes in Computer Science, Vol. 1066, Springer-Verlag, Berlin, pp. 138-148, 1996.
84. “Epsilon-approximation of differential inclusions,” (with V. Borkar and A. Puri), in *Hybrid Systems III: Verification and Control,* R. Alur, T.A. Henzinger and E.D. Sontag, Eds., Lecture Notes in Computer Science, Vol. 1066, Springer-Verlag, Berlin, pp. 362-376, 1996.
85. “Epsilon-approximation of differential inclusions,” (with V. Borkar and A. Puri), *Proc. 34th IEEE Conf. on Decision and Control,* New Orleans, pp. 2892-2897, Dec. 1995.
86. *High-Performance Communication Networks,* (with J. Walrand), Morgan–Kaufmann, 1996.
87. “The I-880 field experiment: database development and incident delay estimation procedures,” (with A. Skabardonis, K. Petty and H. Noeimi), *Seventy-Fifth Annual Meeting of the Transportation Research Board,* Washington, D.C. January 7-11, 1996. Also *Transportation Research Record,* vol. 1554, 204–212, 1996.
88. “Theory of traffic flow in automated highway systems,” (with M. Broucke), *75th Annual Meeting of the Transportation Research Board,* Washington, D.C. January 7-11, 1996. Also, *Transportation Research–C,* Vol. 4, No. 4, pp. 181-210, 1997.
89. “Verification of hybrid systems using abstraction,” (with A. Puri), in *Hybrid Systems II,* P. Antsaklis, W. Kohn, A. Nerode and S. Sastry, Eds., Lecture Notes in Computer Science, Vol. 999, Springer-Verlag, Berlin, pp. 359-369, 1995.

90. “Viable control of hybrid systems,” (with A. Deshpande), in *Hybrid Systems II,* P. Antsaklis, W. Kohn, A. Nerode and S. Sastry, Eds., Lecture Notes in Computer Science, Vol. 999, Springer-Verlag, Berlin, pp. 128-147, 1995.
91. “Bifurcation, chaos, and voltage collapse in power systems,” (with C-W Tan, M. Varghese and F.F. Wu), *Proceedings of the IEEE* Vol. 83, No. 11, pp. 1484-1495, November 1995.
92. “Modeling and verification of hybrid systems,” (with A. Puri), *Proc. American Control Conference,* pp. 4466-4470, 1995.
93. “Information structures for control and verification of hybrid systems,” (with A. Deshpande), *Proc. American Control Conference,* pp. 2642-2647, 1995.
94. “Design of entry and exit maneuvers of IVHS,” (with F. Eskafi and E. Singh), *Proc. American Control Conference,* pp. 3566-3570, 1995.
95. “Driving safely in smart cars,” (with A. Puri), *Proc. American Control Conference,* pp. 3597-3599, 1995.
96. “Billing users and pricing for TCP,” (with R. Edell and N. McKeown), *IEEE J. Selected Areas in Communications,* Vol. 13, No. 7, pp. 1162-1175, September 1995.
97. “Modeling and Performance of Real-Time Disk Access Policies,” (with J. Yee), *Computer Communications,* Vol. 18, No. 10, pp. 725-741, October, 1995.
98. “Burst reducing servers in ATM networks,” (with S. Low), *Queueing Systems: Theory and Applications,* Vol. 20, Nos. 1-2, pp. 61-84, 1995.
99. “Pricing of electric power under uncertainty: Information and efficiency,” (with T. Ishikida), IEEE Power Engineering Society Summer Meeting, Paper No. 94/SM/526-6/PWRS, San Francisco, July 24-28, 1994. Also, *IEEE Transactions on Power Systems,* Vol. 10, No. 2, pp. 884-890, May 1995.
100. “An automated highway system simulator,” (with F. Eskafi and D. Khorramabadi), *Transportation Research-C,* Vol. 3, No. 1, pp. 1-17, 1995.
101. “Nodal prices and transmission rights: a critical appraisal,” (with S.S. Oren, P.T. Spiller, and F. Wu), *The Electricity Journal,* Vol. 8, No. 3, pp. 24-35, April 1995.
102. “The Rabin index and chain automata, with applications to automata and games,” *Proc. Computer Aided Verification’95,* Brussels, Belgium, pp. 253-266, July 3-6, 1995.
103. “Modeling real-time systems using rate automata,” (with J. McManis), in P.R. Kumar and P.P. Varaiya, Eds., *Discrete Event Systems, Manufacturing Systems, and Communication Networks,* pp. 177-200, IMA Volumes in Mathematics and its Applications, Vol. 73, Springer-Verlag, 1995.
104. “An algorithm for optimal service provisioning using resource pricing,” (with S. Low), *Proceedings IEEE Infocom ’94,* 368-373, Vol. 1, Toronto, Canada.
105. “The entropy of traffic streams in ATM virtual circuits,” (with N.T. Plotkin), *Proc. IEEE INFOCOM,* Toronto, pp. 1038-1045, 1994.
106. “Control issues in automated highway systems,” (with J.K. Hedrick and M. Tomizuka), *IEEE Control Systems Magazine, Vol. 14, No. 6,* pp. 21-32, December 1994.

107. "Control of multiple service, multiple resource communication networks," (with S. Jordan), *IEEE Transactions on Communications,* Vol. 42, No. 11, pp. 2979-2988, November 1994.
108. "Multi-armed bandit problem revisited," (with T. Ishikida), *J. Optimization Theory and Applications,* Vol. 83, No. 1, pp. 113-154, October 1994.
109. "Discretization of timed automata," (with A. Gollu and A. Puri), *Proc. 33rd IEEE Conference on Decision and Control,* Lake Buena Vista, FL, pp. 957-958, December 1994.
110. "Formal verification of the PATHO real-time operating system," (with F. Balarin, K. Petty and A.L. Sangiovanni-Vincentelli), *Proc. 33rd IEEE Conf. Decision and Control,* Lake Buena Vista, FL, pp. 2459-2465, December 1994.
111. "SmartDb: An object-oriented simulation framework for Intelligent Vehicle Highway Systems," (with A. Gollu, A. Deshpande and P. Hingorani), *Proceedings of the Fifth Annual Conference on AI, Simulation, and Planning in High Autonomy Systems,* Gainesville, FL, pp. 244-250, December 1994.
112. "A decision-feedback maximum-likelihood decoder for finite-state Markov channels," (with A. Goldsmith), *Proc. IEEE Internat. Conference on Communications,* pp. 1546-1550, May 1994.
113. "Capacity, mutual information, and coding for finite-state Markov channels," (with A. Goldsmith), *Proc. IEEE Internat. Symp. Information Theory,* Trondheim, Norway, p. 322, 1994.
114. "Roadside intelligence for flow control in an intelligent vehicle and highway system," (with B.S.Y. Rao), *Transportation Research—C,* Vol. 2, No. 1, pp. 49-72, 1994.
115. "Decidability of hybrid systems with rectangular differential inclusions," (with A. Puri), *Proc. Computer Aided Verification '94* Lecture Notes in Computer Science, Vol. 818, Springer-Verlag, pp. 95-104, 1994.
116. "Suspension automata: A decidable class of hybrid automata," (with J. Mcmanis), *Proc. Computer Aided Verification '94,* Lecture Notes in Computer Science, Vol. 818, Springer-Verlag, pp. 105-117, 1994.
117. "An analytical model for real-time multimedia disk scheduling," (with J. Yee), in P. Venkat Rangan, Ed., *Network and Operating System Support for Digital Audio and Video,* Third International Workshop Proceedings, Springer-Verlag, pp. 315-320, 1993.
118. "Bifurcation and chaos in power systems," (with C-W Tan, M. Varghese and F. Wu), *Sadhana, Proc. Indian Academy of Sciences,* Vol. 18, Part 5, pp. 761-786, September 1993.
119. "A new approach to service provisioning in ATM networks," (with S. Low), *IEEE/ACM Transactions on Networking,* Vol. 1, No. 5, pp. 547-553, October 1993.
120. "Scheduling cells in an input-queued switch," (with N. McKeown and J. Walrand), *Electronics Letters,* Vol. 29, no. 25, pp. 2174-2175, December 1993.
121. "Investigations into achievable capacities and stream stability with coordinated intelligent vehicles," (with B.S.Y. Rao and F. Eskafi), *Transportation Research Record,* No. 1408, pp. 27-35, 1993.

122. "Flow benefits of autonomous intelligent cruise control in mixed manual and automated traffic," (with B.S.Y. Rao), *Transportation Research Record,* No. 1408, pp. 35-43, 1993.
123. "Protocol design for an automated highway system," (with A. Hsu, F. Eskafi and S. Sachs), *Discrete Event Dynamic Systems,* Vol. 2, pp. 183-206, 1993.
124. "Interruptible electric power service contracts," (with C.-W. Tan), *J. Economic Dynamics and Control,* Vol. 17, No. 3, pp. 495-517, May 1993.
125. "Smart cars on smart roads: Problems of control," *IEEE Transactions on Automatic Control,* Vol. 38, No. 2, pp. 195-207, Feb. 1993.
126. "Optimizing demand side contributions to power system security," (with R.J. Kaye and F.F. Wu) in S.S. Oren and S.A. Smith, Eds., *Service Opportunities for Electric Utilities: Creating Differentiated Products,* Kluwer, pp. 147-168, 1993.
127. "Design of interruptible electric power service contracts with stochastic demands," (with C.-W. Tan and T. Ishikida), *Elektrik: Turkish Journal of Electrical Engineering and Computer Sciences,* Vol. 1, No. 1, pp. 14-26, 1993.
128. "A deterministic fluid model for cell loss in ATM networks," (with M.K. Wong), *Proc. IEEE INFOCOM '93, San Francisco,* pp. 395-400, March 30-April 1, 1993.
129. "Burstiness bounds for some burst reducing servers," (with S.H. Low), *Proc. IEEE INFOCOM '93, San Francisco,* pp. 2-9, March 30-April 1, 1993.
130. "Performance analysis of parallel ATM connections for gigabit speed applications," (with N.T. Plotkin), *Proc. IEEE INFOCOM '93,* San Francisco, pp. 1186-1193, March 30-April 1, 1993.
131. "Stability of a class of dynamic routing protocols (IGRP)" (with S.H. Low), *Proc. IEEE INFOCOM '93,* San Francisco, pp. 610-616, March 30-April 1, 1993.
132. "Increased spectral efficiency through power control," (with A. Goldsmith), *IEEE Internat. Conf. on Communications,* Geneva, pp. 600-604, May 1993.
133. "Real-time task management for communications and control in multi-car platoons," (with K.S. Chang, W. Li and I. Porche), *Proc. Intelligent Vehicles '92,* Detroit, MI, pp. 409-414, June 29-July 1, 1992.
134. "Coordinating vehicles in an automated highway," in A. Bensoussan and J.-P. Verjus, Eds., *Future Tendencies in Computer Science, Control and Applied Mathematics. Proc. International Conference on the Occasion of the 25th Anniversary of INRIA,*Springer-Verlag, pp. 315-330, 1992.
135. "Markets and pricing for interruptible power," (with T.W. Gedra), *IEEE Power Engineering Society 1992 Winter Meeting,* New York, January 26-30, 1992. Also *IEEE Transactions on Power Systems,* Vol. 8, No. 1, pp. 122-128, Feb. 1993.
136. "Chaos in a simple power system," (with H.-D. Chiang, C.-W. Liu, F.F. Wu and M.G. Lauby), *IEEE Power Engineering Society 1992 Winter Meeting,* New York, Paper No. 92/WM/151-1/PWRS, January 26-30, 1992. Also *IEEE Transactions on Power Systems,* Vol. 8, No. 4, pp. 1407-1417, Nov. 1993.
137. "Pricing for system security," (with R.J. Kaye and F.F. Wu), *IEEE Power Engineering Society 1992 Winter Meeting* New York, January 26-30, 1992. Also, *IEEE Transactions on Power Systems,* Vol. 10, No. 1, pp. 575-583, May 1995.

138. "A BCU method for direct analysis of power system transient stability," (with H.D. Chiang and F.F. Wu), Paper No. 91/SM/423-4/PWRS *IEEE Power Engineering Society 1991 Summer Meeting* July 28-August 1, 1991, San Diego, CA. Also *IEEE Transactions on Power Systems,* Vol. 9, No. 3, pp. 1194-1208, August 1994.
139. "The design of platoon maneuvers for IVHS," (with A. Hsu, S. Sachs and F. Eskafi), *Proc. American Control Conference,* Boston, MA, pp. 2545-2550, June 26-28, 1991.
140. "Two proposals to improve traffic flow," (with U. Karaaslan and J. Walrand), *Proc. American Control Conference,* Boston, MA, pp. 2539-2544, June 26-28, 1991.
141. "A preliminary implementation for vehicle platoon control system," (with K.S. Chang, W. Li, A. Shaikhbahai and F. Assaderaghi), *Proc. American Control Conference,* Boston, MA, pp. 3078-3083, June 26-28, 1991.
142. "A model for pricing interruptible electric power service," (with C.-W. Tan), in G.B. Di Masi, A. Gombani and A.B. Kurzhanski, Eds., *Modelling, Estimation and Control of Systems with Uncertainty,* pp. 423-444, Birkhauser, Boston, 1991.
143. "A simple theory of traffic and resource allocation in ATM," (with S. Low), *Proc. IEEE Global Telecommunications Conference,* Phoenix, AZ, Paper 45.5.1, pp. 1633-1637, December 2-5, 1991.
144. "Simulating finitely recursive processes in LISP," (with R.A. Cieslak), *Discrete Event Dynamic Systems: Theory and Applications,* Vol. 1, pp. 373-392, 1992.
145. "Sketch of an IVHS systems architecture," (with S.E. Shladover), *Proc. Vehicle Navigation and Information Systems Conference (VNIS),* Dearborn, MI, pp. 909-922, October 20-23, 1991.
146. "Experimentation with a vehicle platoon control system," (with K.S. Chang, W. Li, P. Devlin, A. Shaikhbahai, J.K. Hedrick, D. MacMahon, V. Narendran and D. Swaroop), *Proc. Vehicle Navigation and Information Systems Conference (VNIS),* Dearborn, MI, pp. 1117-1124, October 20-23, 1991.
147. "Averaged dynamics and large deviations theory for the analysis and synthesis of communication network protocols," (with R.K. Boel and J. Walrand), in J.W. Cohen and C.D. Pack, Eds., *Queueing, Performance and Control in ATM: ITC-13 Workshops, Proceedings of the 13th International Teletraffic Congress,* North-Holland, pp. 225-230, 1991.
148. "Throughput in multiple service, multiple resource communication networks," (with S. Jordan), *IEEE Trans. on Communications,* Vol. 39, No. 8, pp. 1216-1222, August 1991.
149. "A method for specifying, implementing, and verifying media access control protocols," (with C. Brooks and R. Cieslak), *IEEE Control Systems Magazine* Vol. 10, No. 4, pp. 87-94, June 1990.
150. "Undecidability results for deterministic communicating sequential processes," (with R. Cieslak), *IEEE Trans. on Automatic Control,* Vol. 35, No. 9, pp. 1032-1039, September 1990.
151. "Stochastic dynamic optimization, approaches and computation," (with R.J-B. Wets) in: E M. Iri and E K. Tanabe, Eds., *Mathematical Programming: Re-*

cent Developments and Applications, pp. 309-332, Kluwer Academic Publishers, Dordrecht, 1989.

152. “Process models for discrete event models,” in: M.A. Kaashoek et al., Eds., *Realization and Modelling in System Theory: Proc. Internat. Symposium MTNS-89, Vol. 1,* pp. 23-42, Birkhauser, Boston, 1990.
153. “Algebras of discrete event models,” (with K. Inan), *Proceedings of the IEEE,* Vol. 77, No. 1, pp. 24-38, 1989.
154. “The programmable network prototyping system,” (with R. Cieslak, A. Fawaz, S. Sachs, J. Walrand and A. Li), *Computer Magazine,* 67-76, May 1989.
155. “The programmable network prototyping system,” (with R. Cieslak, A. Fawaz, S. Sachs, J. Walrand and A. Li), *Proc. IEEE INFOCOM,* 393-399, April 1989, Ottawa, Canada.
156. “A language for specifying and implementing low-level functions of media access protocols,” (with R. Cieslak), *Proc. IEEE Internat. Phoenix Conf. on Computers and Communications,* pp. 428-433, April 1989, Phoenix, AZ.
157. “On the dynamics of regulated markets, construction standards, energy standards and durable goods: A cautionary tale,” (with J. Quigley) in A. Andersson et al., Eds., *Advances in Spatial Theory and Dynamics,* pp. 263-272, North-Holland, 1989.
158. “Analysis of interrupt handling schemes in real-time systems,” (with A. Fawaz and J. Walrand), *Proc. IEEE Internat. Conf. on Computers and Communications,* Phoenix, AZ, pp. 260-263, April 1989.
159. “Hybrid dynamical systems,” (with A. Gollu), *Proc. 28th IEEE Conference on Decision and Control,* Tampa, FL, pp. 2708-2712, December 1989.
160. “Implementing finitely recursive processes in LISP,” (with R. Cieslak), in E. Arikan, Ed., *Communication, Control, and Signal Processing,* Elsevier, pp. 954-968, 1990.
161. “Throughput in multiple service, multiple resource communication networks,” (with S. Jordan), *Proc. 28th IEEE Conference on Decision and Control,* pp. 236-241, Tampa, FL, December 1989.
162. *Discrete Event Systems: Models and Applications,* (P. Varaiya and A.B. Kurzhanski, Editors), Lecture Notes in Control and Information Sciences, Vol. 103, Springer-Verlag, 1988.
163. “Finitely recursive processes,” (with K. Inan) in *Lecture Notes in Control and Information Sciences,* Vol. 103, Springer-Verlag, 1988.
164. “Foundations of the potential energy boundary surface method for power system transient stability analysis,” (with H.-D. Chiang and F.F. Wu), *IEEE Trans. on Circuits and Systems,* Vol. CAS-35, No. 6, pp. 712-728, 1988.
165. “Degenerate Hopf bifurcations in power systems,” (with R.-L. Chen), *IEEE Trans. on Circuits and Systems,* Vol. CAS-35, No. 7, pp. 818-824, 1988.
166. “Supervisory control of discrete event processes with partial observations,” (with R. Cieslak, C. Desclaux and A. Fawaz), *IEEE Trans. on Automatic Control,* Vol. AC-33, No. 3, pp. 249-260, 1988.
167. “Finitely recursive process models for discrete event systems,” (with K. Inan), *IEEE Trans. on Automatic Control,* Vol. AC-33, No. 7,pp. 626-639, 1988.

168. “Foundations of direct methods for power system transient stability analysis,” (with H-D. Chiang and F.F. Wu), *IEEE Trans. on Circuits and Systems,* Vol. CAS-34, No. 2, pp. 160-173, 1987.
169. “Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays, Part I: IID rewards,” (with V. Anantharam and J. Walrand), *IEEE Trans. on Automatic Control,* Vol. AC-32, No. 1, pp. 968-976, 1987.
170. “Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays, Part II: Markovian rewards,” (with V. Anantharam and J. Walrand), *IEEE Trans. on Automatic Control,* Vol. AC-32, No. 1, pp. 977-982, 1987.
171. “Consensus in distributed estimation,” (with D. Teneketzis), in H.V. Poor, Ed., *Advances in Statistical Signal Processing,*Vol. 1, JAI Press, 1987.
172. “Innovation and changes in regional structure,” (with E.J. Malecki), in P. Nijkamp, Ed., *Handbook of Regional and Urban Economics,* Elsevier Science Publishers, pp. 629-645, 1986.
173. “Scheduling jobs with stochastically ordered processing times on parallel machines to minimize expected flowtimes,” (with R.R. Weber and J. Walrand), *J. Appl. Prob.,* Vol. 23, 1986, 841-847.
174. “An optimal strategy for a conflict resolution problem,” (with V. Anantharam), *Systems and Control Letters* Vol. 7, pp. 329-332, 1986.
175. “Direct methods for transient stability analysis of power systems: recent results” (with F.F. Wu and R.-L. Chen), *Proc. of the IEEE,* Vol. 73, No. 12, pp. 1703-1715, December 1985.
176. “Unstable queues with arrival rate less than service rate,” (with V. Anantharam and S. Parekh), *Proc. IEEE Conference on Decision and Control, Fort Lauderdale,* pp. 1347-1348, December 1985.
177. “Theory of the potential energy boundary surface,” (with H-D. Chiang, F.F. Wu and C-W. Tan), *Proc. IEEE Conference on Decision and Control, Fort Lauderdale,* pp. 49-51, December 1985.
178. “A structure preserving energy function for power system transient stability analysis,” (with N. Tsolas and A. Arapostathis), *IEEE Trans. on Circuits and Systems,* Vol. CAS-32, No. 10, pp. 1041-1049, October 1985.
179. “Consensus in distributed estimation with inconsistent beliefs,” (with D. Teneketzis), *Systems & Control Letters* Vol. 4, pp. 217-221, 1984.
180. “Consensus in distributed estimation,” in A. Bagchi and H.Th. Jongen, Eds., *Lecture Notes in Control and Information Sciences,* Vol. 66, Springer-Verlag, pp. 182-191, 1985.
181. “Optimal resource allocation for two processes,” (with C. Courcoubetis), *AT&T Technical Journal* Vol. 64, No. 1, pp. 1-14, January 1985.
182. “A structure preserving energy function for power system transient stability analysis,” (with N. Tsolas and A. Arapostathis), *IEEE Trans. on Circuits and Systems,* Vol. CAS-32, No. 10, pp. 1041-1049, October 1985.
183. “The $c\mu$ rule revisited,” (with C. Buyukkoc and J. Walrand), *Advances in Applied Probability,* Vol. 17, No. 1, pp. 237-238, March 1985.

184. "Extensions of the multi-armed bandit problem," (with J. Walrand and C. Buyukkoc), *IEEE Trans. Automatic Control,* Vol. AC-30, No. 5, pp. 426-439, May 1985.
185. "Bifurcation models of urban development," (with M. Wiseman), in Ake E. Andersson, Walter Isard, Tonu Puu, Eds., *Regional and Industrial Development Theories, Models and Empirical Evidence,* Amsterdam: Elsevier Science Publishers B.V. (North-Holland), pp. 61-88, 1984.
186. "Arnold diffusion in the swing equations of a power system," (with F.M.A. Salam and J.E. Marsden), *IEEE Trans. Circuits and Systems,* Vol. CAS-31, No. 7, pp. 673-688, August 1984.
187. "Multi-armed bandit problems and resource sharing systems," (with J. Walrand), in G. Iazeolla, P.J. Courtois, and A. Hordijk, Eds., *Mathematical Computer Performance and Reliability,* Amsterdam: North-Holland, pp. 181-196, 1984.
188. "Blocking probability in a switching center with arbitrary routing policy," (with B. Gopinath, J.-M. Garcia), *AT&T Bell Laboratories Technical Journal,* Vol. 63, No. 5, May-June 1984, pp. 709-720.
189. "Nonlinear oscillations in power systems," (with E.H. Abed), *Electrical Power and Energy Systems,* Vol. 6, No. 1, pp. 37-43, January 1984.
190. "The decentralized quickest detection problem," (with D. Teneketzis), *IEEE Trans. Automatic Control,* Vol. AC-29, No. 7, pp. 641-644, July 1984.
191. "Invariance in resource sharing problems," (with C. Courcoubetis and J. Walrand), *Journal of Applied Probability,* Vol. 21, pp. 777-785, 1984.
192. "The behavior of three node power networks," (with A. Arapostathis), *Electrical Power and Energy Systems,* Vol. 5, No. 1, pp. 22-30, January 1983.
193. "Reindustrialization and the outlook for declining cities," (with M. Wiseman), in J. V. Henderson (Ed.), *Research in Urban Economics,* Vol. 3, JAI Press, Greenwich CT and London, pp. 167-190, 1983.
194. "A game-theoretic view of two processes using a single resource," (with C. Courcoubetis), *IEEE Trans. Automatic Control,* Vol. AC-28, No. 11, pp. 1059-1061, November 1983.
195. "Serving process with least thinking time maximizes resource utilization," (with C. Courcoubetis), *IEEE Trans. Automatic Control,* Vol. AC-29, No. 11, pp. 1005-1008, November 1984.
196. "Invariance in resource sharing problems," (with C. Courcoubetis and J. Walrand), *Proc. 21st IEEE Conf. on Decision and Control,* Orlando, pp. 861-863, December 1982.
197. "Optimal causal coding-decoding problems," (with J. Walrand), *IEEE Trans. Information Theory,* Vol. IT-29, No. 6, pp. 814-820, November 1983.
198. "Causal coding and control of Markov chains," (with J. Walrand), *Systems and Control Letters,* Vol. 3, pp. 189-192, September 1983.
199. "Study of nonlinear oscillations due to exciter control using Hopf bifurcation," (with E. Abed and N. Tsolas), *Proc. 1983 IEEE Internat. Symposium on Circuits and Systems,* Newport Beach, pp. 1410-1413, May 1983.

200. “Study of nonlinear oscillations due to exciter control,” (with E. Abed and N. Tsolas), *Proc. 1983 American Control Conference,* San Francisco, pp. 674-678, June 1983.
201. “Chaos and Arnold diffusion in dynamical systems,” (with F.M.A. Salam and J. Marsden), *IEEE Trans. Circuits and Systems,* Vol. CAS-30, No. 9, pp. 697-708, September 1983.
202. “Analytic expressions for the unstable manifold at equilibrium points in dynamical systems of differential equations,” (with F.M.A. Salam and A. Arapostathis), *Proc. 22nd IEEE Conf. on Decision and Control,* San Antonio, TX, Dec. 14-16, pp. 1389-1392, 1983.
203. “Optimal control of service in tandem queues,” (with Z. Rosberg and J.C. Walrand), *IEEE Trans. Automatic Control,* Vol. AC-27, No. 3, pp. 600-610, June 1982.
204. “Asymptotic agreement in distributed estimation,” (with V. Borkar), *IEEE Trans. Automatic Control,* Vol. AC-27, No. 3, pp. 650-656, June 1982.
205. “Analysis of the power flow equation,” (with A. Arapostathis and S. Sastry), *Electrical Power and Energy Systems,* Vol. 3, No. 3, pp. 115-126, July 1981.
206. “Jump behavior of circuits and systems,” (with S. Sastry and C. Desoer), *Proc. 20th IEEE Conf. on Decision & Control,* San Diego, pp. 1043-1047, December 1982.
207. “Routing traffic in telephone networks,” (with J-C. Garcia and B. Gopinath), *Proc. 20th IEEE Conf. on Decision & Control,* San Diego, pp. 386-389, December 1982.
208. “Oscillations in power systems via Hopf bifurcation,” (with E. Abed), *Proc. 20th IEEE Conf. on Decision & Control,* San Diego, pp. 962-992, December 1982.
209. “Global analysis of swing dynamics,” (with A. Arapostathis and S. Sastry), *IEEE Trans. Circuits and Systems,* Vol. CAS-29, No. 10, pp. 673-679, October 1982.
210. “Identification and adaptive control of Markov chains,” (with V. Borkar), *SIAM J. Control and Optimization,* Vol. 20, No. 4, pp. 470-489, July 1982.
211. “Hierarchical stability and alert state steering control of interconnected power systems,” (with S. Sastry), *IEEE Trans. on Circuits and Systems,* Vol. CAS-27, No. 11, pp. 1102-1112, November 1980.
212. “Flows in queuing networks: A martingale approach,” (with J. Walrand), *Mathematics of Operations Research,* Vol. 6, No. 3, pp. 387-404, August 1981.
213. “Coherency for interconnected power systems,” (with S. Sastry), *IEEE Trans. on Automatic Control,* Vol. AC-26, No. 1, pp. 218-226, February 1981.
214. “Finite chain approximation to a continuous stochastic control problem,” (with V. Borkar), *IEEE Trans. on Automatic Control,* Vol. AC-26, No. 2, pp. 466-470, April 1981.
215. “Investment and employment in manufacturing in US metropolitan areas 1960-1976,” *Regional Science and Urban Economics,* Vol. 11, No. 4, pp. 431-469, November 1981.
216. “A simple dynamic routing problem,” (with A. Ephremides and J. Walrand), *IEEE Trans. on Automatic Control,* Vol. AC-25, No. 4, pp. 690-693, August 1980.

217. "Interconnections of Markov chains and quasi-reversible queuing networks," (with J. Walrand), *Stochastic Processes and their Applications,* Vol. 10, pp. 209-219, 1980.
218. "Sojourn times and the overtaking condition in Jacksonian networks," (with J. Walrand), *Advances in Applied Probability,* Vol. 12, pp. 1000-1018, 1980.
219. "A minimum principle for decentralized stochastic control problems," (with J. Walrand), in *Dynamic Optimization and Mathematical Economics,* P.T. Liu, Ed., Plenum Press, pp. 253-266, 1980.
220. "Division of labor and the distribution of income," (with R. Artle and C. Humes), *Regional Science and Urban Economics,* Vol. 9, pp. 71-82, 1979.
221. "Decentralized control in packet switched satellite communication," (with J. Walrand), in *IEEE Trans. Automatic Control,* Vol. AC-24, No. 5, pp. 794-796, 1979.
222. "Adaptive control of Markov chains," (with V. Borkar), in M. Kohlmann and W. Vogel, Eds., *Stochastic Control Theory and Stochastic Differential Systems,* Lecture Notes in Control and Information Sciences, No. 16, Springer-Verlag, Berlin, 1979.
223. "Adaptive control of Markov chains, I: Finite parameter set," (with V. Borkar), *IEEE Trans. Automatic Control,* Vol. AC-24, No. 6, pp. 953-957, 1979.
224. "Multilayer control of large Markov chains" (with J. P. Forestier), *IEEE Trans. Automatic Control,* Vol. AC-23, No. 2, pp. 298-305, 1978.
225. "The effects of discounted cost on the uncertainty threshold principle" (with R. Ku and M. Athans), *Ann. Econ. and Soc. Measurement,* Vol. 6, No. 5, pp. 485-492, 1978.
226. "Optimal and suboptimal stationary controls for Markov chains," *IEEE Trans. Automatic Control,* Vol. AC-23, No. 3, pp. 388-394, 1978.
227. "On delayed sharing patterns" (with J. Walrand), *IEEE Trans. Automatic Control,* Vol. AC-23, No. 3, pp. 443-445, 1978.
228. "Stochastic control of large Markov chains," (with J. P. Forestier), *Control and Management of Integrated Industrial Complexes,* M. Singh and A. Titli, Eds., Pergamon Press, Oxford, 1978.
229. "Decentralized stochastic control," (with J. Walrand), *Control and Management of Integrated Industrial Complexes,* M. Singh and A. Titli, Eds., Pergamon Press, Oxford, 1978.
230. "The age of cities and movement of manufacturing employment in the U.S.," (with M. Wiseman), *Papers of Regional Science Assoc.,* Vol. 41, pp. 127-140, 1978.
231. "Life cycle consumption and homeownership," (with R. Artle), *J. Economic Theory,* Vol. 18, No. 9, pp. 38-58, 1978.
232. "Survey of decentralized control methods for large scale systems" (with N. Sandell, M. Athans and M. Safanov), *IEEE Trans. Automatic Control,* Vol. AC-23, No. 2, pp. 108-128, 1978.
233. "The spatial structure of production with a Leontief technology," (with U. Schweizer), *Regl. Science and Urban Econ.,* Vol. 6, pp. 231-251, 1976.

234. “Optimal control of jump processes,” (with R. Boel), *SIAM J. Control and Opt.,* Vol. 15, No. 1, pp. 92-119, 1977.
235. “The spatial structure of production with a Leontief technology, II: Substitute techniques,” (with U. Schweizer), *Regional Science and Urban Econ.,* Vol. 7, pp. 293-320, 1977.
236. “Division of labor—Simon revisited,” (with R. Artle and C. Humes), *Regional Science and Urban Econ.,* Vol. 7, pp. 185-196, 1977.
237. “A class of Markovian problems related to the districting problem for urban emergency services,” (with U. Schweizer and J. Hartwick), *Ricerche de Automatica,* Vol. 8, No. 1, pp. 1-19, 1977.
238. “Economics of information: Examples in location and land-use theory,” (with K. Stahl), *Regional Science and Urban Econ.,* Vol. 8, pp. 43-56, 1977.
239. “N-Player stochastic differential games,” *SIAM J. Control and Optimization,* Vol. 14, No. 3, pp. 538-545, May 1976.
240. “General equilibrium and location theory,” (with J. Hartwick and U. Schweizer), *J. Urban Economics,* Vol. 3, pp. 285-303, 1976.
241. “Public goods and consumption indivisibility,” (with Urban Systems Group), *Regional Science and Urban Econ.,* Vol. 6, pp. 45-50, 1976.
242. “On the design of rent control,” *IEEE Trans. Automatic Control,* Vol. AC-21, No. 3, pp. 316-319, 1976.
243. “Stochastic models of price adjustment” (with S. Barta), *Annals of Economic and Social Measurement,* Vol. 5, No. 3, pp. 267-281, 1976.
244. “Comparative statics of a residential economy with several classes,” (with J. Hartwick and U. Schweizer), *J. Econ. Theory,* Vol. 13, No. 3, pp. 396-413, 1976.
245. “On the solutions of a stochastic control system, II” (with T.E. Duncan), *SIAM J. Control,* Vol. 13, No. 5, pp. 1077-1092, 1975.
246. “Martingales on jump processes, Part I: Representation results,” (with R. Boel and E. Wong), *SIAM J. Control,* Vol. 13, No. 5, pp. 992-1021, 1975.
247. “Martingales on jump processes, Part II: Applications,” (with R. Boel and E. Wong), *SIAM J. Control,* Vol. 13, No. 5, pp. 1022-1061, 1975.
248. “The martingale theory of jump processes,” *IEEE Trans. Automatic Control,* Vol. AC-20, No. 1, pp. 34-42, 1975.
249. “Economic theories and empirical models of location choice and land use: A survey,” (with R. Artle), *Proc. of the IEEE,* Vol. 63, No. 3, pp. 421-430, 1975.
250. “N-Person stochastic differential games,” in *The Theory and Application of Differential Games,* J.D. Grote and D. Reidel, Eds., Dordrecht and Boston, pp. 97-106, 1975.
251. “Filtering and control of jump processes,” in *Control Theory, Numerical Methods and Computer Systems Modeling,* A. Bensoussan and J.L. Lions, Eds., Lecture Notes in Economics and Mathematical Systems, Vol. 107, Springer-Verlag, Berlin, pp. 509-518, 1975.
252. “An optimizing model of urban development,” (with M. Ripper), *Environment and Planning A,* Vol. 6, pp. 149-168, 1974.
253. “On the multiplicity of an increasing family of sigma-fields” (with M.H.A. Davis), *Annals of Probability,* Vol. 2, No. 5, pp. 958-963, 1974.

254. "Effects of congestion on the shape of a city," (with L. Legey and M. Ripper), *Journal of Economic Theory,* Vol. 6, No. 2, pp. 162-179, April 1973.
255. "Dynamic programming conditions for partially observed stochastic systems," (with M.H.A. Davis), *SIAM J. Control,* Vol. 11, No. 2, pp. 226-261, May 1973.
256. "Optimal control of a partially observable stochastic system," *Proc. Amer. Math. Society,* McKean, Ed.: pp. 173-187, 1973.
257. "Information states for linear stochastic systems," (with M.H.A. Davis), *J. Math. Anal. Appl.,* Vol. 37, No. 2, pp. 384-402, February 1972.
258. "A critique of Forrester's model of an urban area," (with J.N. Gray and D. Pessel), *IEEE Trans. on Systems, Man and Cybernetics,* Vol. SMC-2, No. 2, pp. 139-144, April 1972.
259. "Decentralized information and control: A network flow example," (with R. Lau and R.C.M. Persiano), *IEEE Trans. on Automatic Control,* Vol. AC-17, No. 4, pp. 466-473, August 1972.
260. "Locational implications of transaction costs," (with R. Artle), *Swedish J. Economics,* Vol. 74, pp. 174-183, 1972.
261. "Trends in the theory of control of large systems," *Annals of Economic and Social Measurement,* Vol. 1, No. 4, pp. 493-500, 1972.
262. "Equivalent nonlinear reciprocal networks," (with J.P. Verma), *IEEE Trans. on Circuit Theory,* Vol. CT-18, No. 2, pp. 214-217, March 1971.
263. "On the solutions of a stochastic control system," (with T.E. Duncan), *SIAM J. Control,* Vol. 9, No. 3, pp. 354-371, August 1971.
264. "Target Function approach to linear pursuit problems," (with W. Borgest), *IEEE Trans. on Automatic Control,* Vol. AC-16, No. 5, pp. 449-459, October 1971.
265. "Complexity problems in real time languages," (with W. A. Burkhard), *Information Sciences,* No. 3, pp. 87-199, 1971.
266. "Differential games with dynamical systems," in *Differential Games* H.W. Kuhn and G.P. Szego, Eds., North-Holland, Amsterdam and London, pp. 129-146, 1971.
267. "Differential games," *Proc. Sixth Berkeley Symposium on Mathematical Statistics and Probability,* Vol. III: Probability Theory, UC Press, pp. 687-697, 1971.
268. "N-Person, nonzero sum differential games with linear dynamics," *SIAM J. Control,* Vol. 8, No. 4, pp. 441-449, November 1970.
269. "Existence of saddle points in differential games," (with J. Lin), *SIAM J. Control,* Vol. 7, No. 1, pp. 141-157, February 1969.
270. "E-Capacity of classes of unknown channels," (with L. Forys), *Information and Control,* Vol. 14, No. 4, pp. 376-406, November 1969.
271. "Decomposition of large-scale systems," *System Theory,* L. Zadeh and E. Polak, Eds., McGraw-Hill, pp. 467-488, 1969.
272. "Representation of a differential system," (with D. Eggert), *J. Differential Equations,* Vol. 4, No. 2, pp. 280-299, April 1968.
273. "Difference between number of capacitors and inductors in a lossless n-port," (with L. Silverman), *IEEE Trans. on Circuit Theory,* Vol. CT-15, No. 2, pp. 147-148, June 1968.

274. "Capacity of classes of Gaussian channels," (with W.L. Root), *SIAM J. Applied Math.,* Vol. 16, No. 6, pp. 1350-1393, November 1968.
275. "On the existence of solutions to a differential game," *SIAM J. Control,* Vol. 5, No. 1, pp. 153-162, February 1967.
276. "Nonlinear programming in Banach space," *SIAM J. Applied Math.,* Vol. 15, No. 2, pp. 284-293, March 1967.
277. "Determinable classes of signals and linear channels," *SIAM J. Applied Math.,* Vol. 15, No. 2, pp. 440-449, March 1967.
278. "Minimal realization of a non-anticipative impulse-response matrix," (with C.A. Desoer), *SIAM J. Applied Math.,* Vol. 15, No. 3, pp. 754-764, May 1967.
279. "B.I./B.O. Stability of discrete time systems," (with J. Lin), *IEEE Trans. on Automatic Control,* Vol. AC-12, No. 4, pp. 423-427, August 1967.
280. "Affine dynamical systems," (with D. Eggert), *J. Computer and System Sciences,* Vol. 1, No. 4, pp. 330-348, December 1967.
281. "Contollability of a class of nonlinear systems," (with L. Silverman and E. Davidson), *IEEE Trans. on Automatic Control,* Vol. AC-12, No. 6, pp. 791-792, December 1967.
282. "On the trajectories of a differential system," *Math. Theory of Control,* L.W. Neustadt and A.V. Balakrishnan, Eds., Academic Press, pp. 115-128, 1967.
283. "Decomposition of large-scale system,," *SIAM J. Control,* Vol. 4, No. 1, pp. 173-178, February 1966.
284. "B.I./B.O. Stability of the Lure problem," (with R. Liu), *IEEE Trans. on Automatic Control,* Vol. AC-11, No. 4, pp. 745-746, October 1966.
285. "B.I./B.O. Stability of nonlinear, time-varying system," (with R. Liu), *SIAM J. Control,* Vol. 4, No. 4, pp. 698-704, November 1966.
286. "Normal form and stability of a class of coupled nonlinear networks," (with R. Liu), *IEEE Trans. on Circuit Theory,* Vol. CT-13, No. 4, pp. 413-418, December 1966.
287. "A decomposition technique for nonlinear programming," *I.B.M. Research, Report RJ-345,* July 1965.

List of Contributors and Speakers

Karl J. Astrom
Lund Institute of Technology
SE-221 00 Lund, Sweden
kja@control.lth.se

John S. Baras
University of Maryland
College Park, MD 20742 USA
baras@isr.umd.edu

Vivek S. Borkar
Tata Institute of Fundamental Research
Mumbai 400 005 India
borkar@tifr.res.in

Mark H.A. Davis
Imperial College London
London SW7 2AZ, UK
mark.davis@imperial.ac.uk

Akash R. Deshpande
Teja Technologies, Inc.
San Jose, CA 95113 USA
akash@teja.com

Leonard J. Forys
The Forys Consulting Group
Holmdel, NJ 07733 USA
Lforys@aol.com

Dinesh Garg
Indian Institute of Science
Bangalore 560 012 India
dgarg@csa.iisc.ernet.in

Michael Gastpar
University of California at Berkeley
Berkeley, CA 94720 USA
gastpar@berkeley.edu

Andrea Goldsmith
Stanford University
Stanford, CA 94305 USA
andrea@wsl.stanford.edu

Rajarshi Gupta
University of California at Berkeley
Berkeley, CA 94720 USA
guptar@eecs.berkeley.edu

Roberto Horowitz
University of California at Berkeley
Berkeley, CA 94720 USA
horowitz@me.berkeley.edu

Inseok Hwang
Purdue University
West Lafayette, IN 47907 USA
ihwang@purdue.edu

Tao Jiang
University of Maryland
College Park, MD 20742 USA
tjiang@glue.umd.edu

Ramesh Johari
Stanford University
Stanford, CA 94305 USA
ramesh.johari@stanford.edu

Apostolos Kotsialos
Technical University of Crete
73100 Chania, Greece
appie@dssl.tuc.gr

Sri Kumar
DARPA
Arlington, VA 22203 USA
skumar@darpa.mil

Alexander B. Kurzhanski
University of California at Berkeley
Berkeley, CA 94720 USA
kurzhans@eecs.berkeley.edu

Edward A. Lee
University of California at Berkeley
Berkeley, CA 94720 USA
eal@eecs.berkeley.edu

Luiz Fernando L. Legey
Federal University of Rio de Janeiro
21941-972 Rio de Janeiro RJ, Brazil
legey@ppe.ufrj.br

Xiangheng Liu
Stanford University
Stanford, CA 94305 USA
liuxh@wsl.stanford.edu

Hani S. Mahmassani
University of Maryland
College Park, MD 20742 USA
masmah@umd.edu

D. Manjunath
Indian Institute of Technology
Mumbai 400 076 India
dmanju@ee.iitb.ac.in

Bud Mishra
New York University
New York, 10012 USA
mishra@cs.nyu.edu

Sanjoy Mitter
Massachusetts Institute of Technology
Cambridge, MA 02139 USA
mitter@mit.edu

Laura Muñoz
University of California at Berkeley
Berkeley, CA 94720 USA
lmunoz@me.berkeley.edu

Markos Papageorgiou
Technical University of Crete
73100 Chania, Greece
markos@dssl.tuc.gr

Carla Piazza
University of Udine
Udine, Italy
piazza@dimi.uniud.it

S. Shankar Sastry
University of California at Berkeley
Berkeley, CA 94720 USA
sastry@eecs.berkeley.edu

Steven E. Shladover
California PATH Program
Richmond, CA 94804 USA
steve@path.berkeley.edu

Joseph Sifakis
VERIMAG Laboratory
38610 Gieres, France
joseph.sifakis@imag.fr

Dušan M. Stipanović
University of Illinois
Urbana, IL 61801 USA
dusan@uiuc.edu

Tudor Mihai Stoenescu
California Institute of Technology
Pasadena, CA 91125 USA
tudor@caltech.edu

Xiaotian Sun
University of California at Berkeley
Berkeley, CA 94720 USA
sunx@me.berkeley.edu

Demosthenis Teneketzis
University of Michigan
Ann Arbor, Michigan 48109 USA
teneketzis@eecs.umich.edu

Claire J. Tomlin
Stanford University
Stanford, CA 94305 USA
tomlin@stanford.edu

John N. Tsitsiklis
Massachusetts Institute of Technology
Cambridge, MA 02139 USA
jnt@mit.edu

Hal Varian
University of California at Berkeley
Berkeley, CA 94720 USA
hal@sims.berkeley.edu

Jean Walrand
University of California at Berkeley
Berkeley, CA 94720 USA
wlr@eecs.berkeley.edu

Felix F. Wu
University of California at Berkeley
Berkeley, CA 94720 USA
ffwu@eecs.berkeley.edu

Xuesong Zhou
University of Maryland
College Park, MD 20742 USA
xszhou@wam.umd.edu

Part I

Hybrid Systems

1

Polytopic Approximations of Reachable Sets Applied to Linear Dynamic Games and a Class of Nonlinear Systems*

Inseok Hwang[1], Dušan M. Stipanović[2], and Claire J. Tomlin[3]

[1] Assistant Professor, School of Aeronautics and Astronautics, Purdue University
ihwang@purdue.edu

[2] Assistant Professor, Department of General Engineering, and Coordinated Science Laboratory, University of Illinois at Urbana-Champaign
dusan@uiuc.edu

[3] Associate Professor, Department of Aeronautics and Astronautics, Stanford University
tomlin@stanford.edu

Summary. This chapter presents applications of polytopic approximation methods for reachable set computation using dynamic optimization. The problem of computing exact reachable sets can be formulated in terms of a Hamilton–Jacobi partial differential equation (PDE). Numerical solutions which provide convergent approximations of this PDE have computational complexity which is exponential in the continuous variable dimension. Using dynamic optimization and polytopic approximation, computationally efficient algorithms for overapproximative reachability analysis have been developed for linear dynamical systems [1]. In this chapter, we extend these to feedback linearizable nonlinear systems, linear dynamic games, and norm-bounded nonlinear systems. Three illustrative examples are presented.

1.1 Introduction

Reachability analysis for continuous and hybrid systems is important for the automatic verification of safety properties and for the synthesis of safe controllers for these systems [2, 3]. Convergent approximations of reachable sets for such systems can be computed by solving a particular Hamilton–Jacobi partial differential equation (PDE) [3, 4]. Numerical methods have been devised to compute these convergent overapproximations [5], which work well in up to four to five continuous variable dimensions, yet these methods are not practical for solving high dimensional problems. Therefore, approximate methods for reachable set computation have been proposed.

Tiwari and Khanna [6] and Alur et al. [7] proposed predicate abstraction for reachable set computation: this method can be used to extract equivalent finite state

* This research was supported by DARPA under the Software Enabled Control Program (AFRL contract F33615-99-C-3014), by ONR under MURI contract N00014-02-1-0720, and by an NSF Career Award (ECS-9985072).

models from complex, infinite state models, which are used to find approximate reachable sets of the original systems. In [8], Hwang et al. have used an augmented form of predicate abstraction to compute reachable sets for a simple biological cell network. However, since the accuracy of reachability analysis using predicate abstraction greatly depends on the choice of polynomials for abstraction, it is important to have information about a given system *a priori* (from analysis and simulations) to get good results in the reachability analysis. Chutinan and Krogh [9, 10] present a method to approximate the flows of autonomous systems with convex polyhedra. An experimental system called **d/dt** [7, 11, 12] has been developed to approximate reachable sets for linear dynamical systems using griddy orthogonal polyhedra. Ideas based on projecting the initial or target set into a lower dimensional subset of the state space, performing the reach set computation in the lower dimensional space, and then back projecting to form an overapproximation of the actual reachable set in the full state space, are presented in [13, 14]. In all of these methods, however, it is difficult to compute the control input which is guaranteed to keep the system on the boundary or inside the set, from the boundary of the overapproximative set.

Varaiya [1] has designed, using techniques from optimal control theory, a polytopic approximation for linear systems. Kostousova [15] has developed two-sided approximations of reachable sets for linear dynamic systems using parallelotopes. Kurzhanski and Varaiya [16, 17] proposed an ellipsoidal approximation for forward and backward reachable sets (a computational tool VeriSHIFT [18] has been developed based on their ideas) and in [19, 20], they define various types of reachable sets for linear time-varying systems with bounded perturbations using both open and closed-loop input laws. In [20], they propose ellipsoidal overapproximations of reachable sets for linear systems under uncertainty via solutions of a particular type of differential equation. In [21, 22], the authors have extended reachable set computations to general nonlinear systems with state constraints and obstacles, using nonstandard Hamilton–Jacobi equations and variational inequalities. Overall, this seminal work in exact and approximate reachable set calculation suggests new research directions in computational methods for such problems. This work was indeed motivation for the current chapter.

In this chapter, we review the method proposed by Varaiya [1] to compute reachable sets for linear time invariant systems. Inspired by Kurzhanski and Varaiya [16, 17, 19, 20] and by the work of Khrustalev [23], we compute approximate reachable sets for feedback linearizable nonlinear systems, linear dynamic games, and norm-bounded nonlinear systems. We present three examples, one of which is a two-aircraft three-dimensional collision avoidance example which we have used in other work [5].

This chapter is organized as follows. Motivation for this study is described in Section 1.2. Computations of polytopic reachable sets for linear dynamical systems, feedback linearizable nonlinear systems, linear dynamic games, and norm-bounded nonlinear systems are presented in Section 1.3. Examples are presented in Section 1.4. Conclusions are presented in Section 1.5.

1.2 Background and Motivation

Consider a dynamical system,

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t), d(t)), \\ x(0) \in X_0 \quad &(\text{or } x(t_f) \in \mathcal{Y}_0), \;\; t \in [0, t_f] \end{aligned} \tag{1.1}$$

where $0 \leq t_f < \infty$, $x \in \mathbb{R}^n$, $u \in U \subset \mathbb{R}^m$ is the control input, $d \in D \subset \mathbb{R}^p$ is the disturbance input, $X_0 = \{x : l(x) \leq 0\}$ is an initial set of states, and $\mathcal{Y}_0 = \{x : y(x) \leq 0\}$ is a target set of states. We assume f to be Lipschitz. The spaces of admissible control input trajectories and disturbance input trajectories are denoted as the spaces of piecewise continuous functions $\mathcal{U} = \{u(\cdot) \in PC^0 | u(t) \in U, 0 \leq t \leq t_f\}$ and $\mathcal{D} = \{d(\cdot) \in PC^0 | d(t) \in D, 0 \leq t \leq t_f\}$ respectively. The forward and the backward reachable sets of the system (1.1) are defined as follows.

Definition 1. *The forward reachable set $\mathcal{X}(\tau)$ at time τ $(0 < \tau \leq t_f)$, of the system (1.1) from the initial set X_0, is the set of all states $x(\tau)$, such that there exists a control input $u(t) \in \mathcal{U}$ $(0 \leq t \leq \tau)$, for all disturbance inputs $d(t) \in \mathcal{D}$ $(0 \leq t \leq \tau)$, for which $x(\tau)$ is reachable from some $x(0) \in X(0)$, along a trajectory satisfying (1.1).*

Definition 2. *The backward reachable set $\mathcal{Y}(\tau)$ at time τ $(0 \leq \tau < t_f)$, of the system (1.1) from the target set $\mathcal{Y}_0$, is the set of all states $x(\tau)$, such that there exists a control input $u(t) \in \mathcal{U}$ $(\tau \leq t \leq t_f)$, for all disturbance inputs $d(t) \in \mathcal{D}$ $(\tau \leq t \leq t_f)$, for which some $x(t_f) \in \mathcal{Y}_0$ are reachable from $x(\tau)$, along a trajectory satisfying (1.1).*

It has been shown that a forward reachable set computation can be formulated as a dynamic optimization problem [17, 23]. The forward reachable set of the dynamical system (1.1) at time τ $(0 < \tau \leq t_f)$ is shown to be [17]:

$$\mathcal{X}(\tau) = \{x : v(x, \tau) \leq 0\} \tag{1.2}$$

where $v(x, \tau)$ is a (viscosity) solution of the Hamilton–Jacobi–Isaacs (HJI) partial differential equation,

$$D_t v(x, t) + \max_{u \in U} \min_{d \in D} \{< D_x v(x, t), f(x, u, d) >\} = 0 \tag{1.3}$$

with $v(x, 0) = l(x)$, $< p, q >= p^T q$ the inner product in $\mathbb{R}^n$, and where $D_{()}$ represents the partial derivative with respect to the subscripted variable. Thus, the forward reachable set of the dynamical system (1.1) is the zero sublevel set of the solution to the HJI equation in (1.3).

Similarly, the backward reachable set of the dynamical system (1.1) at time τ $(0 \leq \tau < t_f)$ is the zero sublevel set of the solution to the HJI equation [17],

$$D_t v(x, t) + \min_{u \in U} \max_{d \in D} \{< D_x v(x, t), f(x, u, d) >\} = 0 \tag{1.4}$$

with $v(x, t_f) = y(x)$.

In [4, 5], a numerical tool for computing convergent approximations for backwards reachable sets is designed and presented. This method is based on the level set method for computing solutions to PDEs [24]. The computational complexity of this tool is exponential in the number of continuous variables dimensions: it has been shown to work well in up to four or five continuous variables dimensions, yet for larger problems computation time is currently prohibitive. Numerical convergence has been demonstrated on several examples; we will use a "benchmark" three-dimensional example from [5] in this chapter.

Consider planar kinematic models of two aircraft, labeled 1 and 2. Let the relative position and orientation of aircraft 2 with respect to aircraft 1 be represented by $(x_r, y_r, \psi_r) \in \mathbb{R}^2 \times [-\pi, \pi)$. Given the absolute positions and orientations of the two aircraft, denoted as x_i, y_i, ψ_i for $i = 1, 2$, the relative coordinates are defined as: $x_r = \cos\psi_1(x_2 - x_1) + \sin\psi_1(y_2 - y_1), y_r = -\sin\psi_1(x_2 - x_1) + \cos\psi_1(y_2 - y_1), \psi_r = \psi_2 - \psi_1$. The relative kinematics are thus given by:

$$\begin{aligned} \dot{x}_r &= -\sigma_1 + \sigma_2 \cos\psi_r + \omega_1 y_r, \\ \dot{y}_r &= \sigma_2 \sin\psi_r - \omega_1 x_r, \\ \dot{\psi}_r &= \omega_2 - \omega_1, \end{aligned} \tag{1.5}$$

where σ_i is the linear velocity of aircraft i and ω_i is its angular velocity. Safety is encoded as a 5 nautical mile radius cylinder "protected zone" centered at the origin of the relative frame. In this chapter, following the notation in Definition 2 (which is different from that in [5]), we define the angular velocity of aircraft 2 (ω_2) as the control input that steers the system (1.5) into the target set and the angular velocity of aircraft 1 (ω_1) as the disturbance input that keeps the system (1.5) outside of the target set. Posing this problem as a game, we label aircraft 1 as "evader" and aircraft 2 as "pursuer", and we compute the set of states (x_r, y_r, ψ_r) for which for all possible disturbance inputs, ω_1 action of the evader, there is a control input, ω_2 action of the pursuer, such that the system state enters the protected zone, which we consider the target set of the game. For values $\sigma_1 = \sigma_2 = 5$ and $\omega_i \in [-1, 1]$ ($i \in \{1, 2\}$), the problem has been solved numerically, and the results (solid surface) are shown in Fig. 1.4 (Courtesy of I. Mitchell [5]). This computation took approximately 4 minutes to run on a Sun UltraSparc II, in which 50 grid nodes in each dimension were used.

A version of this example may also be solved analytically [25], and it may be verified using this that the average error in computation is less than one tenth of a grid cell, with maximum error always less than one grid cell.

In the following section, we extend Varaiya's method [1] to treat this kind of system and in Section 1.4, we compare the computation above with the resulting approximation.

1.3 Computation of polytopic reachable sets

We first define the overapproximate reachable set [17] (here we specialize to the case of (1.1) in which there are no disturbances). Assume that $x_*(0) \in X_0$

and $u_*(t) \in \mathcal{U}$ for all $t \geq 0$ such that $x_*(\tau) \in \mathcal{X}(\tau)$ $(0 \leq t \leq \tau)$. Then, an overapproximate solution to the solution of the HJI equation in (1.3) is defined as a function $v^+(x,t)$ satisfying [17, 23]:

$$\begin{array}{l} \frac{dv^+(x,t)}{dt}|_{x=x_*(t),u=u_*(t),\dot{x}=f(x,u)} \\ = D_t v^+(x_*,t) + < D_x v^+(x_*,t), f(x_*,u_*) > \\ \leq D_t v^+(x_*,t) + \max_{u \in U}\{< D_x v^+(x_*,t), f(x_*,u) >\} \\ \leq \mu(t) \end{array} \tag{1.6}$$

where $v^+(x_*,t)$ is a piecewise continuous function, and $\mu(t)$ is a positive-definite, integrable function. By integrating (1.6) from 0 to τ, we obtain an overapproximative reachable set of the dynamical system (1.1) at time τ as:

$$V^+(\tau) = \{x|v^+(x,\tau) \leq \int_0^\tau \mu(t)dt + \max_{x(0)\in X_0} v^+(x(0),0)\}. \tag{1.7}$$

Next, we review the polytopic overapproximation of reachable sets for linear dynamical systems and derive computational methods for polytopic overapproximate reachable sets for feedback linearizable nonlinear systems, linear dynamic games, and norm-bounded nonlinear systems.

1.3.1 Linear dynamical systems

In this section, we review the polytopic overapproximation of reachable sets for linear systems from [1]. Consider a time-varying linear dynamical system

$$\dot{x}(t) = A(t)x(t) + B(t)u(t), \quad x(0) \in X_0,\ u(t) \in U \tag{1.8}$$

where the initial set X_0 and the admissible control input set U are assumed to be convex polytopes which have N and N_u faces respectively. In this chapter, we assume the initial set X_0 is a polytope, but in general the number of faces of the initial set is a design parameter since X_0 may be a convex compact set and thus the more the number of faces of X_0 the better the overapproximate reachable set.

A convex polytope $\mathcal{P}$ with K faces can be represented in two ways; it can be represented as the bounded intersection of K half spaces,

$$\mathcal{P} = \bigcap_{i=1}^{K} \{x|h_i^T x \leq \gamma_i\} \tag{1.9}$$

where h_i is a normal vector to the i^{th} face of the polytope $\mathcal{P}$. A convex polytope can also be represented as the convex hull of its vertices: if a convex polytope $\mathcal{P}$ has m vertices $\{v^1, \ldots, v^m\}$, then

$$\mathcal{P} = \{x|x = \sum_{i=1}^{m} \alpha_i v^i,\ \alpha_i \geq 0,\ \sum_{i=1}^{m} \alpha_i = 1\} \tag{1.10}$$

Define a set of linear functions as

$$v_i^+(x,t) = h_i^T(t)x, \quad i \in \{1,2,\ldots,N\}. \tag{1.11}$$

These linear functions are used to represent a convex polytope as shown in (1.9). In order to find a polytopic overapproximate reachable set, we solve for $v_i^+(x,t)$ in (1.11) that satisfies (1.6). Then, (1.6) becomes

$$\begin{aligned} &D_t v_i^+(x,t) + \max\nolimits_{u\in U}\{< D_x v_i^+(x,t), f(x,u) >\} \\ &\quad =< \dot{h}_i(t), x(t) > + < A(t)^T h_i(t), x(t) > +\max\nolimits_{u\in U}\{< h_i(t), B(t)u(t) >\} \\ &\quad \leq \mu(t). \end{aligned} \tag{1.12}$$

From optimal control theory [26], the adjoint equation for linear systems when the input set does not depend on x is $\dot{\lambda}(t) = -A(t)^T\lambda(t)$. If we choose $h_i(t) = \lambda(t)$ ($i \in \{1,2,\ldots,N\}$), then

$$< \dot{h}_i(t), x(t) > + < A(t)^T h_i(t), x(t) >= 0. \tag{1.13}$$

This represents the evolution of the normal vector of the i^{th} face. Let $h_i(0),\ i \in \{1,2,\ldots,N\}$ be the normal vectors of the faces of the initial set X_0. Then, the solution to (1.13) is

$$h_i(t) = \Phi(t,0)h_i(0), \quad i \in \{1,2,\ldots,N\} \tag{1.14}$$

where $\Phi(t,0)$ is the state transition matrix satisfying $\dot{\Phi} = -A(t)^T\Phi,\ \Phi(0,0) = I$. If the system dynamics in (1.8) is time invariant, then $\Phi(t,0) = e^{-A^Tt}$ and (1.14) becomes

$$h_i(t) = e^{-A^Tt}h_i(0), \quad i \in \{1,2,\ldots,N\}. \tag{1.15}$$

Thus, for a linear time invariant system, the evolution of normal vectors can be determined analytically. We denote $\{u^1,\ldots,u^{m_u}\}$ as the vertices of the input set U. Since U is a convex polytope, the following must hold: (for $j \in \{1,\ldots,m_u\}$)

$$\max_{u\in U} < h_i(t), B(t)u(t) >= \max_j < h_i(t), B(t)u^j >\leq \mu(t) \tag{1.16}$$

that is, the maximum is achieved at a vertex of U [1]. Furthermore, if the system dynamics in (1.8) is time invariant, (1.16) is simplified to

$$\max_j < h_i(t), Bu^j >= \max_j < e^{-A^Tt}h_i(0), Bu^j >\leq \mu(t) \tag{1.17}$$

for $j \in \{1,\ldots,m_u\}$. We choose $\mu(t) = \max_j < h_i(t), B(t)u^j >$ and note that $\mu(t)$ is always positive for a properly chosen input set U (e.g., chosen such that $0 \in U$). Then, the linear function $v_i^+(x,t)$ in (1.11) is a supporting hyperplane of the exact reachable set [1]. A polytopic overapproximate forward reachable set $V^+(t)$ for the dynamical system (1.8) is the intersection of half spaces as follows:

$$\begin{aligned} V^+(t) = \textstyle\bigcap_{i=1}^N\{\ x : v_i^+(x,t) \leq \int_0^t \max_j < h_i(s), B(s)u^j > ds \\ + \max\nolimits_{x(0)\in X_o} v_i^+(x(0),0)\}. \end{aligned} \tag{1.18}$$

The set $V^+(t)$ is a convex polytope which contains the exact reachable set at time t since each $v_i^+(x,t)$ in (1.18) is a supporting hyperplane of the exact reachable set. If the system dynamics is linear time invariant, $V^+(t)$ becomes

$$\begin{aligned} V^+(t) = \textstyle\bigcap_{i=1}^{N} \{\, x : v_i^+(x,t) \le \int_0^t \max_j < e^{-A^T s} h_i(0), Bu^j > ds \\ + \max_{x(0)\in X_o} v_i^+(x(0),0) \}. \end{aligned} \tag{1.19}$$

1.3.2 Feedback linearizable nonlinear systems

In this section, we consider a class of nonlinear systems [27], in which $u(t)$ is a feedback control:

where

$$\dot{x}(t) = f(x) + g(x)u(t) \tag{1.20}$$

$$u(t) = a(x(t)) + b(x(t))v(t). \tag{1.21}$$

We assume that there exists a diffeomorphism T: such that $z = T(x)$, which transforms, with a control input $u(t)$, a nonlinear system (1.20) into an equivalent linear system [27]. Then, we can compute an overapproximate forward reachable set for the nonlinear system (1.20) as follows:

- Step 1: Transform the nonlinear system (1.20) to an equivalent linear system, $\dot{z}(t) = A(t)z(t) + B(t)v(t)$ with appropriate $u(t)$ and T.
- Step 2: Compute a polytopic overapproximate forward reachable set $V^+(t)$ of the linear system following the procedure in Section 1.3.1.
- Step 3: Using the inverse state transformation $x = T^{-1}(z)$, we obtain the overapproximate forward reachable set for the original nonlinear system (1.20) from $V^+(t)$.

Since there is no approximation during the transformation and the transformation is a diffeomorphism on a given domain of interest, the forward reachable set obtained in Step 3 is guaranteed to be an overapproximate forward reachable set of the nonlinear system (1.20).

1.3.3 Linear dynamic games

Now, we consider the linear dynamic game:

$$\begin{gathered} \dot{x}(t) = A(t)x(t) + B(t)u(t) + C(t)d(t), \\ x(0) \in X_0,\ u(t) \in U,\ d(t) \in D \end{gathered} \tag{1.22}$$

where the initial set X_0, the admissible control input set U, and the disturbance input set D are assumed to be convex polytopes which have N, N_u, and N_d faces respectively. Then, the HJI equation in (1.3) for a forward reachable set computation becomes [19, 20],

$$\begin{aligned} D_t v(x,t) + \max_{u\in U} \min_{d\in D} \{ < D_x v(x,t), \\ A(t)x(t) + B(t)u(t) + C(t)d(t) > \} = 0. \end{aligned} \tag{1.23}$$

To find an overapproximate solution to (1.23), we look for a set of linear functions $v_i^+(x,t)$ in (1.11) satisfying (1.13), and compute

$$\begin{aligned} D_t v_i^+(x,t) &+ \max\nolimits_{u\in U} \min\nolimits_{d\in D}\{< D_x v_i^+(x,t), \\ &\quad A(t)x(t)+B(t)u(t)+C(t)d(t)>\} \\ &= \max\nolimits_{u\in U}\{< h_i(t), B(t)u(t)>\} \\ &\quad + \min\nolimits_{d\in D}\{< h_i(t), C(t)d(t)>\} \\ &\le \mu(t). \end{aligned} \tag{1.24}$$

We denote $\{u^1,\ldots,u^{m_u}\}$ and $\{d^1,\ldots,d^{m_d}\}$ as the vertices of U and D respectively. Since (1.24) is linear with respect to u and d, the maximum and the minimum in (1.24) are achieved at vertices of U and D as follows:

$$\max_j < h_i(t), B(t)u^j > + \min_k < h_i(t), C(t)d^k > \le \mu(t) \tag{1.25}$$

for $j \in \{1,\ldots,m_u\}$, $k \in \{1,\ldots,m_d\}$.

By choice of $\mu(t) = \max_j < h_i(t), B(t)u^j > + \min_k < h_i(t), C(t)d^k >$, the polytopic overapproximate reachable set $V^+(t)$ for the linear dynamic game (1.22) is

$$V^+(t) = \bigcap_{i=1}^{N}\{x : v_i^+(x,t) \le \int_0^t \mu(s)ds + \max_{x(0)\in X_0} v_i^+(x(0),0)\}. \tag{1.26}$$

1.3.4 Norm-bounded nonlinear systems

In this section, we consider a norm-bounded nonlinear system,

$$\begin{aligned} &\dot{x}(t) = A(t)x(t) + B(t)u(t) + \phi(x,t), \\ &x(0)\in X_0,\ u(t)\in U,\ \|\phi(x,t)\| \le \beta(t) \end{aligned} \tag{1.27}$$

where the initial set X_0 and the admissible control input set U are assumed to be convex polytopes which have N and N_u faces respectively. $\|\cdot\|$ represents the Euclidean norm; $\beta(\cdot)$ is a positive-definite function. Then, the HJI equation in (1.3) becomes

$$D_t v(x,t) + \max_{u\in U}\{< D_x v(x,t), A(t)x(t) + B(t)u(t) + \phi(x,t) >\} = 0. \tag{1.28}$$

To compute an overapproximate solution to the HJB equation in (1.28), we find the linear functions $v_i^+(x,t)$ in (1.11) satisfying (1.13), and compute

$$\begin{aligned} D_t v_i^+(x,t) &+ \max\nolimits_{u\in U}\{< D_x v_i^+(x,t), A(t)x(t)+B(t)u(t)+\phi(x,t)>\} \\ &= \max\nolimits_{u\in U}\{< h_i(t), B(t)u(t)>\} + < h_i(t), \phi(x,t)> \\ &\le \max\nolimits_{u\in U}\{< h_i(t), B(t)u(t)>\} + \tfrac{1}{2}(\|h_i(t)\|^2 + \|\phi(x,t)\|^2) \\ &\le \max\nolimits_j\{< h_i(t), B(t)u^j >\} + \tfrac{1}{2}(\|h_i(t)\|^2 + \beta(t)^2) \\ &\le \mu(t). \end{aligned} \tag{1.29}$$

If we choose $\mu(t)$ such that

$$\mu(t) = \max_j < h_i(t), B(t)u^j > + \frac{1}{2}(\|h_i(t)\|^2 + \beta(t)^2), \tag{1.30}$$

then a polytopic overapproximate reachable set $V^+(t)$ for the norm-bounded dynamical system (1.27) is

$$\begin{aligned} V^+(t) = \bigcap_{i=1}^N \{x : v_i^+(x,t) \le \textstyle\int_0^t [\max_j < h_i(s), B(s)u^j > \\ + \tfrac{1}{2}(\|h_i(s)\|^2 + \beta(s)^2)]ds + \max_{x(0)\in X_0} v_i^+(x(0),0)\}. \end{aligned} \tag{1.31}$$

If $\phi(x,t)$ belongs to a polytope with vertices $\{\phi^1, \ldots, \phi^{m_\phi}\}$, a polytopic overapproximate reachable set $V^+(t)$ becomes

$$\begin{aligned} V^+(t) = \bigcap_{i=1}^N \{x : v_i^+(x,t) \le \textstyle\int_0^t [\max_j < h_i(s), B(s)u^j > \\ + \max_k \{< h_i(s), \phi^k >\}]ds + \max_{x(0)\in X_0} v_i^+(x(0),0)\}. \end{aligned} \tag{1.32}$$

1.4 Examples

We consider three examples: a linear system, a norm-bounded nonlinear system, and we conclude with the example which motivated this study, a nonlinear, feedback linearizable, dynamic game. Note that equation (1.7) provides overapproximations of the sets of reachable states over a range of times (the flow). In the implementation, we compute overapproximations of the reachable sets at specific instants of time without interpolation between the sets.

1.4.1 Linear dynamical systems

In this section, we consider a linear dynamical system $\dot{x} = Ax + Bu$, $x(0) \in X_0$ where the control input $u(t)$ can vary inside a convex polytope U and the initial set X_0 is also a convex polytope. The system parameters (A, B, X_0, and U) given in [11] are used. Fig. 1.1 shows the evolution of the projection on x_3 and x_4 over time. This result is similar to that in [11], yet computation time with the method shown in Section 1.3.1 is 1.17 seconds (which includes plotting the result shown in Fig. 1.1) using MATLAB on a 700MHz Pentium III PC. For comparison, the algorithm proposed in [11] takes 18 seconds using the same parameters.

1.4.2 Norm-bounded nonlinear systems

We consider a norm-bounded nonlinear system

$$\dot{x} = A(t)x + B(t)u(t) + \phi(x,t), \; x(0) \in X_0, \; u(t) \in U \tag{1.33}$$

where the initial set X_0 and the control input set U are convex polytopes. The nonlinear function $\phi(x,t)$ is assumed to be norm-bounded i.e., $\|\phi(x,t)\| \le \frac{1}{3}t$ where $t > 0$. The system parameters are defined as follows:

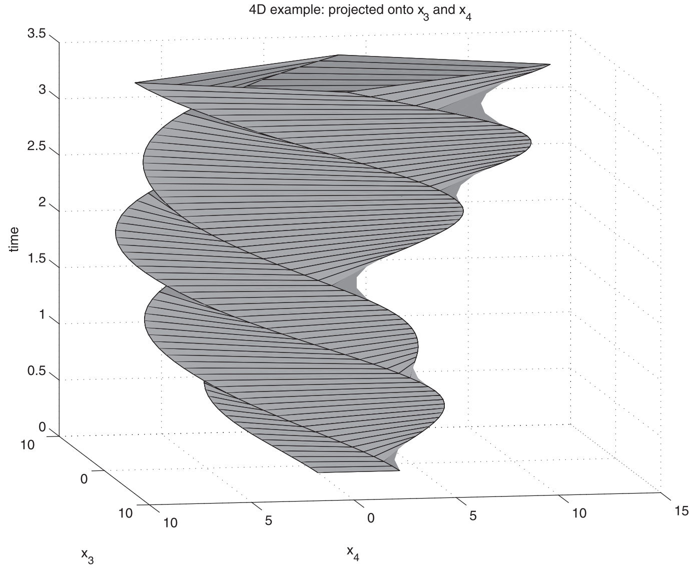


Fig. 1.1. The forward reachable set of a four-dimensional linear dynamical system (projection onto x_3 and x_4).

$$A = \begin{bmatrix} -0.5 & 4.0 \\ -3.0 & -0.5 \end{bmatrix}, \quad B = \begin{bmatrix} -1 \\ 0 \end{bmatrix},$$

$$X_0 = [4,5] \times [4,5], \; U = [-0.1, 0.1].$$

The evolution of the forward reachable set over time is shown in Fig. 1.2 and its computation time is 0.87 seconds (including plotting the result) using MATLAB on the same PC.

1.4.3 Conflict resolution between two aircraft

Last, we consider the two aircraft collision avoidance problem, as an example of feedback linearizable nonlinear systems and linear dynamic games. This is the same problem (the motivation for this research) described in Section 1.2. Fig. 1.3 shows the relative configuration between two aircraft showing the protected zone.

Aircraft 1 tries to avoid a conflict with aircraft 2 within the limits of its capability. Thus, we want to compute a backward reachable set (unsafe set) from the target set (protected zone). The target set represents the states from which the two aircraft would eventually have a conflict no matter how aircraft 1 tries to avoid it [5].

Using dynamic extension [27] with σ_i as a new state variable (compared to (1.5)), we obtain a new nonlinear model which is feedback linearizable [28],

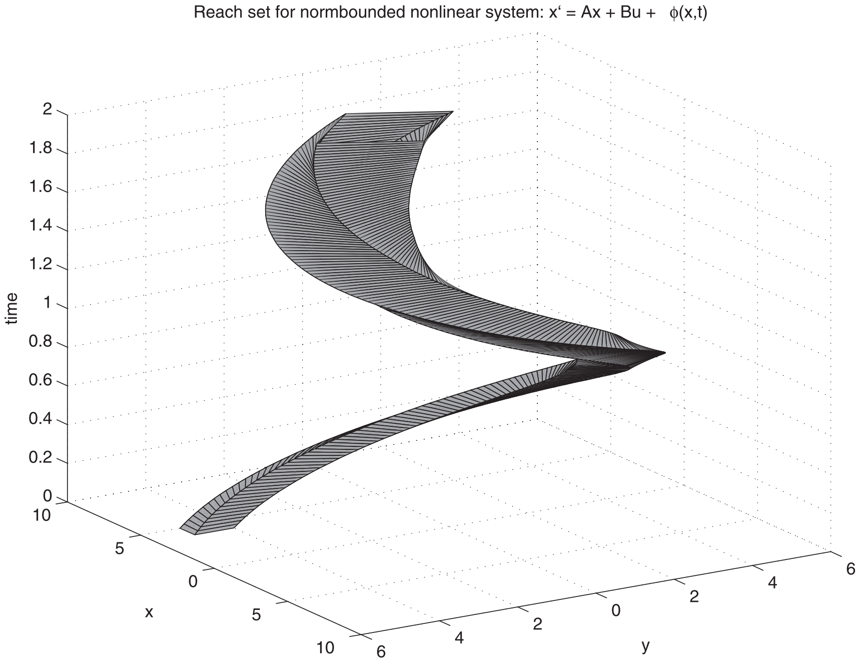


Fig. 1.2. The forward reachable set of a norm-bounded nonlinear system.

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{\psi}_i \\ \dot{\sigma}_i \end{bmatrix} = \begin{bmatrix} \sigma_i \cos\psi_i \\ \sigma_i \sin\psi_i \\ \omega_i \\ a_i \end{bmatrix}, \ (i \in \{1,2\}), \tag{1.34}$$

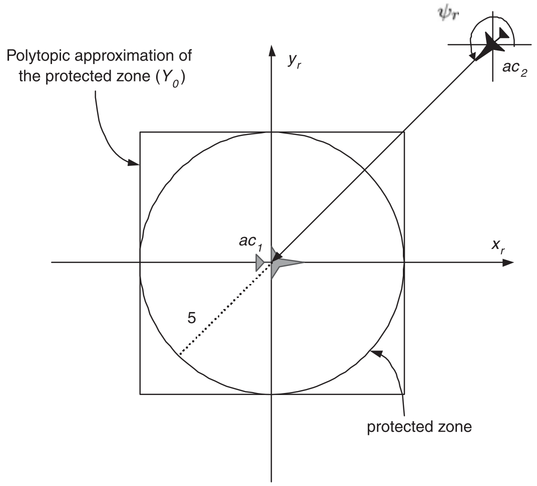


Fig. 1.3. Relative configuration of two aircraft showing the protected zone.

where a_i is the acceleration of aircraft i and is a new control input. Thus, the new state and input variables are $\xi_i := [x_i \, y_i \, \psi_i \, \sigma_i]^T$ and $\eta_i := [a_i \, \omega_i]^T$ respectively. We introduce a change in state variables, $z_i = T(\xi_i)$, and a change of the input variables, $\eta_i = M(\xi_i)u_i$, as in [28]. We denote that T and M are diffeomorphisms everywhere except at $\sigma_i = 0$. Then, the feedback linearized model of the nonlinear kinematic aircraft model in (1.34) obtained through the transformations T and M is [28]:

$$\dot{z}_i = \frac{\partial T}{\partial \xi}_{i} \, \dot{\xi}_i \Rightarrow \dot{z}_i = Az_i + Bu_i \tag{1.35}$$

with A and B as defined in [28].

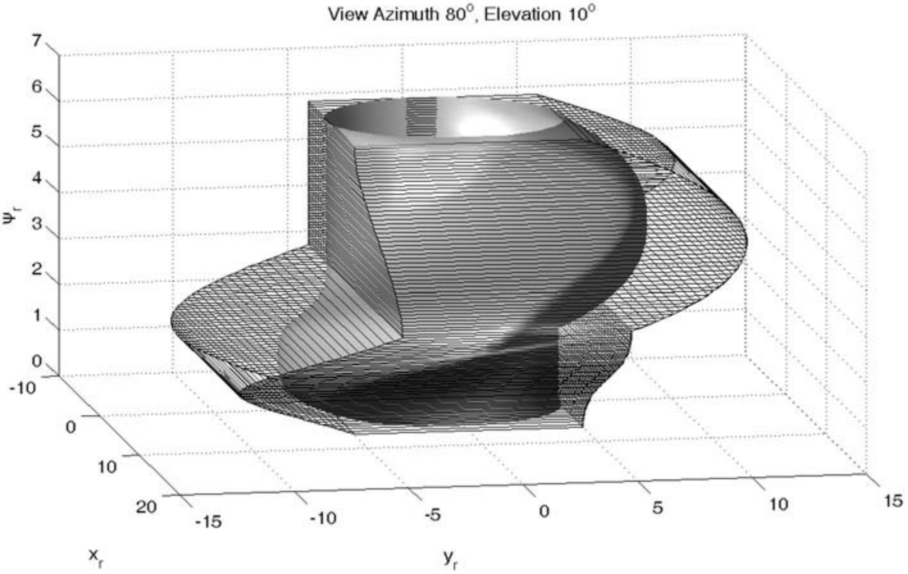


Fig. 1.4. Comparison between overapproximate (grid) and exact (solid) backward reachable sets (unsafe sets) of conflict resolution between two aircraft.

The relative kinematic aircraft model between two aircraft can be obtained by introducing new states $\xi_r := \xi_2 - \xi_1$ in the original nonlinear state space and $z_r := z_2 - z_1$ in the linearized state space. Thus, a linearized relative kinematic aircraft model is

$$\dot{z}_r = Az_r + Bu_2 - Bu_1, \quad u_2 \in U, \, u_1 \in D, \tag{1.36}$$

where the admissible control input set U and the disturbance input set D are polytopes. This is a linear dynamic game since aircraft 1 (u_1) tries to keep aircraft 2 from entering into its protected zone (target set) to prevent a conflict, but aircraft 2 (u_2) tries to enter the protected zone of aircraft 1. A target set (protected zone) is assumed to be $\mathcal{Y}_0 = [-5, 5] \times [-5, 5] \times [-\pi, \pi]$. Using dynamic extension, we have performed the computation in four dimensions (1.36) and projected the result onto the relative coordinate in three-dimensional space. A polytopic overapproximate backward reachable set is first computed in the linearized space, and then the overapproximate backward reachable set in the original state space is obtained through the transformations T and M. The overapproximate backward reachable set for conflict resolution with heading changes only, using the target set $\mathcal{Y}_0$, normalized aircraft speeds $\sigma_1 = \sigma_2 = 5$, angular velocities $|\omega_1| \leq 1$ and $|\omega_2| \leq 1$ is compared with the exact solution in [4] in Fig. 1.4.

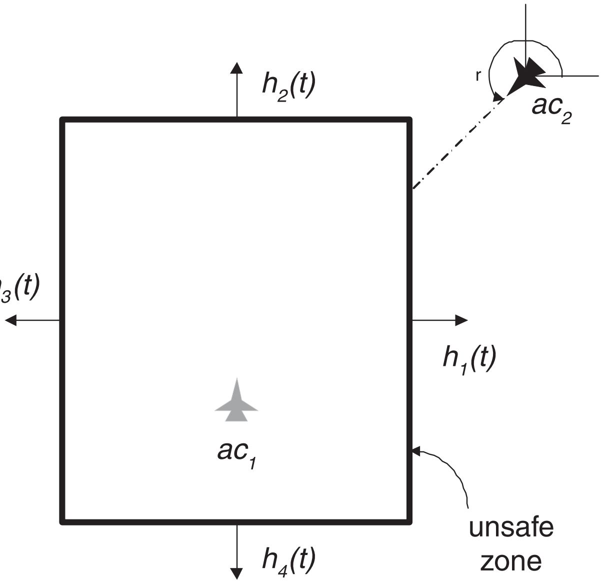


Fig. 1.5. Conflict scenario: Aircraft 2 reaches the boundary of the unsafe zone of aircraft 1 with a given initial relative angle ψ_r.

The backward reachable set obtained by using the polytopic approximation is overapproximate of the exact reachable set and its computation time is about 1.0 seconds (including plotting the result as shown in Fig. 1.4) using MATLAB on the same PC, where the numerical solution to the exact PDE [5] takes approximately 4 minutes on a Sun UltraSparc II with 50 grid nodes in each dimension. Fig. 1.5 shows a conflict scenario in which aircraft 2 tries to enter the unsafe zone. When aircraft 2 reaches the boundary of the unsafe zone, the optimal control input for aircraft 1 can be easily obtained as follows:

$$\begin{aligned} u_1^*(t) &= \arg\max_{u_1 \in D}\{< D_x v(x,t), -B(t)u_1(t)) >\} \\ &= \arg\max_j < e^{-A^T t} h_1(0), -Bu_1^j > . \end{aligned} \tag{1.37}$$

Fig. 1.6 shows a simulation for conflict resolution between the two aircraft with the initial condition $(x_r = 10,\ y_r = -20, \psi_r = 115^o)$. Since both aircraft behave optimally, the relative position of aircraft 2 moves along the boundary of the unsafe

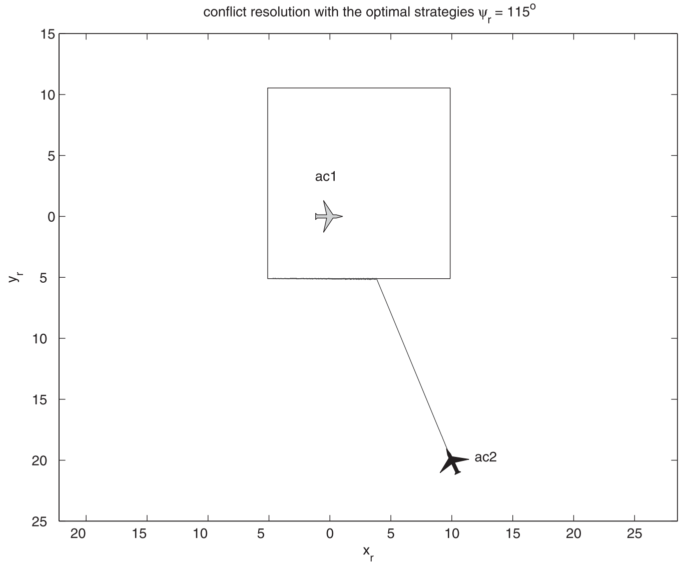


Fig. 1.6. Conflict resolution simulation with relative initial states $(x_r = 10,\ y_r = -20, \psi_r = 115^o)$. Aircraft 1 tries to avoid a conflict with aircraft 2 with the optimal strategy.

set. As expected, chattering occurs along the boundary. To avoid such a phenomenon, one would introduce a buffer zone around the boundary so that the control inputs change smoothly as aircraft 2 approaches the boundary.

Using similar analysis to the above, we may obtain the underapproximate backward reachable set. This is obtained for the collision avoidance example, using the same parameters, and compared in Fig. 1.7 with the overapproximate set.

1.5 Conclusions

The polytopic approximation gives an overapproximation of the exact reachable set and is computationally efficient: it requires solving matrix exponentials instead of a Hamilton–Jacobi partial differential equation. The data structure of the polytopic approximation method becomes more complicated than that of the ellipsoidal approximation method [17] as the number of faces of the polytope increases, yet the computation of the matrix exponential is easier than solving the (usually Riccati type) differential equation required for the ellipsoidal methods. The optimal control input can be easily computed from the Hamiltonian since the Hamiltonian is linear with respect to the control, and the control input set is a convex polytope. The polytopic approximation method can be applied to high dimensional systems which may not be solved exactly without substantially increasing the computational time. This may be done by decomposing the computation of an approximation (over or under)

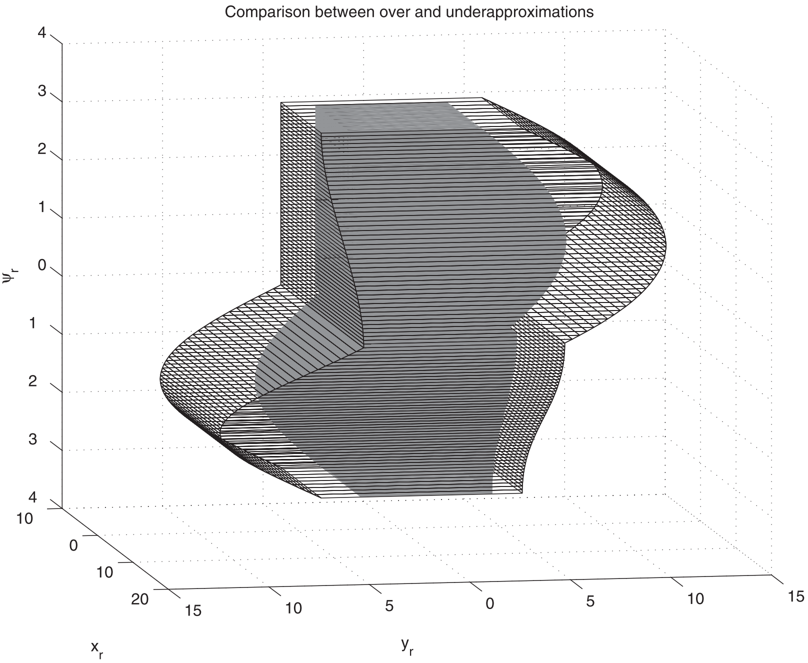


Fig. 1.7. Comparison between the under and overapproximate backward reachable sets for conflict resolution between two aircraft.

of the reachable set into a number of computations of approximations of subsystem reachable sets [29].

References

[1] P. Varaiya, Reach set computation using optimal control, In *Proceedings of the KIT Workshop on Verification of Hybrid Systems*, 377–383, Grenoble, France, 1998.

[2] J. Lygeros, C.J. Tomlin, and S. Sastry, Controllers for reachability specifications for hybrid systems, *Automatica*, **35** (3):349–370, 1999.

[3] C.J. Tomlin, J. Lygeros, and S. Sastry, A game theoretic approach to controller design for hybrid systems, *Proceedings of the IEEE*, **88** (7):949–970, July 2000.

[4] I. Mitchell, A. Bayen, and C.J. Tomlin, Validating a Hamilton-Jacobi approximation to hybrid system reachable sets, In M.D. DiBenedetto and A. Sangiovanni-Vincentelli, Eds., *Hybrid Systems: Computation and Control*, LNCS **2034**:418–431, Springer-Verlag, 2001.

[5] I. Mitchell, A.M. Bayen, and C.J. Tomlin, Computing reachable sets for continuous dynamic games using level set methods, *IEEE Transactions on Automatic Control*, 2005, to appear.

[6] A. Tiwari and G. Khanna, Series of abstraction for hybrid automata, In C.J. Tomlin and M.R. Greenstreet, Eds., *Hybrid Systems: Computation and Control*, LNCS **2289**:465–478, Springer-Verlag, 2002.

[7] R. Alur, T. Dang, and F. Ivančić, Reachability analysis of hybrid systems via predicate abstraction, In C.J. Tomlin and M.R. Greenstreet, Eds., *Hybrid Systems: Computation and Control*, LNCS **2289**:35–48, Springer-Verlag, 2002.

[8] I. Hwang, H. Balakrishnan, R. Ghosh, and C.J. Tomlin, Reachability analysis of delta-notch lateral inhibition using predicate abstraction, In S. Sahni, V.K. Prasanna and U. Shukla, Eds., *High Performance Computing-HiPC2002*, LNCS **2552**:715–724, Springer-Verlag, 2002.

[9] A. Chutinan and B.H. Krogh, Verification of polyhedral-invariant hybrid automata using polygonal flow pipe approximations, In F. Vaandrager and J.H. van Schuppen, Eds., *Hybrid Systems: Computation and Control*, LNCS **1569**:76–90, Springer-Verlag, 1999.

[10] A. Chutinan and B.H. Krogh, Computational techniques for hybrid system verification, *IEEE Transactions on Automatic Control*, **48** (1):64–75, 2003.

[11] E. Asarin, O. Bournez, T. Dang, and O. Maler, Approximate reachability analysis of piecewise-linear dynamical systems, In B. Krogh and N. Lynch, Eds., *Hybrid Systems: Computation and Control*, LNCS **1790**:20–31, Springer-Verlag, 2000.

[12] T. Dang and O. Maler. Reachability analysis via face lifting, In T.A. Henzinger and S. Sastry, Eds., *Hybrid Systems: Computation and Control*, LNCS **1386**:96–109, Springer-Verlag, 1998.

[13] I. Mitchell and C.J. Tomlin, Overapproximating reachable sets by Hamilton-Jacobi projections, *Journal of Scientific Computing*, **19** (1–3):323–346, December 2003.

[14] E. Asarin and T. Dang. Abstraction by projection and application to multi-affine systems, In R. Alur and G.J. Pappas, Eds., *Hybrid Systems: Computation and Control*, LNCS **2993**:32–47, Springer-Verlag, 2004.

[15] E.K. Kostousova, State estimation for dynamic systems via parallelotopes, *Optimization Methods and Software*, **9**:269–306, 1998.

[16] A. Kurzhanski and I. Valyi, *Ellipsoidal Calculus for Estimation and Control*, Birkhauser, 1997.

[17] A. Kurzhanski and P. Varaiya, Dynamic optimization for reachability problems, *Journal of Optimization Theory and Applications*, **108** (2):227–251, 2001.

[18] O. Botchkarev and S. Tripakis, Verification of hybrid systems with linear differential inclusions using ellipsoidal approximations, In B. Krogh and N. Lynch, Eds., *Hybrid Systems: Computation and Control*, LNCS **1790**:73–88, Springer-Verlag, 2000.

[19] A. Kurzhanski and P. Varaiya, On reachability under uncertainty, *SIAM Journal on Control and Optimization*, **41** (1):181–216, 2002.

[20] A. Kurzhanski and P. Varaiya, Reachability analysis for uncertain systems-the ellipsoidal technique, *Dynamics of Continuous, Discrete & Impulsive Systems Series B: Applications & Algorithms*, **9** (3):347–367, 2002.

[21] A.B. Kurzhanski, I. Mitchell, and P. Varaiya, Control synthesis for state constrained systems and obstacle problems, In *Proceedings of the IFAC (NOLCOS) Symposium*, Elsevier, Stuttgart, Germany, 2004.

[22] A.B. Kurzhanski and P. Varaiya, On some nonstandard dynamic programming problems of control theory, In F. Giannessi and A. Maugeri, Eds., *Variational Analysis and Applications*, Kluwer Academic Publishers, New York, 2004.

[23] M.M. Khrustalev, Exact description of reachable sets and global optimality conditions, *Avtomatika i Telemekhanika*, (5):62–70, 1988.

[24] S. Osher and J.A. Sethian, Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, *Journal of Computational Physics*, **79**:12–49, 1988.

[25] I. Mitchell, *Games of two identical vehicles*, Technical Report, Stanford University Department of Aeronautics and Astronautics Report (SUDAAR 740), Stanford University, July 2001.

[26] A. Bryson and Y.C. Ho, *Applied Optimal Control*, Taylor and Francis, 1975.

[27] S. Sastry, *Nonlinear Systems: Analysis, Stability, and Control*, Springer-Verlag, New York, 1999.

[28] D.M. Stipanović, G. İnalhan, R. Teo, and C.J. Tomlin, Decentralized overlapping control of a formation of unmanned aerial vehicles, *Automatica*, **40**:1285–1296, 2004.

[29] I. Hwang D.M. Stipanović and C.J. Tomlin, Computation of an over-approximation of the backward reachable set using subsystem level set functions, *Dynamics of Continuous, Discrete & Impulsive Systems Series A: Mathematical Analysis*, **11**:399–411, 2004.

2

On the Problem of Measurement Feedback Control: Ellipsoidal Techniques

A.B. Kurzhanski

University of California at Berkeley
and Moscow State (Lomonosov) University

Summary. This chapter deals with the problem of measurement feedback control under set-membership uncertainty for systems with original linear structure and hard bounds on the uncertain items. It indicates feedback control strategies which ensure guaranteed deviation from a given terminal set despite the uncertain disturbances and incomplete feedback. Routes for numerical treatment of the solutions are suggested on the basis of ellipsoidal techniques.

2.1 Introduction

The problem of measurement feedback control under uncertain disturbances (noise) is one of the central topics in the theory of control synthesis. It is well motivated by applied issues and has been well developed in a stochastic setting as a combination of stochastic filtering theory with the theory of stochastic control in itself. However a considerable number of problems in control design have to deal with systems subjected to information conditions which are other than stochastic.

Here the uncertain items are treated as unknown but bounded, with preassigned bounds. Such problems have to rely on the theory of guaranteed state estimation, where the estimates of the system dynamics are set-valued, with further procedures of controlling the evolution of set-valued systems. The related approaches naturally require new types of techniques and totally new formalization of the overall measurement feedback control problems (see [1]–[6]).

These approaches heavily rely on nonlinear analysis, set-valued calculus as well as on minmax theory and differential games (see [7]–[12]), but are not simple and lead to rather cumbersome technical and numerical procedures. Such perhaps are the reasons why there are relatively few papers on set-valued, minmax or game-theoretic approaches to measurement feedback control, though the problems are pending, are seriously motivated and have to be solved.

This chapter gives a solution to a problem of measurement feedback control under set-membership uncertainty with hard bounds on the uncertain items. Here, the problem is solved using procedures based on a specific version of the ellipsoidal technique as developed jointly with P. Varaiya, [13]–[15]. The suggested schemes ensure exact solution representations with recursive computation schemes.

2.2 The Basic Problem. The Cost Functional

Let us first discuss the setting of the problem and the general approach to its solution. Given is the system

$$dx/dt = A(t)x + B(t)u + C(t)f(t), \tag{2.1}$$

with continuous matrix coefficients $A(t), B(t), C(t)$ and hard bounds on the control u and unknown disturbance $f(t)$:

$$u \in \mathcal{P}(t),\ f(t) \in \mathcal{Q}(t). \tag{2.2}$$

The system evolution is considered within time interval $t \in [t_0, t_1]$. Here $\mathcal{P}(t), \mathcal{Q}(t)$ are set-valued functions with values in the variety of convex compact sets in $\mathbb{R}^p, \mathbb{R}^q$, continuous in the Hausdorff metric. The on-line information on vector x arrives from observations due to the measurement equation

$$y(t) = H(t)x + \xi(t), \tag{2.3}$$

with $y(t) \in \mathbb{R}^m$ being the available measurement and $\xi(t)$ the unknown but bounded continuous disturbance (measurement noise):

$$\xi(t) \in \mathcal{R}(t),\ \ t \in [t_0, t_1]. \tag{2.4}$$

The set-valued function $\mathcal{R}(t)$ is similar to $\mathcal{P}(t)$, and $H(t)$ is continuous. The initial condition is given by the inclusion

$$x(t_0) \in X^0, \tag{2.5}$$

where X^0 is a given convex compact in $\mathbb{R}^n$.

The pair $\{t_0, X_0\}$ is said to be *the starting position* of the system. Given starting position $\{t_0, X^0\}$, functions $A(t), B(t), C(t), H(t)$, realization $u[s], s \in [t_0, t)$, of the implemented control as well as the set-valued functions $\mathcal{P}(t), \mathcal{Q}(t), \mathcal{R}(t)$ and the measured values

$$y_t(\sigma) = y(t+\sigma), \sigma \in [-(t-t_0), 0],$$

one may solve the problem of "guaranteed estimation" [18], [17], [5], [19]. This problem consists in specifying the *"information set"* $\mathcal{X}(t, y_t(\cdot)) = \mathcal{X}(t, \cdot) = \mathcal{X}[t]$ of system (2.1)–(2.4), which consists of the ends $x(t)$ of all the trajectories of system (2.1), consistent with equation (2.3), and constraints (2.2), (2.4), (2.5), under given realizations $u(s),\ y(s),\ s \in [t_0, t]$.

The on-line position (state) of the overall system (2.1)–(2.4) may now be taken as $\{t, \mathcal{X}[t]\}$. In a loose setting, Problem I consists in specifying a feedback strategy $U(t, \mathcal{X}[t])$ which would steer the overall system from any starting position $\{\tau, \mathcal{X}[\tau]\},\ \tau \in [t_0, t_1]$ to a preassigned μ-neighborhood $\mathcal{M}_\mu$ of a given target set $\mathcal{M}$ at given time t_1, *despite the unknown disturbances f and the incomplete measurements*. At the same time, the class $\mathcal{U} = \{U(t, \mathcal{X}[t])\}$ of such strategies should ensure the existence and prolongability of solutions to the differential inclusion

$$\dot{x} \in A(t)x + B(t)U(t, \mathcal{X}[t]) + C(t)f(t), \tag{2.6}$$

within the interval $t \in [t_0, t_1]$.

We will further indicate a rigorous formulation of the problem. However already at the preliminary stage one may observe that Problem I may be separated into two, namely into Problem GE of guaranteed state estimation and Problem CS of control synthesis in a generalized state space.

The overall Problem I may be posed with the aid of the next functional:

$$\begin{aligned}\mathcal{V}(t_0, X^0) = \min_U \max_f \max_{x(\cdot)} \{ & -h_+(x(t_0), X^0) \\ & - \int_{t_0}^{\tau} h_+(y^*(t) - H(t)x(t), \mathcal{R}(t))dt \\ & + h_+(x(t_1), \mathcal{M}) \, | U \in \mathcal{U}; f(t) \in \mathcal{Q}(t), t \in [t_0, \tau]\}. \end{aligned} \tag{2.7}$$

Here, $y^*(t)$ is the available measurement,

$$h_+(Q'', Q') = \min\{\varepsilon : Q' + \varepsilon\mathcal{B}(0) \supseteq Q''\}$$

where Q', Q'' are compacts in $\mathbb{R}^n$.

The value $h_+(Q'', Q')$ is the *Hausdorff semidistance*, while

$$h(Q'', Q') = \max\{h_+(Q'', Q'), h_+(Q', Q'')\}$$

is the *Hausdorff distance*. If $Q'' = q \in \mathbb{R}^n$ is an isolated point, then

$$h_+(Q'', Q') = d(q, Q') = \min\{(q - q', q - q')^{1/2} \, | q' \in Q'\}$$

is the *Euclid distance* from point q to set Q'.

Remark 1. (i) In the previous relation the minimum should be taken over set-valued strategies $U \in \mathcal{U}$, where $U = U(t, \mathcal{X}[t]) \subseteq \mathcal{P}(t)$ is Hausdorff-continuous in t and upper semicontinuous in $\mathcal{X}[t]$ and where the latter is the information set which defines the on-line position $\{t, \mathcal{X}[t]\}$ of the overall system.
(ii) The existence of a solution to equation (2.7) under a given class of strategies $U = U(t, \mathcal{X}[t])$ was indicated in [5].
(iii) The maximum over $x(\cdot)$ is to be taken over all motions caused by the multivalued nature of the related differential inclusion (2.6).

In order to simplify the calculations we shall treat the given problem in another coordinate system. Namely, denoting $G(t, \tau)$ to be the fundamental transition matrix of the homogenous system (2.1),

$$\frac{\partial G(t, \tau)}{\partial t} = A(t)G(t, \tau), \; G(\tau, \tau) = I.$$

Introducing the transformation $x = G(t, \vartheta)\mathbf{x}$ to a new variable $\mathbf{x}$, making the changes and returning after that to the original notations, without loss of generality we may transform the original system (2.1), (2.2) to

$$\dot{x} = B(t)u + C(t)f(t), \tag{2.8}$$
$$y(t) = H(t)x + \xi(t), \tag{2.9}$$

under constraints (2.2), (2.4), (2.5).

Before passing to the solution of the formulated problem we shall present system (2.8), (2.9) in the form of the next array of systems:

$$dx^*/dt = B(t)u, \ x^*(t_0) = 0, \tag{2.10}$$

and

with
$$dw/dt = C(t)f(t), \ w(t_0) \in \mathcal{X}^0, \tag{2.11}$$
$$z(t) = H(t)z + \xi(t), \tag{2.12}$$

where

$$x^* + w = x, \ \ z(t) = y(t) - H(t)\int_{t_0}^{t} B(s)u^*(s)ds.$$

With realization $u = u^*(s), \ s \in [t_0, t)$ given, there will be a one-to-one mapping between realizations $y^*(s)$ and $z^*(s)$. Similarly to $\mathcal{X}[t]$, one may now define the information set of system (2.11), (2.12), denoting it as $\mathcal{W}(t, z_t(\cdot)) = \mathcal{W}(t, \cdot) = \mathcal{W}[t]$. Then we have $\mathcal{X}[t] = x^*(t) + \mathcal{W}[t]$.

The given representations allow us to solve Problem E only for system (2.11), (2.12), separating this solution from the problem of specifying the control itself.

We further proceed with solving Problem GE.

2.3 The Problem of Guaranteed (Minmax) Estimation

Problem GE for system (2.11), (2.12), may be formulated in two versions: E_1 and E_2.

Problem E_1: Given are equations (2.11), (2.12) and starting position $\{t_0, \mathcal{X}^0\}$ under constraints (2.2), (2.4), (2.5), as well as the available measurements, the realization $z = z^*(t), \ t \in [t_0, \tau]$. Specify the *information set* $\mathcal{W}[\tau]$, of solutions $w(\tau)$ to system (2.11), consistent with $z^*(t), \ t \in [t_0, \tau]$ and constraints (2.2), (2.4), (2.5).

The information set $\mathcal{W}[\tau]$ is the guaranteed estimate of the realized vector $w(\tau)$. The specification of this set is the subject of the theory of *guaranteed (minmax) estimation* [16], [5],[18], [19], [3]. However, for the problems of this chapter it is necessary not only to calculate set $\mathcal{W}[\tau]$, but to arrange the calculations on-line, following the evolution of $\mathcal{W}[\tau]$ in time. Such procedures may be organized through the solution of the following problem of *dynamic optimization*.

Problem E_2: Given are the starting position $\{t_0, \mathcal{X}^0\}$, and the realization $z_t^*(\cdot)$. Find value function

$$V(t, w) = \min\{d(w(t_0), \mathcal{X}^0) \ |f(t) \in \mathcal{Q}(t), t \in [t_0, \tau]\}$$

due to equation (2.11), under additional conditions

$$w(\tau) = w; \;\; z^*(s) - Hw(s) \in \mathcal{R}(s), \;\; s \in [t_0, \tau]. \tag{2.13}$$

The second of conditions (2.13) is actually an *on-line state constraint.*

Lemma 1. *The following relation is true:*

$$\mathcal{W}[\tau] = \{w : V(t, w) \leq 0\}.$$

We shall now calculate function $V(t, w)$ by using the techniques of convex analysis. In order to do that we fix the given measurement realization $z^*(t), t \in [t_0, \tau]$, and consider the relations which are true for $d(w(t_0), \mathcal{X}^0) > 0$:

$$d(w(t_0), \mathcal{X}^0) = \max\Big\{(l, w(t_0)) - \rho(l|\mathcal{X}^0) \;\Big|\; (l, l) \leq 1\Big\} \tag{2.14}$$

and

$$\int_{t_0}^{\tau} d(z^*(t) - H(t)w(t), \mathcal{R}(t))d\alpha(t) = \max\bigg\{\int_{t_0}^{\tau}\Big((\lambda(t), z^*(t) \\ - H(t)w(t)) - \rho(\lambda(t)|\mathcal{R}(t))\Big)d\alpha(t) \;\bigg|\; \lambda(\cdot) \in \mathcal{K}[t_0, \tau]\bigg\}$$

for any $\lambda(\cdot) \in \mathcal{K}[t_0, \tau], \alpha(\cdot) \in \mathrm{Var}_+[t_0, \tau]$. Here $\mathcal{K}$ is a compact set in the space $C_r[t_0, \tau]$ of r-dimensional continuous functions and $\mathrm{Var}_+[t_0, \tau]$ — the space of nondecreasing functions of unit variation are selected [21]. The symbol $\rho(l|\mathcal{W}) = \max\{(l, w) \mid w \in \mathcal{W}\}$ denotes the value of the support function of compact $\mathcal{W}$ along direction l.

We have

$$V(\tau, w) = \max\bigg\{(s(\tau), w) - \int_{t_0}^{\tau} \rho(s(t)|C(t)\mathcal{Q}(t))dt \\ + \int_{t_0}^{\tau}\Big((\lambda(t), z^*(t)) - \rho(\lambda(t)|\mathcal{R}(t))\Big)d\alpha(t) - \rho(l|\mathcal{X}^0)\bigg\} \tag{2.15}$$

where $s(t)$ is the solution to the adjoint equation of Problem E_2, the equation

$$ds = -sA(t)dt - \lambda'(t)H(t)d\alpha(t), \;\; s(\tau) = l. \tag{2.16}$$

Note that here and in the sequel function $z^*(t)$ is considered continuous, while function $\alpha(t)$ is right-continuous. The maximums in problem (2.15) are attained and are unique.

Given $V(\tau, w)$, one may now calculate the support function

$$\rho(l|\mathcal{W}[\tau]) = \max\{(l, w)|V(\tau, w) \leq 0\}.$$

(Following [5], one may also calculate the same item directly.) We have

$$\rho(l|\mathcal{W}[\tau]) = \min_{\lambda,\alpha}\Bigg\{ \rho(s(t_0)|\mathcal{X}^0) + + \int_{t_0}^{\tau} (\rho(s(t)|C(t)\mathcal{Q}(t)))\lambda(s)d\alpha(s)$$
$$+ \int_{t_0}^{\tau} \rho(\lambda|Z(s))d\alpha(s) \;\Bigg|\; \lambda(\cdot) \in \mathcal{K}[t_0,\tau], \alpha(\cdot) \in \mathrm{Var}_+[t_0,\tau] \Bigg\}, \quad (2.17)$$

where $Z(t) = z(t) - \mathcal{R}(t)$, and $s(t)$ is the solution to the adjoint equation (2.16).

The "motion" of set $\mathcal{W}[\tau]$ may be described by an evolution equation of the "funnel" type

$$\lim_{\sigma \to 0} h_+(\mathcal{W}[\tau+\sigma], \mathcal{W}[\tau] \cap Z(\tau) + \sigma C(\tau)\mathcal{Q}(\tau)\} = 0 \quad (2.18)$$

with $\mathcal{W}[t_0] = \mathcal{X}^0$. Set $\mathcal{W}[\tau]$ will be the maximal solution with respect to inclusion of equation (2.18), (see [22], [23]).

We may now pass to the problem of control in the space of "trajectories" $\mathcal{W}[\tau]$.

2.4 The Synthesizing Control

Consider the on-line position $\{\tau, x^*, \mathcal{W}\}$ of the overall system (2.10)–(2.12).

Problem CS: For the evolution system (2.10),(2.18), specify a set-valued control strategy $U^0(t, x^*, \mathcal{W}) \supseteq \mathcal{P}(t)$, which would ensure the inclusion

$$x^*(\vartheta) + \mathcal{W}[\vartheta] \subseteq \mathcal{M} + \mu\mathcal{E}(0, I),$$

for some $\mu^0 \geq 0$, whatever be the disturbances $f(t), \xi(t)$ and the unknown starting point $w(t_0) = x(t_0)$ subjected to constraints (2.2),(2.4), (2.5) for $t \in [t_0, \vartheta]$. Here $\mathcal{E}(0, I) = \{x : (x, x) \leq 1\}$.

In order to solve the problem of control synthesis, consider first the following auxiliary problem.

Let $W(\vartheta, z^*(\cdot); \tau, \mathcal{W})$ be the reach set of system (2.11), (2.12) from position $\{\tau, \mathcal{W}\}$ over inputs $f(t) \in \mathcal{Q}(t)$, under state constraint

$$H(t)w(t) \in z^*(t) - \mathcal{R}(t), \; t \in [\tau, \vartheta],$$

where function $z^*(t)$ is generated due to a specific triplet $\zeta^* = \{w^*, f^*(\cdot), \xi^*(\cdot)\}$ subjected to constraints

$$w^* \in \mathcal{W}, \; f^*(t) \in \mathcal{P}(t), \; \xi(t) \in \mathcal{R}(t), \; t \in [\tau, \vartheta]. \quad (2.19)$$

Problem CS-V: Find the value function $\mathcal{V}(\vartheta, \tau, x^*, \mathcal{W})$, according to the formula

$$\mathcal{V}(\vartheta, \tau, x^*, \mathcal{W}) = \max_{\zeta} \min_{u} d(x^*(\vartheta), W(\vartheta, z(\cdot); \tau, \mathcal{W})),$$

with fixed $\{x^*(\tau) = x^*, \; \mathcal{W}(\tau) = \mathcal{W}\}$.

Here the minimum is to be taken over all functions $u(s) \in \mathcal{P}(s), s \in [\tau, \vartheta]$, and the maximum over all triplets $\zeta = \{w, f(\cdot), \xi(\cdot)\}$, where $\zeta \in \mathcal{Z}_\tau(\mathcal{W})$ and

$$\mathcal{Z}_\tau(W) = \Big\{ \{w, f(\cdot), \xi(\cdot)\} : w \in \mathcal{W},\ f(t) \in \mathcal{Q}(t),\ \xi(t) \in \mathcal{R}(t),\ \ t \in [\tau, \vartheta] \Big\}.$$

Remark 2. In the monograph [5] it is demonstrated that the maximum over triplets $\mathcal{Z}_\tau(W)$ in Problem V-1 is equivalent to the maximum over all pairs $\{w, z(\cdot)\}$, where $w \in W$ and

$$z(t) \in Z(\tau, \mathcal{W}) = \{H(t)w(t) + \xi(t) | \zeta \in Z(\tau, \mathcal{W}),,\ t \in [\tau, \vartheta]\},$$

due to equality

$$w(t) = w + \int_\tau^t C(s)f(s)ds.$$

The solution to this problem follows the techniques of convex analysis as applied along the lines of monograph [5], sections 17,18, and also [13], [24]. We have the next proposition.

Theorem 1. *The following relation is true:*

$$\mathcal{V}(\vartheta, \tau, x^*, W) = \max\{\Psi(l, \tau, \vartheta, x^*, \mathcal{W}) | (l, l) \le 1\},$$

where

$$\Psi(l, \tau, \vartheta, x^*, \mathcal{W}) = l' x^*[\tau] + \rho(l | \mathcal{W}[\tau]) + \int_\tau^\vartheta (\rho(l | C(t)\mathcal{Q}(t))dt$$
$$- \int_\tau^\vartheta (\rho(-l | B(t)\mathcal{P}(t))dt - \rho(l | \mathcal{M}). \tag{2.20}$$

We will be further interested in the variety $\mathcal{X}[t, \mathcal{W}]$ of all convex compact sets of type $\mathcal{X}[\tau] = \{x^* + \mathcal{W}\}$ which satisfy for a fixed $\mathcal{W}$ the inequality

$$\mathcal{V}(\tau, x^*, \mathcal{W}) \le 0,\ \forall x^* \in \mathcal{X}[\tau]. \tag{2.21}$$

Following conventional reasoning (see [5], [24], [13]) we come to the next conclusion.

Corollary 1. *Among the variety of sets $\mathcal{X}[\tau] \in \mathcal{X}[\tau, \mathcal{W}]$ there exists a set $\mathcal{X}^*[\tau]$ which is maximal with respect to inclusion, namely*

$$\mathcal{X}[\tau] \in \mathcal{X}^*[\tau],\ \forall \mathcal{X}[\tau] \in \mathcal{X}[t, \mathcal{W}].$$

The set $\mathcal{X}^*[t]$ may be represented through a multivalued integral as

$$\mathcal{X}^*[t] = \left(G(\tau, \vartheta)\mathcal{M} - \int_\tau^\vartheta G(\tau, s)B(s)\mathcal{P}(s)ds \right)$$
$$= \int_\tau^\vartheta G(\tau, s)C(s)\mathcal{Q}(s)ds \tag{2.22}$$

where

$$\mathcal{X}' \dot{-} \mathcal{X}'' = \{x : x + \mathcal{X}'' \subseteq \mathcal{X}'\}$$

stands for the geometric (Minkowski) difference of convex sets $\mathcal{X}', \mathcal{X}''$.

Assumption 1. *The function*

$$\begin{aligned} k(l,\tau)) \; = \; & \left(\rho(l|G(\tau,\vartheta)\mathcal{M}) + \int_\tau^\vartheta (\rho(-l|G(\tau,s)B(s)\mathcal{P}(s))ds\right) \\ & - \int_\tau^\vartheta \rho(l|G(\tau,s)C(s)\mathcal{Q}(s))ds \end{aligned}$$

is proper, convex in l for all $\tau \in [t_0, \vartheta]$.

We take this assumption to be true. Then

$$\begin{aligned} \rho(l|\mathcal{X}^*[\tau]) = {} & \rho(l|G(\tau,\vartheta)\mathcal{M}) \\ & + \int_\tau^\vartheta \Big(\rho(-l|G(\tau,s)B(s)\mathcal{P}(s)) - \rho(l|G(\tau,s)C(s)\mathcal{Q}(s))\Big)ds. \quad (2.23) \end{aligned}$$

Relation (2.23) may be presented as a map $\mathcal{X}^*[\tau] = T_\tau^\vartheta \mathcal{M}$.

Remark 3. Under Assumption 1 we shall use set $\mathcal{X}^*[t]$ for constructing the synthesizing control strategy. In the absence of this assumption one has to consider instead of $\mathcal{X}^*[\tau]$ a multivalued integral of the alternated Pontryagin type, [25], [26], [15], which is the Hausdorff limit

$$\lim_{N \to \infty} \mathcal{X}_N[\tau] = \mathcal{X}^0[\tau]$$

of a nonempty superposition

$$\mathcal{X}_N[\tau] = T_\tau^{\tau_1} T_{\tau_1}^{\tau_2} ... T_{\tau_N}^{\vartheta} \mathcal{M}$$

taken over a sequence of partitions $\{\tau, \tau_1, ..., \tau_N, \vartheta\}$ of the interval $[t_0, \vartheta]$ with uniformly increasing density when $N \to \infty$.

The emphasis of this chapter is on the ellipsoidal technique rather than the general scheme. This may justify the acceptance of Assumption 1.

We shall now indicate the solution strategy for solving the Problem CS. Suppose that at time τ the on-line realizations of systems (2.10), (2.11), (2.12) are $\{x^* = x^*[\tau], \mathcal{W} = \mathcal{W}[\tau]\}$. Consider function

$$\begin{aligned} \mathcal{V}(\tau, x^*, \mathcal{W}) &= h_+(x^* + \mathcal{W}[\tau], \mathcal{X}^*[\tau]) \\ &= \max\{(l, x^*) + \rho(l\; \mathcal{W}[\tau]) - \rho(l|X^*[\tau]) \mid (l,l) \le 1\}. \quad (2.24) \end{aligned}$$

This problem has a unique maximizer l^0 for $\mathcal{V}(\tau, x^*, \mathcal{W}) > 0$. If $\mathcal{V}(\tau, x^*, \mathcal{W}) = 0$ we take $l^0 = 0$. Proceeding further, introduce the set-valued strategy

$$U^0(\tau,x^*,\mathcal{W}) =$$
$$\arg\min_u \left\{ \max\left\{ \left.\frac{d\mathcal{V}(\tau,x^*,\mathcal{W})}{dt}\right|_{u,\zeta} \middle| w\in\mathcal{W}, \zeta\in\mathcal{Z}_\tau(\mathcal{W})\right\} \middle| u\in\mathcal{P}(\tau)\right\}, \quad (2.25)$$

if $\mathcal{V}(\tau,x^*,\mathcal{W}) > 0$, and $U^0(\tau,x^*,\mathcal{W}) = \mathcal{P}(\tau)$, if $\mathcal{V}(\tau,x^*,\mathcal{W}) = 0$.

Here $d\mathcal{V}(\tau,x^*,\mathcal{W})/dt|_{u,\zeta}$ is the total derivative of functional $\mathcal{V}(\tau,x^*,\mathcal{W})$ due to the "motions" $\{x^*(\tau),\mathcal{W}[\tau]\}$ described by (2.10), (2.18) and formula (2.17), being taken along "directions" $\{u,\zeta\}$. We shall prove that strategy $U^0(\tau,x^*,\mathcal{W})$ ensures functional $\mathcal{V}(\tau,x^*,\mathcal{W})$ to be nondecreasing along the "motions" $\{x^*(\tau),\mathcal{W}[\tau]\}$.

Let us calculate the total derivative of function $\mathcal{V}(\tau,x^*,\mathcal{W}) > 0$. Denote the unique maximizer in (2.24) as l^0. Using the rules of differentiating functions of "maximum" type, we have

$$d\mathcal{V}(\tau,x^*,\mathcal{W}[\tau])/d\tau|_{u,\zeta} = \{(l^0,\dot{x^*}) - d\Phi(l^0,\tau)\}/d\tau|_{u,\zeta}, \quad (2.26)$$

where

$$\Phi(l,\tau) = \rho(l|\mathcal{M}) + \int_\tau^\vartheta \Big(\rho(-l|B(s)\mathcal{P}(s)) - \rho(l|C(s)\mathcal{Q}(s))\Big)ds$$
$$- \rho(l|\mathcal{W}[\tau]).$$

This gives, in view of relations (2.23),(2.24)

$$d\mathcal{V}(\tau,x^*,\mathcal{W}[\tau])/d\tau|_{u,\zeta} = (l^0,B(\tau)u) + \rho(-l^0|B(\tau)P(\tau)))$$
$$- \rho(l^0|C(\tau)Q(\tau))) + d(\rho(l^0|\mathcal{W}[\tau])/d\tau\;|_\zeta,$$

where

$$d(\rho(l^0|\mathcal{W}[\tau])/d\tau|_\zeta = (l^0,C(\tau)f(\tau))) + \rho(\lambda^0 H(\tau)|w-\mathcal{W})$$
$$+ \rho(\lambda^0|\xi(\tau)-\mathcal{R}(\tau)).$$

Here it was assumed that function $\lambda^0(\tau)$ is the maximizer for a problem such as (2.21), but taken for the interval $[\tau,\tau+\sigma],\ \sigma>0$. It is also assumed that here the comaximizer $\alpha^0 \equiv t$, which means that $\alpha^0(t)$ has no jump at time τ. The maximizer $\lambda^0(\tau)$ is assumed continuous and unique, allowing representation $d\Lambda^0(\tau)/d\tau = \lambda^0(\tau)d\alpha^0(\tau)$ with $\Lambda(\tau)$ being a function of bounded variation, of dimension m. The uniqueness of maximizer $\lambda^0(\tau)$ may be ensured by assuming $\mathcal{R}(\tau)$ to be a nondegenerate ellipsoid (such an $\mathcal{R}$ is taken in the next section). Here we also used the property that either $\lambda^0(\tau)H(\tau) = l^{0'}(\tau)$, or $\lambda^0(\tau)=0$, (see [23], Section 17).

Note that the last derivative is calculated along "directions" $\{u,\zeta^*\}$ where $\zeta^* = \{w^*, f^*(\tau),\xi^*(\tau)\}$. After a maximization over all $\zeta^*\in\mathcal{Z}_\tau(\mathcal{W})$, we come to

$$\max_\zeta\{d\mathcal{V}(\tau,x^*,\mathcal{W}[\tau])/d\tau|_{u,\zeta^*}|\zeta\in\mathcal{Z}_\tau\} \quad (2.27)$$
$$= -(-l^0,B(\tau)u) + \rho(-l^0|B(\tau)\mathcal{P}(\tau)) \quad (2.28)$$

We may now specify the desired solution of Problem CS as strategy

$$U^0(\tau, x^*, \mathcal{W}) = \arg\max\{(-l^0, B(\tau)u) | u \in \mathcal{P}(\tau)\}. \tag{2.29}$$

Summarizing the above, we come to the assertion

Theorem 2. *Strategy $U^0(\tau, x^*, \mathcal{W})$ of (2.25), (2.29) ensures the inequality*

$$\{d\mathcal{V}(\tau, x^*, \mathcal{W}[\tau])/d\tau|_{u,\zeta}\} \leq 0,$$

for any $u \in U^0(\tau, x^*, \mathcal{W})$, whatever be $\zeta \in \mathcal{Z}_\tau$. Strategy $U^0(\tau, x^*, \mathcal{W})$ depends on the maximizer $-l^0 = l^0(\tau, x^*, \mathcal{W})$ of problem (2.24), according to (2.29).

Thus, the crucial point in solving the overall problem of measurement feedback control synthesis is to find the vector $l^0 = l^0(\tau, x^*, \mathcal{W})$. In its turn, this requires us to calculate set $X^*(\tau, \mathcal{W}(\tau))$, leading one to calculate an array of other set-valued functions. The required calculations may appear to be rather cumbersome. We shall show that these calculations may be quite feasible if based on ellipsoidal-valued calculus as introduced in [13], [24].

2.5 A Solution Through Ellipsoidal Techniques

We shall now indicate some ellipsoidal techniques for solving the problems of this chapter. Denote a nondegenerate ellipsoid with center p and shape matrix P^{-1} as

$$\mathcal{E}(p, P) = \{x : (x - p, P^{-1}(x - p)) \leq 1\}.$$

Note that its support function is

$$\rho(l|\mathcal{E}(p, P)) = (l, x) + (l, Pl)^{1/2}.$$

We further assume the target set $\mathcal{M} = \mathcal{E}(m, M)$ to be an ellipsoid and the hard bounds on $x(t_0), u, f, \xi$ also to be ellipsoidal, of respective dimensions, namely

$$\begin{gathered} x(t_0) \in \mathcal{X}^0 = \mathcal{E}(x^0, X^0),\ u \in \mathcal{P}(t) = \mathcal{E}(p(t), P(t)), \\ f(t) \in \mathcal{Q}(t) = \mathcal{E}(q(t), Q(t)),\ \xi(t) \in \mathcal{R}(t) = \mathcal{E}(0, R(t)). \end{gathered} \tag{2.30}$$

Here $M = M' > 0,\ X^0 = X^{0'} > 0$, and $\mathcal{P}(t) = \mathcal{P}(t)' > 0,\ \mathcal{Q}(t) = \mathcal{Q}(t)' > 0,\ \mathcal{R}(t) = \mathcal{R}'(t) > 0$.

We will solve the problem of measurement feedback control in several stages.

Stage 1. Solve Problem E of finding the information set $\mathcal{W}[\tau]$ for system (2.11) under constraints (2.30). Here we are actually to describe the reach set $\mathcal{W}[\tau] \in \mathcal{W}(\tau, t_0, \mathcal{E}(x^0, X^0))$ of system

$$\dot{w} \in C(t)\mathcal{E}(q(t), Q(t)),\ w(t_0) \in \mathcal{E}(x^0, X^0),$$

under on-line state constraint

$$z^*(t) - H(t)w(t) \in \mathcal{E}(0, R(t)),\ t \in [t_0, \tau],$$

where $z^*(t)$ is given. The last inclusion may be rewritten as

$$H(t)w(t) \in \mathcal{E}(z^*(t), R(t)). \tag{2.31}$$

As indicated in [13], there exists a parametrized family of external ellipsoids $\mathcal{E}(w_+(t), W_+(t))$ which approximate $\mathcal{W}[\tau]$ from above:

$$\mathcal{W}[\tau] \subseteq \mathcal{E}(w_+(t), W_+(t)),$$

and are described through ordinary differential equations:

$$\dot{w_+} = C(t)q(t) + L(t)z^*(t),\ w(t_0) = x^0, \tag{2.32}$$

$$\begin{aligned}\dot{W}_+ = &-(L'(t)H(t)W_+ + W_+H'(t)L(t)) + (\pi_u(t) + \pi_z(t))W_+ \\ &+(\pi_u(t))^{-1}C(t)Q(t)C'(t) + (\pi_z(t))^{-1}L(t)R(t)L'(t),\ W_+(t_0) = \mathcal{X}^0\end{aligned} \tag{2.33}$$

Here $L(\cdot) \in \mathcal{K}$ may be chosen in a compact set $\mathcal{K}$ of piecewise - continuous matrix functions, while $\pi_u(t) > 0, \pi_z(t) > 0$ among continuous positive functions.

Let us denote the external ellipsoids as $\mathcal{E}(w_+(\tau), W_+(\tau)|\omega(\tau)\}$ indicating their dependence on the parametrizing functions

$$\omega(\tau) = \{L(\cdot), \pi_u(\cdot), \pi_z(\cdot)\}, \quad L(\cdot) \in \mathcal{K}; \pi_u(t) > 0, \pi_z(t) > 0, t \in [t_0, \tau].$$

The next assertion is true (see [14]).

Theorem 3. *Set $\mathcal{W}[\tau]$ may be described either through its support function*

$$\rho(l|\mathcal{W}[\tau]) = -\inf\{\rho(l|\mathcal{E}(w_+(\tau), W_+(\tau)|\omega(\tau)))|\omega(\tau)\},$$

or, in set-valued terms, as

$$\mathcal{W}[\tau] = \cap\{\mathcal{E}(w_+(\tau), W_+(\tau)|\omega(\tau)))|\omega(\tau)\}.$$

Remark 4. Among the parametrizing functions $\omega(\cdot)$ there may exist for each vector l such triplets $\omega^0(\tau) = \{L^0(\cdot), \pi_u^0(\cdot), \pi_z^0(\cdot)\}$, which ensure the external ellipsoids to be *tight*, namely,

$$\rho(l|\mathcal{W}[\tau]) = \rho(l|\mathcal{E}(w_+(\tau), W_+(\tau)|\omega^0(\tau))).$$

The description of such parametrizers and a recursive form of their calculation is given in [14].

Stage 2. Given set $\mathcal{W}$, find set $\mathcal{X}^*[\tau] \in \mathcal{X}[\tau, \mathcal{W}]$. This set is described by (2.23). The calculation of ellipsoidal bounds for such sets was given in [15], [24]. We have

$$\mathcal{E}_-(x_e(\tau), X_(\tau)) \subseteq \mathcal{X}*[\tau] \subseteq \mathcal{E}_+(x_e(\tau), X_+(\tau))$$

where

$$\dot{x_e} = B(t)p(t) + C(t)q(t), \tag{2.34}$$

$$\begin{aligned}\dot{X}_+ &= -\gamma_u(t)X_+ - \pi_u^{-1}(t)B(t)P(t)B'(t)\\ &\quad + X_+^* S(t)C(t)Q^{1/2}(t) + S'(t)C'(t)Q^{1/2}(t)X_+^{*'}\end{aligned} \tag{2.35}$$

and

$$\begin{aligned}\dot{X_-} &= \gamma_f(t)X_- + \gamma_f^{-1}(t)C(t)Q(t)C'(t)\\ &\quad - (X_-^* S_1(t)B(t)P^{1/2}(t) + P^{1/2}(t)B'(t)S_1'(t)X_-^*).\end{aligned} \tag{2.36}$$

Here $X_+^* X_+^{*'} = X_+, \;\; X_-^* X_-^{*'} = X_-$ and $\gamma_u(t), \;\; \gamma_f(t)$, are positive, continuous parametrizing functions, while $S(t), S_1(t), S_2(t)$ are piecewise continuous parametrizing orthogonal matrices: $SS' = I, \; S_1S_1' = I, \; S_2S_2' = I$. The boundary conditions are

$$x_e(\vartheta) = m, \; X_+(\vartheta) = X_-(\vartheta) = M. \tag{2.37}$$

According to [15], the given ellipsoidal approximations allow exact representations of the approximated set $\mathcal{X}^*[\tau]$. Denote the duplet and triplet defined on $[\tau, \vartheta]$ as

$$\chi_+(\tau) = \{\gamma_u(\cdot), S(\cdot)\}, \;\; \chi_-(\tau) = \{\gamma_f(\cdot), S_1(\cdot), S_2(\cdot).\}$$

and also denote

$$\mathcal{E}(x_e, X_+(\tau)) = \mathcal{E}(x_e, X_+(\tau)|\chi_+(\tau)), \; \mathcal{E}(x_e, X_-(\tau)) = \mathcal{E}(x_e, X_-(\tau)|\chi_-(\tau)),$$

then we have

Theorem 4. *The following representations are true:*

$$\rho(l|\mathcal{X}^*[\tau]) = \min \rho(l|\mathcal{E}(x_e, X_+(\tau)|\chi_+(\tau))|\chi_+(\tau)\}, \tag{2.38}$$

and

$$\rho(l|\mathcal{X}^*[\tau]) = \max \rho(l|\mathcal{E}(x_e, X_-(\tau)|\chi_-(\tau))|\chi_-(\tau)\}. \tag{2.39}$$

In the previous relations the minimum and maximum are attained on some specific triplets. This is due to existence of tight approximations, [15].

Stage 3. Specify the control strategy $U^0(\tau, x^*, \mathcal{W})$. Here we first have to consider the problem (2.24) of finding the maximizer $l^0(\tau, x^*, \mathcal{W})$ for

$$\mathcal{V}(\tau, x^*, \mathcal{W}) = \max_l \{(l, x^*) + \rho(l|\mathcal{W}[\tau]) - \rho(l|\mathcal{X}^*[\tau])|(l,l) \leq 1\}$$

but treating it in terms of an *ellipsoidal approximation*.

In order to achieve a *guaranteed estimate* $\mathcal{V}_{\mathcal{E}}(\tau, x^*, \mathcal{W})$ of $\mathcal{V}(\tau, x^*, \mathcal{W})$, we shall substitute an *internal ellipsoid* $\mathcal{E}(w_c, W_+(\tau)|\omega(\tau))$ for $\mathcal{X}^*[\tau]$ and an *external ellipsoid* $X_-(\tau)|\chi_-(\tau))$ for $\mathcal{W}[\tau]$. Then we will have

$$\mathcal{V}_{\mathcal{E}}(\tau, x^*, \mathcal{W}) \leq \max_l \{(l, x^*) + \rho(l|\mathcal{E}(w_c, W_+(\tau)|\omega(\tau)) \\ - \rho(l|\mathcal{E}(x_c(\tau), X_-(\tau)|\chi_-(\tau))|(l,l) \leq 1\} = \mathcal{V}_{\mathcal{E}}(\tau, x^*, \mathcal{W}). \quad (2.40)$$

We shall now calculate the total derivative $d\mathcal{V}_{\mathcal{E}}(\tau, x^*, \mathcal{W})/d\tau$. Applying relations (2.32)–(2.36), we shall use the specific parametrizing functions π_f, π_z, γ_f and matrices S_1 that ensure the respective ellipsoids $\mathcal{E}(w_c, W_+),\ \mathcal{E}(x_c, X_-)$ to be *tight*. Such parametrizers will be marked by upper index 0, as $\pi_f = \pi_f^0, \omega = \omega^0$, for example. The necessary formulas for these "tightening" parametrizers are borrowed from [15].

Denote the unique maximizer for (2.40) with $\mathcal{V}_{\mathcal{E}}(\tau, x^*, \mathcal{W}) > 0$ as l_e and for $\mathcal{V}_{\mathcal{E}}(\tau, x^*, \mathcal{W}) = 0$ take maximizer $l_e = 0$. Applying the rules for differentiating functions of "maximum" type, we have

$$d(l_e, x^*)/d\tau = (l_e, B(\tau)u(\tau)), \quad (2.41)$$

$$d\rho(l_e|\mathcal{E}(x_c(\tau),X_-(\tau)|\chi_-^0(\tau))/d\tau = \frac{1}{2}(l_e, B(\tau)p(\tau) + C(\tau)q(\tau))(l_e, X_-(\tau)l_e)^{-1/2} \\ \times \Big(\gamma_f^0(\tau)(l_e, X_-(\tau)l_e) + (\gamma_f^0)^{-1}(\tau)(l_e, C(\tau)l_e) \\ - 2(l_e, X_*(\tau)S_1(\tau)B(\tau)P^{1/2}(\tau)l_e)\Big) \\ = (l_e, B(\tau)p(\tau) + C(\tau)q(\tau)) + (l_e, C(\tau)Q(\tau)C'(\tau))l_e)^{1/2} \\ - (l_e, B(\tau)P(\tau)B'(\tau)l_e). \quad (2.42)$$

Here we have used the relations

$$\gamma_f^0(\tau) = (l_e, C(\tau)Q(\tau)C'(\tau)l_e)^{1/2}(l_e, X_-l_e)^{-1/2},$$

$$(l_e, X_-^*(\tau)S_1^0(\tau)B(\tau)P^{1/2}(\tau)l_e) = (l_e, B(\tau)P(\tau)B'(\tau)l_e)^{1/2}(l_e, X_-(\tau)l_e)^{1/2}$$

of [15], Section 5.

Proceeding further and following Remark 4, denote the optimal parametrizing triplet for l_e as $\omega_e^0\tau = \{L_e^0, \pi_{fe}^0, \pi_{ze}^0\}$. Then we have

$$d\rho\,(\,l_e|\mathcal{E}(x_w(\tau), W_+(\tau)|\omega_e^0(\tau))/d\tau = \frac{1}{2}(l_e, W_+l_e)^{-1/2}\Big(-(l_e, (L_e^{0'}(\tau)H(\tau)W_+ \\ + W_+H'(\tau)L_e^0(\tau))l_e) + (\pi_{fe}^0(\tau) + \pi_{ze}^0(\tau))W_+ \\ + (\pi_{fe}^0)^{-1}(l_e, C(\tau)Q(\tau)C'(\tau)l_e) - (\pi_{ze}^0)^{-1}(l_e, L_e^{0'}(\tau)R(\tau)L_e^0(\tau)l_e)\Big) \\ - (l_e, L_e^{0'}H(\tau)w_c) + (l_e, C(\tau)q(\tau)) + (l_e, L_e^0z(t)) \\ = -(l_e, L_e^{0'}(\tau)H(\tau)W_+l_e)(l_e, W_+l_e)^{-1/2} + (l_e, C'(\tau)Q(\tau)C(\tau)l_e)^{1/2} \\ - (l_e, L_e^{0'}(\tau)R(\tau)L_e^0(\tau)l_e)^{1/2} + (l_e, L_e^{0'}(\tau)H(\tau)(w(\tau) - w_c(\tau))) \\ + (l_e, L_e^{0'}(\tau)\xi(\tau)) + (l_e, C(\tau)q(\tau)). \quad (2.43)$$

In the lines above we have used relations

$$\pi_f^{0e}(\tau) = (l_e, C(\tau)l_e)^{1/2}(l_e, W_+l_e)^{-1/2},$$
$$\pi_{ze}(\tau) = (l_e, L_e^{0'}(\tau)R(\tau)L_e^0(\tau)l_e)^{1/2}(l_e, W_+l_e)^{-1/2}.$$

Using the Cauchy–Bunyakovski inequality

$$\begin{aligned}(l_e, L_e'(\tau)H(\tau)W_+l_e) &= (H'(\tau)L_e(\tau)l_e, W_+l_e)\\ &\leq (H'(\tau)L_e(\tau)l_e, W_+H'(\tau)L_e(\tau)l_e)^{1/2}(l_e, W_+l_e)^{1/2}\end{aligned}$$

and summarizing (2.41) - (2.43), we come to the next assertion.

Lemma 2. *The following formula is true:*

$$\begin{aligned}d\mathcal{V}_{\mathcal{E}}(\tau, x^*, \mathcal{W})/d\tau &= (l_e, B(\tau)(u(\tau) - p(\tau))) + (l_e, B(\tau)P(\tau)B'(\tau)l_e)\\ &+ (l_e, L_e^{0'}(\tau)\xi(\tau)) - (l_e, L_e^{0'}(\tau)R(\tau)L_e^0(\tau)l_e)^{1/2}\\ &+ (H'(\tau)L_e^0(\tau)l_e, (w(\tau) - w_c(\tau)))\\ &- (H'(\tau)L_e^0(\tau)l_e, W_+H'(\tau)L_e^0(\tau)l_e)^{1/2}. \qquad (2.44)\end{aligned}$$

Let us now specify the control $U_{\mathcal{E}}(\tau, x^* + \mathcal{W})$ as

$$U_{\mathcal{E}}^0(\tau, x^* + \mathcal{W}) = \arg\max_u\{(-l_e, u)|u \in \mathcal{E}(p, P)\}. \qquad (2.45)$$

Relation (2.44) yields the conclusion:

Theorem 5. *Suppose $u^0 \in U_{\mathcal{E}}^0(\tau, x^*)$. Then the following is true:*
(i) the inequality

$$d\mathcal{V}_{\mathcal{E}}(\tau, x^*, \mathcal{W})/d\tau|_{u^0,z} \leq 0, \qquad (2.46)$$

whatever be the elements

$$z = Hw + \xi, \quad \xi \in \mathcal{E}(0, R), \quad w \in \mathcal{E}(w_c, W_+);$$

(ii) with $\mathcal{V}_{\mathcal{E}}(t_0, 0, \mathcal{E}(x^0, X^0)) = \mu^0 > 0$ the strategy $U_{\mathcal{E}}^0(\tau, x^*)$ yields the inequality $\mathcal{V}_{\mathcal{E}}(\tau, x^*, \mathcal{W}) \leq \mu^0$, ensuring at time ϑ the inclusion

$$x^*(\vartheta) + \mathcal{W}[\vartheta] \subseteq \mathcal{M} + \mu^0\mathcal{E}(0, I),$$

whatever be the system and measurement input disturbances $f(t), \xi(t)$ and the unknown initial vector $x(t_0) \in \mathcal{E}(x^0, X^))$.

Strategy $U_{\mathcal{E}}^0(\tau, x^*, \mathcal{W})$ depends on vector $l_e(\tau, x^*, \mathcal{W})$ which at each instant of time τ ensures the inequality

$$(l_e, x^*(\tau)) + \rho(l_e|\mathcal{E}(w_c(\tau), W_+(\tau)|\omega(\tau)) = \rho(l_e|\mathcal{E}(x_c(\tau), X_-(\tau))$$

for some $\mu(\tau) \leq \mu^0$, being the unique maximizer in (2.40). Moreover, the approximating ellipsoids are selected through their tightening parametrizers, so as to ensure the equality $l^0 = l_e$ for the maximizers of (2.24), (2.40), with

$$\begin{aligned}\rho(l^0|x^*(\tau) + \mathcal{W}[\tau])) &= \rho(l^0|x^*(\tau) + \mathcal{E}(w_c(\tau), W_+(\tau))) \\ &= \rho(l^0|\mathcal{E}(x_c(\tau), X_-(\tau)) + \mu(\tau)\mathcal{E}(0, I) \\ &= \rho(l^0|\mathcal{X}^*(\tau)) + \mu(\tau)\mathcal{E}(0, I) \qquad (2.47)\end{aligned}$$

for some $\mu(\tau) > 0$. Then the Hausdorff semidistance

$$h_+(x^* + \mathcal{W}[\vartheta], \mathcal{X}^*(\vartheta)) \leq \mu^0.$$

Remark 5. (i) The specification of the desired control strategy is thus reduced to the problem of maximizing a function of type

$$(l, x) - (l, Xl)^{1/2} + (l, Wl)^{1/2}, \;\; X = X' > 0, \; W = W' > 0,$$

over a unit ball: $l \in \mathcal{E}(0, I)$, where the latter may be substituted by an ellipsoid $\mathcal{E}(0, D), \; D = D' > 0$. Selecting an appropriate matrix D may facilitate the maximization. Such, for example, is the case, when

$$\mathcal{E}(0, D) \subseteq \mathcal{E}(0, X_-^*) \dot{-} \mathcal{E}(0, W_+),$$

and the approximation is tight.

(ii) Note that all the solutions to the differential equations for the ellipsoids used throughout this section are given in recurrent form. This is because all the parameters and parametrizing functions in these equations do not require recalculation anew online.

(iii) In the absence of Assumption 1 the essential schemes and the ellipsoidal formulas are the same, but the theoretical proofs of Section 3 would be longer, as they have to involve manipulations with alternated integrals.

2.6 Conclusion

In this chapter we indicated a solution scheme for the problem of measurement feedback control under set-membership uncertainty with hard bounds on the uncertain items. The solution is given in terms of guaranteed estimation theory and set-valued control procedures. The recommended numerical solution is based on applying tight ellipsoidal representations of set-valued functions, making use of recurrent dynamic relations in the form introduced by P. Varaiya and the author.

References

[1] A.N. Krasovski and N.N. Krasovski, *Control Under Lack of Information*, Birkhäuser Boston, 1995.

[2] T. Basar and P. Bernhard, *H^∞ Optimal Control and Related Minmax Design Problems*, Birkhäuser Boston, 2nd edition, 1995.

[3] M.R. James and J.S. Baras, Partially observed differential games, infinite-dimensional Hamilton-Jacobi-Isaacs equations, and nonlinear H^∞ control, *SIAM Jounal on Control and Optimization*, **34** (4):1342–1364, 1996.

[4] J.W. Helton and M.R. James, Extending H^∞ control to nonlinear systems, *SIAM, Advances in Design and Control*, 1996.

[5] A.B. Kurzhanski, *Control and Observation Under Uncertainty*, Nauka, Moscow, (in Russian), 1997.

[6] A.B. Kurzhanski, On the problem of measurement feedback control, *PMM – Applied Mathematics and Mechanics* (transl. from Russian), **68** (4):547–563, 2004.

[7] R.T. Rockafellar and R.J.B. Wets, *Variational Analysis*, Springer-Verlag, New York-Berlin, 1998.

[8] J-P. Aubin, *Viability Theory*, Birkhäuser, Boston, 1991.

[9] V.F. Demianov, *Minimax: Directional Differentiability*, (in Russian), Leningrad Univ. Press, 1974.

[10] N.N. Krasovski, *Rendezvous Game Problems*, Nat. Tech. Inf. Serv., Springfield, VA, 1971.

[11] P. Varaiya and J. Lin, Existence of saddle points in differential games, *SIAM Journ. on Control and Optimization*, **5** (1):142–157, 1969.

[12] N.N. Krasovski and A.I. Subbotin, *Game-Theoretical Control Problems*, Springer-Verlag, New York - Berlin, 1998.

[13] A.B. Kurzhanski and P. Varaiya, Ellipsoidal techniques for reachability analysis: Part I: External approximations. Part II: Internal approximations, box-valued constraints, In: *Optimization. Methods and Software*, (4):177–237, 2002.

[14] A.B. Kurzhanski and P. Varaiya, Reachability under state constraints - the ellipsoidal technique, In: *Proc. of the 15-th World Congress of IFAC, Barcelona, Spain*, Elsevier, Amsterdam, 2002.

[15] A.B. Kurzhanski and P. Varaiya, Reachability analysis for uncertain systems - the ellipsoidal technique. *Dynamics of Continuous, Discrete and Impulsive Systems, Ser. B*, **9** (3):347–367, 2002.

[16] H.S. Witsenhausen, Set of possible states of linear systems given perturbed observations, *IEEE Trans Aut. Control*, **13**:556–558, 1968.

[17] A.B. Kurzhanski, Differential games of observation, *Sov. Math Doklady,* **13** (6):1556–1560, 1972.

[18] F.C. Schweppe, *Uncertain Dynamic Systems*, Prentice Hall, Englewood Cliffs, NJ, 1973.

[19] M. Milanese et al., *Bounding Approaches to System Identification*, Plenum Press, 1995.

[20] J.S. Baras and A.B. Kurzhanski, Nonlinear filtering: The set-membership (bounding) and the H_∞ approaches, *Proceedings of the IFAC Nonlinear Control Systems (NOLCOS) Symposium*, Lake Tahoe, CA, Plenum Press, 1995.

[21] M.I. Gusev and A.B. Kurzhanski, Optimization of controlled systems with bounds on the controls and the phase coordinates, I. *Differentsialniye Uravneniya—Differential Equations*, Transl. from Russian, **7** (9):1591–1602, 1970.

[22] A.B. Kurzhanski and O.I. Nikonov, On the problem of synthesizing evolution equations for tubes of trajectories of synthesized control systems, *Russ. Acad. of Sciences Math. Doklady*, **48** (3): 606–611, 1994.

[23] A.B. Kurzhanski and T.F. Fillippova, On the theory of trajectory tubes: a mathematical formalism for uncertain dynamics, viability and control, *Advances in Nonlinear Dynamics and Control*, Ser. PSCT **17**:122–188, Birkhäuser Boston, 1993.

[24] A.B. Kurzhanski and I. Vályi, *Ellipsoidal Calculus for Estimation and Control*, SCFA, Birkhäuser Boston, 1997.

[25] L.S. Pontryagin, Linear differential games of pursuit., Mat. Sbornik, (Transl. to Engl. as Sbornik: Mathematics), **112**, 1980.

[26] A.B. Kurzhanski, Pontryagin's alternated integral in the theory of control synthesis, *Proc. Steklov Math. Inst.* (Engl. Transl.), **224**:234–248, 1999.

3

Stability of Hybrid Systems and Related Questions from Systems Biology*

C. Piazza[1] and B. Mishra[2]

[1] Dept. of Mathematics and Computer Science, University of Udine, Udine, Italy
[2] Courant Institute of Mathematical Science and School of Medicine, NYU, New York, U.S.A.
piazza@dimi.uniud.it, mishra@cs.nyu.edu

3.1 Prologue

Pravin Varaiya's research career is marked by an ever-expanding breadth of interests starting with classical areas of electrical and communication engineering, but frequently intersecting with fields as far apart as highway traffic systems, game theory and economics. Indirectly through his students, post-doctoral fellows, "mentees" and even others, who came in contact with him only in chance encounters, his intellectual reach has gone much further.

From the mid-1990's to the present, a research theme that Pravin Varaiya has explored deeply concerns with "hybrid automata." These are systems describing a discrete program in a continuous environment. The best natural example that comes to mind would be a description of developmental stages of an organism embedded inside an environment composed of a variety of biological macromolecules (DNA, RNA and protein) synthesizing, duplicating, modulating and degrading each other in a complex manner. The basic developmental program interacts with the environment through injuries, infection, immune interactions, mutations, diseases, aging and evolutionary processes. While unfortunately the asymptotic destinies of these systems and their components are degradation, death, and extinction, the transient behaviors of these hybrid automata remain infinitely fascinating to us for obvious reasons.

Consequently, even though hybrid automata of the kind that Pravin Varaiya explored were motivated by examples from complex engineered systems, there are many questions that he had raised in the engineering context that remain equally interesting also in the biological situation. In a paper that Pravin Varaiya wrote with Mikhail Kourjanski, they explored the question of how to characterize "stability of

* This work was supported by grants from NSF's Qubic program, NSF's ITR program, Defense Advanced Research Projects Agency (DARPA), Howard Hughes Medical Institute (HHMI) biomedical support research grant, the US Department of Energy (DOE), the US air force (AFRL), National Institutes of Health (NIH) and New York State Office of Science, Technology & Academic Research (NYSTAR).

hybrid systems." (See [16].) In this paper they studied a particular class of hybrid automata that are now called rectangular automata, and restricted their attention to the ones in which discrete states go through a loop and also contains an infinite trajectory starting from some state. Such a viable system was shown to be exactly characterized by rectangular systems with fixed point or infinite cycle.

Because of our biological motivations, we extend the notion to *functional hybrid automata* whose flow and reset conditions are based on real functions (and even further restricted to semi-algebraic functions when we seek algorithmic solutions). We are now able to ask similar questions about stability (rather simple in this case) and limit cycles.

In particular, we show that functional hybrid automata, which can be used to model biological systems, can be reduced to systems of

differential equations. As a consequence many results obtained in dynamical systems theory (e.g., Lyapunov's stability theorems and LaSalle invariance principle [17]) apply *mutatis mutandis*.

The chapter is organized as follows: we start with a brief but comprehensive overview of biological system models and one interesting example, the *circadian clock*, whose cyclic rhythm governs our daily function (Section 3.2), and follow it with a formal introduction to functional hybrid automata and the question of their stability (Section 3.3). We then focus on our technical approach involving a direct translation of a subclass of functional hybrid automata into systems of differential equations (Section 3.4), thus making our problem amenable to classical approaches. We place our work in the context of other related works (Section 3.5) and conclude in Section 3.6 with a discussion of how new challenges from systems biology may rely on the revolution that Pravin Varaiya and his colleagues started.

3.2 Biological System Models

The central dogma of biology translates easily to a mathematical formalism for biochemical processes involved in gene regulation. This principle states that biochemical information flow in cells is unidirectional—DNA molecules code information that gets transcribed into RNA, and RNA then gets translated into proteins. To model a regulatory system for genes, we must also include an important subclass of proteins (transcription activators), which also affects and modulates the transcription processes itself, thus completing the cycle. We can write down *kinetic mass-action* equations for the time variation of the concentrations of these species, in the form of a system of *ordinary differential equations* (ODE's) [10, 15, 24]. In particular, the transcription process can be described by equations of the *Hill* type, with its Hill coefficient n depending on the *cooperativity* among the transcription binding sites. If the concentration of DNA and RNA are denoted by M_x, M_y, etc., and those of proteins by P_x, P_y, etc., then the relevant equations are of the form:

$$\dot{M}_x = -k_1 M_x + k_3 \frac{1 + \theta P_y^n}{1 + P_y^n}, \tag{3.1}$$

$$\dot{P}_x = -k_2 P_x + k_4 M_x. \tag{3.2}$$

Each equation above is an algebraic differential equation consisting of two algebraic terms, a positive term, representing synthesis and a negative term, representing degradation. For both RNA and DNA the degradation is represented by a linear function; for RNA, synthesis through transcription is a highly nonlinear but a rational Hill-type function; and for proteins, synthesis through translation is a linear function of the RNA concentration. In the equation for transcription, when $n = 1$, the equations are called *Michaelis–Menten* equations; P_y denotes the concentration of proteins involved in the transcription initiation of the DNA, k_1 and k_2 are the forward rate constants of the degradation of RNA and proteins, respectively, k_3 and k_4 are the rate constants for RNA and protein synthesis and θ models the saturation effects in transcription.

If one knew all the species involved in any one pathway, the mass-action equations for the system could be expressed in the form

$$\dot{X}_i = f_i(X_1, X_2, \ldots, X_n), \qquad i = 1, 2, \ldots, n. \tag{3.3}$$

When the number of species becomes large, the complexity of the system of differential equations grows rapidly. Furthermore, the mathematics of the dynamical system becomes increasingly complex. The integrability of the system of equations, for example, depends on the algebraic properties of appropriate bracket operations [19, 20]. We can approximately describe the behavior of such a system using a *hybrid automata* [3, 21]. The discrete states of the hybrid system describe regimes of system behavior which are qualitatively different in terms of which species and reactions predominate, and so forth. The "flows," "invariants," "guards," and "reset" conditions can be approximated by algebraic systems and the decision procedures for determining various properties of these biological systems can be developed using the methods of symbolic algorithmic algebra. As we enlarge the scopes of the biological models by considering metabolic processes, signal transduction processes and subcellular biochemical processes that are specific to locations and transportation between cellular compartmentalizations, the challenges to the algorithmic complexity and approximability deepen the need for better algorithmic algebraic techniques. In the process, we are also forced to explore the connection among constructive approaches for differential algebra, commutative algebra, Tarski-algebra, etc.

As a simple illustrative example, where its limiting cyclic behavior is rather important, consider the following model of "circadian clock." A widely-studied model of the mechanism for circadian rhythm was first proposed by Goldbeter [14] in terms of the dynamics involved in the degradation of the period protein (PER) and took into account multiple phosphorylation of PER and the negative feedback exerted by PER on the transcription of the period (*per*) gene. Informally, the *per* gene transcribes its corresponding mRNA in the nucleus at a rate negatively governed by nuclear PER protein—more nuclear PER protein implies less *per* mRNA and vice versa. The transcribed *per* mRNA leaves the nucleus to get translated into PER protein, which after post-translational modifications (several successive phosphorylation steps) diffuses back into the nucleus—more *per* mRNA implies more nuclear PER protein and vice versa. All these effects can be expressed succinctly in the forms of the ODEs we have described earlier. This minimal biochemical model, supported by experimental observations, resulted in a better understanding of the limit cycle of the molecular dynamics inherent to circadian oscillation. The mathematical model, created from the Michaelis–Mentens type kinetic models, is a five-dimensional system of first-order-ODE's and involved algebraic rational functions of low degree.

A more detailed model takes into account the role played by the formation of a complex between the PER and TIM proteins, and requires considering a sequence of steps for TIM similar to the ones shown below. The more complex system is 10 dimensional and omitted from discussion. Including further evidence that the TIM light response is relevant to light-induced phase shifts of the circadian clock, and its modeling through discrete mode switches, bring us back to the realm of hybrid automata. While we do not describe such a complex model here, we do emphasize the fact that understanding the limiting behavior of hybrid models such as these are important if we wish to understand how light acts as a major environmental signal for the entrainment of circadian rhythms.

In the equations below: *per* mRNA, whose cytosolic concentration is denoted by M, is synthesized in the nucleus and transferred into the cytosol, where it is degraded; the rate of synthesis of PER is proportional to M. In order to take into account the fact that PER is multiply phosphorylated, while keeping the model as simple as possible, only three states of the protein are considered: unphosphorylated (P_0), monophosphorylated (P_1) and bisphosphorylated (P_2); P_N is the nuclear PER protein.

Crucial to the mechanism of oscillations in the model is the negative feedback exerted by the nuclear form P_N in the formation of the PER-TIM complex on the synthesis of *per* (and, in the more detailed model, also *tim*) mRNAs. The negative feedback is described by a Hill-type equation. The equations below are also somewhat idealized as they ignore the linear degradation terms characterized by a relatively small, nonspecific rate constant. This rate constant does not play an important role in the system's oscillatory behavior but ensures that a steady state exists even when degradations are inhibited.

$$\dot{M} = \nu_5 \frac{K_1^n}{(K_1^n + P_N^n)} - \nu_m \frac{M}{(K_m + M)}, \tag{3.4}$$

$$\dot{P_0} = \kappa_5 M - V_1 \frac{P_0}{(K_1 + P_0)} + V_2 \frac{P_1}{(K_2 + P_1)}, \tag{3.5}$$

$$\dot{P_1} = V_1 \frac{P_0}{(K_1 + P_0)} - V_2 \frac{P_1}{(K_2 + P_1)} - V_3 \frac{P_1}{(K_3 + P_1)} + V_4 \frac{P_2}{(K_4 + P_2)}, \tag{3.6}$$

$$\dot{P_2} = V_3 \frac{P_1}{(K_3 + P_1)} - V_4 \frac{P_2}{(K_4 + P_2)} - \kappa_1 P_2 + \kappa_2 P_N - \nu_d \frac{P_2}{(K_d + P_2)}, \tag{3.7}$$

$$\dot{P_N} = \kappa_1 P_2 - \kappa_2 P_N, \tag{3.8}$$

$$P_t = P_0 + P_1 + P_2 + P_N. \tag{3.9}$$

The mathematical model indicates that during oscillation, the peak in *per* mRNA precedes by several hours the total PER protein. The key insight was that multiple PER phosphorylation introduces time-delays which strengthen the negative feedback to produce oscillation. An algebraic analysis shows that the rhythm only occurs in a range bounded by two critical values of the "maximum rate of PER degradation." The same analysis can be used to show a "rough homeomorphism" between this high-dimensional system and a simpler two-dimensional van der Pol equation. The other critical parameter was found to be the "average rate of PER transport into the nucleus." The critical dependence of the limit cycle on the degradation parameter was a key for biologists to understand the altered period of *per* mutants.

In future, we may wish to study further extensions of this initial model: the PER-TIM model of Goldbeter, that incorporates the other protein TIM, whose dimerization with PER plays an important role in providing stability to the limit cycle; a better model of Tyson et al., that takes into account the detailed structure of PER-phosphorylation and inherent competition among several key processes and light-sensitivity of TIM. Many of these detailed models will require description in terms of hybrid modes. While these extended models are more complex, they appear to remain homeomorphic to simple van-der-Pol-like system, while adding to the stability of the over-all system.

Another interesting avenue to explore concerns the feasibility of synthetic cellular clocks. Is it feasible to design simple oscillating systems of a desired periodicity by genetic engineering in appropriate cell hosts? If so, such a system could be used as a stringent test system of our ability to model complex cellular pathways. We may conceive of a simple transcriptional feedback system, using temperature sensitive competitive inhibitors (so that clocks can be reset by temperature shifts) and fluorescent reporter systems (so that the phase of the cycle can be examined in individual cells and in the population). The advantages of such a system reside in its ease of manipulation, ease of monitoring, coupled to the use of genetic selection to explore unanticipated behaviors.

3.3 Hybrid Automata: Stability and Limit Cycles

3.3.1 Functional Hybrid Automata—Syntax

The notion of *Hybrid Automata* was first introduced in [4] as a model and specification language for systems consisting of a discrete program within a continuously changing environment. For our purpose, it is convenient to introduce a specialized notion of *functional hybrid automata*, whose flow and reset conditions are further restricted to functions over the reals.

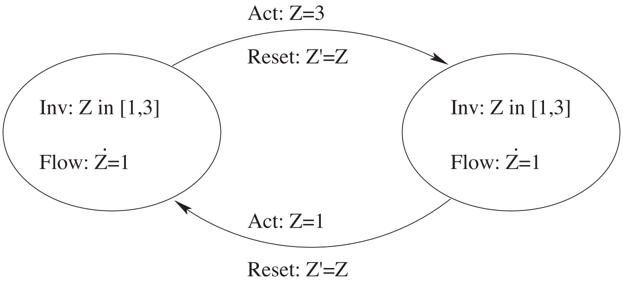
Following notations and conventions will be used through out the chapter: Capital letters $Z_1, \ldots, Z_k, Z'_1, \ldots, Z'_k$ will denote variables which range over $\mathbb{R}$. Moreover, Z will denote the vector of variables $\langle Z_1, \ldots, Z_k \rangle$; similarly, Z' will denote the vector $\langle Z'_1, \ldots, Z'_k \rangle$ and Z^n, the vector $\langle Z^n_1, \ldots, Z^n_k \rangle$. The variable T will be used for time, ranging over $\mathbb{R}^+$. The small letters $p, q, r, s, \ldots$ will denote k-dimensional vectors of real numbers.

Given a formula (function) φ we will use the notation $\varphi(Z_1, \ldots, Z_n)$ to stress the fact that the set of variables occurring in φ is included in $\{Z_1, \ldots, Z_n\}$. By extension, $\varphi(Z^1, \ldots, Z^n)$ will indicate that the variables of φ are included in the set of components of the vectors $Z^1, \ldots, Z^n$. Given a formula (function) $\varphi(Z^1, \ldots, Z^{i-1}, Z^i, Z^{i+1}, \ldots, Z^n)$, the formula (function) obtained by componentwise substitution of the elements of Z^i with the elements of p will be denoted by $\varphi(Z^1, \ldots, Z^{i-1}, p, Z^{i+1}, \ldots, Z^n)$. If the only variables in φ are the elements of Z^i, then after the substitution, the value of $\varphi(p)$ will be assumed to be available.

Definition 1 (Hybrid Automata). *A* hybrid automaton $H = (Z, \dot{Z}, Z', \mathcal{V}, \mathcal{E}, Inv, Flow, Act, Reset)$ *of dimension k has the following components:*

- $Z = \langle Z_1, \ldots, Z_k \rangle$, $\dot{Z} = \langle \dot{Z}_1, \ldots, \dot{Z}_k \rangle$, *and* $Z' = \langle Z'_1, \ldots, Z'_k \rangle$ *are vectors of variables ranging over* $\mathbb{R}$; Z *denotes the values of the continuous variables;* $\dot{Z}$ *denotes the first-order derivatives taken with respect to the time* $T \in \mathbb{R}^+$ *during continuous change;* Z' *denotes the values after a discrete jump;*
- $\langle \mathcal{V}, \mathcal{E} \rangle$ *is a finite directed graph; the nodes,* $\mathcal{V}$, *are called* control modes*, the edges,* $\mathcal{E}$, *are called* control switches*;*
- *Each vertex* $v \in \mathcal{V}$ *is labeled by the formulae* $Inv(v)(Z)$ *and* $Flow(v)(Z, \dot{Z})$*;* $Inv = \{Inv(v)(Z) \mid v \in \mathcal{V}\}$ *and* $Flow = \{Flow(v)(Z, \dot{Z}) \mid v \in \mathcal{V}\}$*;*
- *Each edge* $e \in \mathcal{E}$ *is labeled by the formulae* $Act(e)(Z)$ *and* $Reset(e)(Z, Z')$*;* $Act = \{Act(e)(Z) \mid e \in \mathcal{E}\}$ *and* $Reset = \{Reset(e)(Z, Z') \mid e \in \mathcal{E}\}$.

Example 1. Consider the following simple hybrid automaton "oscillating" between two values:



Starting in the control mode to the left Z grows at constant rate of 1. After 3 time units, upon reaching the value of $Z = 3$, it immediately jumps to the alternate control mode to the right, where Z now decreases until it reaches a value of $Z = 1$. Under this condition, it jumps back to the mode to the left. The automaton moves back and forth forever between these two modes.

Definition 2 (Functional Hybrid Automata and its Syntax). *A* functional hybrid automaton $H = (Z, \dot{Z}, Z', \mathcal{V}, \mathcal{E}, Inv, Flow, Act, Reset)$ *of dimension k is a hybrid automaton of the same dimension satisfying the following additional properties:*

- *Each invariant formula $Inv(v)$ characterizes a closed subset of $\mathbb{R}^k$;*
- *Each flow formula $Flow(v)$ is of the form $\dot{Z} = \psi(v)(Z)$ and the Cauchy problem $\dot{Z} = \psi(v)(Z)$ with initial condition $Z(0) = r$ has a unique solution for each r satisfying $Inv(v)$;*
- *For each r on the frontier set of the invariant $\delta(Inv(v))$ the solution $Z = \varphi(T)$ of the Cauchy problem $\dot{Z} = \psi(v)(Z)$ with initial condition $Z(0) = r$ further satisfies the following property:*

$$\forall \epsilon > 0, \varphi(\epsilon) \notin Inv(v);$$

- *Each activity formula $Act(\langle v,\ u\rangle)$ characterizes a subset of the frontier set $\delta(Inv(v))$;*
- *Each reset formula $Reset(e)$ is of the form $Z' = \rho(e)(Z)$, where $\rho(e)$ is an injective function.*

Example 2. The hybrid automaton of Example 1 is a functional hybrid automaton. For another example, see the hybrid automata proposed in [13] to model the Delta-Notch signaling process; these can be rewritten as functional hybrid automata by using closed invariant conditions. This change has no effect on the behaviors of the automata.

Henceforth, we restrict our discussions only to functional hybrid automata.

3.3.2 Hybrid Automata—Semantics

The semantics of functional hybrid automata can be defined in terms of execution traces. Traces are sequences of pairs with each pair consisting of a point and a control mode. Maximal traces are traces which cannot be extended.

Definition 3 (Functional Hybrid Automata and its Semantics).
Let $H = (Z,\ \dot{Z},\ Z',\ \mathcal{V},\ \mathcal{E},\ Inv,\ Flow,\ Act,\ Reset)$ be a hybrid automaton of dimension k.

A location *ℓ of H is a pair $\langle v,\ r\rangle$, where $v \in \mathcal{V}$ is a state and $r = \langle r_1,\ \ldots,\ r_k\rangle \in \mathbb{R}^k$ is an assignment of values for the variables of Z. An* admissible location *$\langle v, r\rangle$ is one for which $Inv(v)(r)$ holds.*

The continuous reachability transition relation *$\rightarrow_C$ between admissible locations is defined as follows:*

$$\begin{aligned}&\langle v, r\rangle \rightarrow_C \langle v, s\rangle\\ &\quad \textit{iff}\ \exists t > 0, \Big(f(0) = r \wedge f(t) = s \wedge \forall t' \in [0, t]\big(Inv(v)(f(t'))\big)\Big)\end{aligned}$$

where f is the solution of the Cauchy problem $\dot{Z} = \psi(v)(Z)$ with initial condition $Z(0) = r$.

The discrete reachability transition relation *$\rightarrow_D$ between admissible locations is defined as follows:*

$$\begin{aligned}&\langle v, r\rangle \to_D \langle u, s\rangle\\ &\quad \textit{iff}\ \ \langle v, u\rangle \in \mathcal{E} \wedge Act(\langle v, u\rangle)(r) \wedge s = \rho(\langle v, u\rangle)(r).\end{aligned}$$

A trace *of H is a sequence $\ell_0, \ell_1, \ldots, \ell_n, \ldots$ of admissible locations such that for each $i \geq 0$ either $\ell_i \to_C \ell_{i+1}$ or $\ell_i \to_D \ell_{i+1}$. A trace of H is* maximal *if it is not a proper prefix of another trace of H.*

Notice that our definition of trace is rather general: (1) the length of a trace can be either finite or infinite; (2) maximal traces can be of finite length.

3.3.3 Cyclic Traces

As discussed in Section 3.2, well-controlled robust periodic behavior is crucial to many biological systems: cell cycles, circadian clocks, cyclic expression patterns of segmentation clocks (e.g., the Delta/Notch signal transduction system), etc. When we model them with hybrid automata (see, e.g., [2, 13]) periodic behaviors correspond to cyclic traces. Hence, for a given hybrid automaton H, one may wish to determine:

Can this hybrid automaton H exhibit a cyclic trace? More formally, does there exist a trace of H taking the form $\ell_0, \ell_1, \ldots, \ell_n, \ell_0$ with $n \geq 0$?

There are only a handful of results that directly and explicitly address this question in the context of hybrid automata—efforts directed at the question of stability of cyclic traces are even rarer. In fact, since hybrid automata are *highly nondeterministic*, the problem of analyzing cyclic trace in the full generality is difficult. This limitation does not always apply, when it comes to biological systems. Hence, by modeling biochemical processes with functional hybrid automata, we try to limit the non-determinism, and exploit this property to study cyclic traces by suitably modifying results developed in the area of dynamic systems.

Let us begin by classifying cyclic traces in order to understand what makes them difficult to detect. If $\langle v, r\rangle$ is an admissible location of H, such that $\psi(v)(r) = 0$, then the trace $\langle v, r\rangle, \langle v, r\rangle$ is a cyclic trace of H. We call such a cyclic trace a *first gender cycle*.

Proposition 1. *Let H be a functional hybrid automaton. If for each vertex v the function $\psi(v)$ and the formula $Inv(v)$ are polynomials over the reals, then the existence of first gender cycles in H is decidable.*

Proof. For each vertex v consider the following first order formula

$$Inv(v)(Z) \wedge \psi(v)(Z) = 0.$$

The solutions of this formula are the points r such that $\langle v, r\rangle, \langle v, r\rangle$ is a first gender cycle. Since the satisfiability of the formula for any vertex v is decidable [22] and since the number of nodes v is finite, the first gender cycle problem is decidable, as claimed. □

We remark parenthetically that the result, shown above, can also be extended to *o-minimal* theories [12].

Assume further that $\psi(v)$ is such that a point r satisfying $Inv(v)$ exists and the solution of the Cauchy problem with initial condition $Z(0) = r$ is a periodic function with its image included in $Inv(v)$. Then the trace $\langle v, r\rangle$, $\langle v, r\rangle$ is a cyclic trace of H. We call a cyclic trace of this form a *second gender cycle*.

In order to detect second gender cycles it is necessary to study all the differential systems $\psi(v)$'s and check if they admit periodic solutions. Many results have been developed in the areas of dynamical systems and numerical analysis to detect periodic solutions and study their stability properties. Most of these results are built upon Lyapunov's stability theorems and LaSalle invariance principle [17]. Principles which apply to monotone systems have been recently studied in [6, 7].

In general, a cyclic trace can be $\langle v_0, r_0\rangle$, $\langle v_1, r_1\rangle$, $\ldots$, $\langle v_n, r_n\rangle$, $\langle v_0, r_0\rangle$ and may contain repeated copies of several discrete nodes internally, i.e., there may exist $i \neq j \leq n$ with $v_i = v_j$. We will call a cyclic trace of this form a *third gender cycle*, a detailed study of which is the key topic of this chapter. In particular, we aim to reduce this problem to a more classical problem: namely, that of studying periodic solutions of systems of differential equations, as in the case of second gender cycles.

In a trace there could be many consecutive continuous transitions as well as many consecutive discrete transitions. However, when we are looking for cyclic traces we can restrict our attention to traces in which each continuous transition is followed by a discrete transition.

Definition 4. *Let H be a functional hybrid automaton. A trace ℓ_0, ℓ_1, . . ., ℓ_n, . . . is said to be in* normal form *if it holds that $\ell_i \rightarrow_C \ell_{i+1}$ implies $\ell_{i+1} \not\rightarrow_C \ell_{i+2}$, for $i = 0, 1, \ldots$.*

Lemma 1. *Let H be a functional hybrid automaton. If H admits a cyclic trace, then it admits a cyclic trace in normal form.*

Proof. Let $T = \ell_0, \ldots, \ell_n, \ell_0$ be a cyclic trace of H. If $n = 0$, then the trace is already in normal form. Otherwise, $n > 0$, and each place the trace contains a subsequence of the form $\ell_i \rightarrow_C \ell_{i+1} \rightarrow_C \ell_{i+2}$ in T, we may replace it with $\ell_i \rightarrow_C \ell_{i+2}$. By repeated replacement of this kind, until it is no longer possible, we obtain a sequence which is a cyclic trace of H and is in normal form. □

3.4 From Deterministic Hybrid Automata to ODEs

In our definition of functional hybrid automata we limit the non-determinism to the following cases:

1. There exists a point which satisfies more than one invariant condition;
2. There exists a point which satisfies more than one activation condition.

Inside a vertex, the behavior of a functional hybrid automaton, by the second condition of Definition 2, is deterministic, as it imposes existence and uniqueness of the solution for each initial condition. Note, further, that when a functional hybrid automaton reaches the frontier of an invariant, it must jump immediately, since we imposed that the solutions immediately cross the frontier. Once the automaton decides (perhaps non-deterministically) which edge it may take, it uses a reset condition in a deterministic manner, as its reset condition is a function. Thus it remains to show that this second source of nondeterminism can be removed, and we can translate a functional hybrid automaton into a system of differential equations.

Definition 5 (Deterministic Functional Hybrid Automata). *Let $H = (Z, \dot{Z}, Z', \mathcal{V}, \mathcal{E}, Inv, Flow, Act, Reset)$ be a functional hybrid automaton. We say that H is* deterministic, *if for each vertex $v \in \mathcal{V}$ and for each pair of edges $e_1, e_2 \in \mathcal{E}$ with a common source vertex v we have*

$$Act(e_1) \cap Act(e_2) = \emptyset.$$

In our definition of deterministic functional hybrid automata there is still an apparent source of non-determinism and it is due to the fact that given a point $r \in \mathbb{R}^k$ it is possible to start from more than one location of the form $\langle v, r\rangle$.

Lemma 2. *Let H be a deterministic functional hybrid automaton and $\langle v, r\rangle$ be an admissible location of H. Then there exists one maximal trace in normal form $\ell_0, \ell_1, \dots, \ell_n, \dots$ with $\ell_0 = \langle v, r\rangle$.*

Proof. The sequence $\langle v, r\rangle$ is always a trace of H. Hence, it can be extended to at least one maximal trace Tr. As in the proof of Lemma 1, we can map Tr into a maximal trace in normal form which starts from $\langle v, r\rangle$.

We may derive a contradiction as follows, by assuming that there are two maximal traces in normal form, both starting from $\langle v, r\rangle$. We use $\ell_0, \ell_1, \dots, \ell_n, \dots$ and $\ell'_0, \ell'_1, \dots, \ell'_n, \dots$ to denote the two traces. Let i be the smallest index such that $\ell_i \neq \ell'_i$. It must be that $i > 0$. The following four cases must be considered:

1. $\ell_{i-1} \to_C \ell_i$ and $\ell_{i-1} = \ell'_{i-1} \to_C \ell'_i$;
2. $\ell_{i-1} \to_D \ell_i$ and $\ell_{i-1} = \ell'_{i-1} \to_D \ell'_i$;
3. $\ell_{i-1} \to_C \ell_i$ and $\ell_{i-1} = \ell'_{i-1} \to_D \ell'_i$;
4. $\ell_{i-1} \to_D \ell_i$ and $\ell_{i-1} = \ell'_{i-1} \to_C \ell'_i$.

Since the last two cases are essentially equivalent, we need consider only the first three cases. The first case can be ruled out since in each control mode the solutions of the differential equations are unique. The second case cannot occur since the activation conditions of H are disjoint and the reset are functional. Finally, the third case cannot occur because from $\ell_{i-1} = \ell'_{i-1} \to_D \ell'_i$ we conclude that $\ell_{i-1} = \langle u, s\rangle$ and s is on the frontier of $Inv(u)$, implying that the solution of $\dot{Z} = \psi(u)(Z)$ goes outside $Inv(u)$. This leads to the desired contradiction: it cannot be that $\ell_{i-1} \to_C \ell_i$. □

Given an admissible location ℓ we use the notation $Tr(\ell)$ to denote the maximal trace in normal form starting from ℓ.

Henceforth, we focus our attention on a deterministic functional hybrid automaton H. We aim to encode H into a system of differential equations whose solutions correspond to the traces of H. We start by encoding the nodes of $\mathcal{V}$. Let $|\mathcal{V}| = n$, and consider an ordering $[v_1, \ldots, v_n]$ of $\mathcal{V}$. We map each vertex of $\mathcal{V}$ to a point in $\mathbb{R}^n$ as follows:

$$\begin{aligned} \mu \, : \; & \mathcal{V} \to \mathbb{R}^n \\ & v_i \mapsto \langle 0, 0, \ldots, 1, \ldots, 0 \rangle, \end{aligned}$$

where 1 is in position i.

Let $R_1, \ldots, R_n, S_1, \ldots, S_n$ be $2n = |\mathcal{V}|$ fresh variables. Let also $W_1, \ldots, W_k$ be k fresh variables, where k is the dimension of H.

For each vertex $v \in \mathcal{V}$ we consider the system of differential equations $\Psi(v)$ on $\mathbb{R}^{2k+2n}$ defined as:

$$\begin{cases} \dot{Z} &= \psi(v)(Z) \\ \dot{W} &= \psi(v)(W) \\ \dot{R} &= 0 \\ \dot{S} &= 0. \end{cases}$$

This system describes the continuous evolution in v. The variables Z's and W's evolve as described in the mode v. The variables R's and S's do not evolve. They are used simply to encode the fact that the automaton is in vertex v.

Now we can glue together the systems of the different modes, i.e., we will encode the discrete jumps into differential systems. The basic ideas behind the encoding are as follows. Let us assume that we are in a point of the form $\langle z, z, \mu(v_i), \mu(v_i)\rangle$ and z satisfies $Act(\langle v_i, v_j\rangle)$. We use two time instants to jump from $\langle z, z, \mu(v_i), \mu(v_i)\rangle$ to $\langle \rho(\langle v_i, v_j\rangle)(z), \rho(\langle v_i, v_j\rangle)(z), \mu(v_j), \mu(v_j)\rangle$. During the first instant: Z moves on the segment between z and $\rho(\langle v_i, v_j\rangle)(z)$ at constant speed $\rho(\langle v_i, v_j\rangle)(z) - z$; W remains fixed since it is used to determine the constant speed at which Z moves; R moves on the segment between $\mu(v_i)$ and $\mu(v_j)$ at constant speed 1; S does not move so that it is clear that we are moving from $\mu(v_i)$ to $\mu(v_j)$ and not the converse. During the second instant we need to update W and S. Hence in this case, Z does not move; W moves on the segment between z and $\rho(\langle v_i, v_j\rangle)(z)$ at constant speed; R does not move; S moves on the segment between $\mu(v_i)$ and $\mu(v_j)$ at constant speed. In particular, to determine the segment on which W has to move we need to use the values of S and R after one instant (these encode the edge) and the value of Z after one instant (to determine the constant speed).

We start with the system for the first instant. For each edge $\langle v_i, v_j\rangle$ we consider the system $\Psi_1(\langle v_i, v_j\rangle)$ defined as:

$$\begin{cases} \dot{Z} &= \rho(\langle v_i, v_j\rangle)(W) - W \\ \dot{W} &= 0 \\ \dot{R} &= \mu(v_j) - \mu(v_i) \\ \dot{S} &= 0. \end{cases}$$

As far as the second instant is concerned, we proceed as follows: For each edge $\langle v_i, v_j\rangle$ we consider the system $\Psi_2(\langle v_i, v_j\rangle)$ defined as:

$$\begin{cases} \dot{Z} &= 0 \\ \dot{W} &= Z - \rho^{-1}(\langle v_i, v_j \rangle)(Z) \\ \dot{R} &= 0 \\ \dot{S} &= \mu(v_j) - \mu(v_i). \end{cases}$$

To conclude our construction, we collect and assemble the systems $\Psi(v)$, $\Psi_1(e)$, and $\Psi_2(e)$ combining the invariant and activation conditions of H. For a given formula $\gamma(Z)$ whose solutions denote a subset $G \subseteq \mathbb{R}^k$, we use $Op(\gamma)(Z)$ to denote the formula associated with the interior of G. Moreover, consider $\emptyset$, the system of differential equations which equates all the derivatives to 0. Let the system $\mathcal{H}$ be defined as follows:

$$\begin{cases} \Psi(v_i), & \text{if } Op(Init(v_i))(Z) \wedge R = S = \mu(v_i); \\ \Psi_1(\langle v_i, v_j \rangle), & \text{if } Act(\langle v_i, v_j \rangle)(W) \wedge S = \mu(v_i) \wedge R_j < 1; \\ \Psi_2(\langle v_i, v_j \rangle), & \text{if } R = \mu(v_j) \wedge S_i > 0; \\ \emptyset, & \text{otherwise.} \end{cases}$$

Notice that this construction uses $2n$ variables to encode the discrete part of the automaton. This construction avoids intersections of the solutions during the jumps. We could obtain the same result using only 6 variables, since given n points in $\mathbb{R}^3$ we can always connect them with n^2 non-intersecting curves.

We prove that the solutions of the system $\mathcal{H}$ and the traces of the deterministic functional hybrid automaton H *correspond* to each other, i.e., they are in a sense equivalent. We limit our arguments to traces of infinite length, since for cyclic solutions this suffices. The definitions and results can be modified appropriately to deal with traces of finite length.

Definition 6. *Let H be a functional hybrid automaton of dimension k with n control modes. Let $f : \mathbb{R}^+ \to \mathbb{R}^{n+k}$ be a function and $Tr = \ell_0, \ell_1, \ldots, \ell_m, \ldots$ be a trace of H of infinite length. We say that f and Tr* agree *if there exists an increasing sequence $t_0, t_1, \ldots, t_m, \ldots$ of positive reals such that for each i, $\ell_i = f(t_i)$.*

Theorem 1. *Let H be a deterministic functional hybrid automaton of dimension k and $\langle v, r \rangle$ be an admissible location of H such that $Tr(\langle v, r \rangle)$ has infinite length. The solution $Z = f^1_{\langle v,r \rangle}(t)$, $R = f^3_{\langle v,r \rangle}(t)$ of $\mathcal{H}$ with initial conditions $Z = W = r$ and $R = S = \mu(v)$ and the trace $Tr(\langle v, r \rangle)$ agree.*

Proof. We use $Z = f^1_{\langle v,r \rangle}(t)$, $W = f^2_{\langle v,r \rangle}(t)$, $R = f^3_{\langle v,r \rangle}(t)$, and $S = f^4_{\langle v,r \rangle}(t)$ to denote the solution of $\mathcal{H}$ with initial conditions $Z = W = r$ and $R = S = \mu(v)$. We have to define the sequence $t_0, t_1, \ldots, t_m, \ldots$ satisfying Definition 6. Let $Tr(\langle v, r \rangle)$ be of the form $\langle v, r \rangle$, $\langle w_1, s_1 \rangle$, ..., $\langle w_m, s_m \rangle$, ... We define $t_0 = 0$. The initial value clearly satisfies Definition 6. Let us assume inductively that we have defined t_0, ..., t_i satisfying Definition 6; we define t_{i+1} as follows:

- if $\langle w_i, s_i \rangle \to_D \langle w_{i+1}, s_{i+1} \rangle$, then $t_{i+1} = t_i + 2$;
- if $\langle w_i, s_i \rangle \to_C \langle w_{i+1}, s_{i+1} \rangle$, then $t_{i+1} = min\{t > t_i \mid f^1_{\langle w_i, s_i \rangle}(t) = s_{i+1}\}$.

In the first case we see that we still satisfy Definition 6, since $\langle w_i, s_i \rangle$ is in the activation region of $\langle w_i, w_{i+1} \rangle$ and after two time units the system $\mathcal{H}$ reaches the point reachable with the discrete jump. As far as the second case is concerned, we get the same conclusion as a consequence of the facts that we are considering autonomous systems and that the trace $Tr(\langle v, r \rangle)$ is in normal form (hence the next transition is discrete). □

Thus, we conclude that cyclic traces of H *agree* with periodic orbits of $\mathcal{H}$.

Corollary 1. *Let H be a deterministic functional hybrid automaton. H admits a cyclic trace if and only if $\mathcal{H}$ has a periodic orbit.*

Notice that if the second condition of Definition 2 fails, for example, because the flows can have either no solution or more than one solution, then Lemma 2 is false. Nonetheless, we can still construct $\mathcal{H}$ and prove correspondence between traces of H and solutions of $\mathcal{H}$.

3.5 Related Literature

To place the results described here in the context of a growing literature, we mention a few related results.

The closest in spirit to our results are those in [18]. There, hybrid automata are studied from a dynamical systems perspective. The paper rigorously proves necessary and sufficient conditions for existence, uniqueness, and continuity of traces. Under these assumptions, Lyapunov's theorem on stability via linearization and LaSalle's invariance principle are generalized to hybrid automata. While our notion of deterministic functional hybrid automata is intuitively similar to the notion of deterministic hybrid automata introduced in [18], there are many fundamental differences: we do not impose that the flows are globally Lipschitz continuous, but we assume that they have a unique solution for each initial condition; we impose on the resets an injectiveness condition. When the flows of a deterministic functional hybrid automaton H are globally Lipschitz continuous all the results proved in [18] apply to H. In the general case we can map H into the dynamic system $\mathcal{H}$ and try to directly apply stability and invariance results to $\mathcal{H}$.

In [11] hybrid systems are defined as sets of systems of differential equations. Which system has to be used is decided by the initial conditions and by a discrete control. On these hybrid systems, stability conditions are studied explicitly. The systems in [11] are not continuously linked in the following sense: when there is a switch in the discrete part, there is a jump in the continuous part, hence stability results for dynamic systems cannot be directly applied. The main difference with our construction is that we connect the flows continuously so that we get a piecewise defined dynamic system.

In [1] an affine hybrid automaton H is mapped into a new automaton $Bl(H)$ which has the same periodic orbits and equilibrium points, but no Zeno behaviors.

The basic idea behind the mapping is to split each control switch adding a new control mode and to introduce a time delay in the new modes. This is similar to what we do in our construction when we use 2 time instants for each edge crossing. In fact, we can prove that the Zeno behaviors of H corresponds to solutions of $\mathcal{H}$ in which the time flow is unbounded.

In [9] domains of convergence are studied by mapping systems of differential equations into discrete automata with an infinite number of states. By combining the construction we describe in this chapter with that defined in [9] we get a discretization method for hybrid automata. Relationships with other discretization methods (e.g., [5, 23]) remain to be analyzed.

3.6 Conclusion

Finally, we return to the biological questions that initiated this journey into the stability of hybrid automata. At present, we lack the ability to analyze all but the simplest regulatory structures composed of a handful of genes and we have no means of even intelligently conjecturing what universal principles unify biology. Our notions of biological robustness and arguments in its favor are often anecdotal, speculative and unsupported by data. For instance, there have been raging debates about the nature of the robustness exhibited by a circadian clock model that is composed of analogs of both PER and TIM, but also taking into account the reality that the copy number of PER-TIM complexes can only assume a small and random number. For instance, in the work of Naama Barkai and Stan Leibler [8], they speculate existence of an unmodeled hysteresis mechanism in circadian clock models to confer on it some degree of robustness. And yet, there are others, who using similar simulations, have argued that the original model is already robust as it is. Clearly, if the truth must be found, it will need formal methods that no amount of simulation can deliver. Pravin Varaiya's insights and instincts, buried among his results on engineering hybrid systems, may provide the methods we seek to solve such problems in systems biology.

References

[1] A. Aaron and S. Sastry, Blowing up affine hybrid systems, In *IEEE Conference on Decision and Control*, 473–478, Dec. 2004.

[2] R. Alur, C. Belta, F. Ivancic, V. Kumar, M. Mintz, G.J. Pappas, H. Rubin, and J. Schug, Hybrid modeling and simulation of biomolecular networks, In M.D. DiBenedetto and A.L. Sangiovanni-Vincentelli, Eds., *Hybrid Systems: Computation and Control*, LNCS **2034**:19–32, Springer-Verlag, 2001.

[3] R. Alur, C. Courcoubetis, N. Halbwachs, T.A. Henzinger, P.H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine, The Algorithmic analysis of hybrid systems, *Theoretical Computer Science*, **138** (1):3–34, 1995.

[4] R. Alur, C. Courcoubetis, T.A. Henzinger, and P.H. Ho, Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems, In R.L. Grossman, A. Nerode, A.P. Ravn and H. Richel, Eds., *Hybrid Systems*, LNCS **736**:209–229, Springer-Verlag, 1993.

[5] R. Alur, T. Dang, and F. Ivancic, Progress on reachability analysis of hybrid systems using predicate abstraction, In O. Maler and A. Pnueli, Eds., *Hybrid Systems: Computation and Control*, LNCS **2623**:4–19, Springer-Verlag, 2003.

[6] D. Angeli and E.D. Sontag, Monotone control systems, *IEEE Transaction on Automatic Control*, **48**:1684–1698, 2003.

[7] D. Angeli and E.D. Sontag, An analysis of a circardian model using the small-gain approach to monotone systems (I), In *IEEE Conference on Decision and Control*, 575–578, Dec. 2004.

[8] N. Barkai and S. Leibler, Circadian clocks limited by noise, *Nature*, **403**:267–268, 2000.

[9] R.W. Brockett, Pulses, periods, and cohomological terms in functional expansions, In *System Theory*, Springer-Verlag, 1993.

[10] A. Cornish-Bowden, *Fundamentals of Enzyme Kinetics* (3rd edition), Portland Press, 2004.

[11] R. DeCarlo, M. Branicky, S. Pettersson, and B. Lennartson, Perspectives and results on the stability and stabilizability of hybrid systems, *Proceedings of IEEE, Special Issue on Hybrid Systems*, **88** (7):1069–1082, 2000.

[12] L. Van Den Dries, Tame topology and o-minimal structures, *London Mathematical Society Lecture Note*, **248**, Cambridge University Press, 1998.

[13] R. Ghosh, A. Tiwari, and C. Tomlin, Automated symbolic reachability analysis; with application to delta-notch signaling automata, In O. Maler and A. Pnueli, Eds., *Hybrid Systems: Computation and Control (HSCC'03)*, LNCS **2623**:233–248, Springer-Verlag, 2003.

[14] A. Goldbeter, A Model for circadian oscillations in the drosophila period protein (PER), *Proc. Royal Soc. London B Biol. Sci.*, **261**:319–324, 1995.

[15] J.P. Keener and J. Sneyd, *Mathematical Physiology*, Springer-Verlag, 1998.

[16] M. Kourjanski and P. Varaiya, Stability of hybrid systems, In R. Alur, T.A. Henzinger, E.D. Sontag, Eds., *Hybrid Systems III: Verification and Control*, LNCS **1066**:413–423, Springer-Verlag, 1996.

[17] J.P. LaSalle, Stability theory for ordinary differential equations, *J. Diff. Equat.*, **4**:57–65, 1968.

[18] J. Lygeros, K.H. Johansson, S.N. Simić, J. Zhang, and S. Sastry, Dynamical properties of hybrid automata, *IEEE Transactions on Automatic Control*, **48** (1):2–14, 2003.

[19] B. Mishra, *Computational Differential Algebra*, 111–145, World-Scientific, Singapore, 2000.

[20] B. Mishra, A Symbolic approach to modeling cellular behavior, In S. Sahni, V.K. Prasanna and U. Shukla, Eds., *High Performance Computing*, LNCS **2552**:725–732, Springer-Verlag, 2002.

[21] A. Nerode and W. Kohn, Hybrid systems and constraint logic programming, In D.S. Warren, Ed., *International Conference on Logic Programming*, 18–24, MIT Press, 1993.

[22] A. Tarski, *A Decision Method for Elementary Algebra and Geometry*, Univ. California Press, 1951.

[23] A. Tiwari and G. Khanna, Series of abstraction for hybrid automata, In C.J. Tomlin and M. Greenstreet, Eds., *Hybrid Systems: Computation and Control*, LNCS **2289**:465–478, Springer-Verlag, 2002.

[24] E.O. Voit, *Computational Analysis of Biochemical Systems: A Practical Guide for Biochemists and Molecular Biologists*, Cambridge University Press, 2000.

Part II

System Theory and Design

4

Martingale Representation and All That

Mark H.A. Davis

Department of Mathematics
Imperial College London
London SW7 2AZ, UK
mark.davis@imperial.ac.uk

Summary. This chapter gives a survey of the theory of square-integrable martingales and the construction of basic sets of orthogonal martingales in terms of which all other martingales may be expressed as stochastic integrals. Specific cases such as Brownian motion, Lévy processes and stochastic jump processes are discussed, as are some applications to mathematical finance.

Key words: Stochastic integral, martingale, Lévy process, mathematical finance

4.1 Introduction

I have (so far) co-authored three papers with Pravin Varaiya [11],[12],[13]. The first one [11] concerns linear systems and is, I believe, the first paper anywhere to use weak solutions of stochastic differential equations in a control theory context. Our best-known paper is certainly [12] which treats stochastic control by martingale methods and gives a result sometimes referred to as the Davis–Varaiya maximum principle. The third paper [13] is the Cinderella of the set and has more or less disappeared without trace. It concerns the multiplicity of a filtration — an attempt to characterize the minimal number of martingales needed to represent all martingales as stochastic integrals. While our paper may have disappeared, interest in questions of martingale representation certainly has not. In particular the martingale representation property is equivalent to the very fundamental idea of 'complete markets' in mathematical finance. For this reason it seems time to rescue Cinderella from obscurity and invite her to the ball.

The setting for this chapter is the conventional filtered probability space of modern stochastic analysis. The reader can consult textbooks such as Øksendal [20], Protter [24] or Rogers and Williams [25] for background. We let $(\Omega, \mathcal{F}, P)$ be a complete probability space and $(\mathcal{F}_t)_{0\leq t\leq\infty}$ be a filtration satisfying *les conditions habituelles*. We assume $\mathcal{F}_\infty = \mathcal{F}$. We denote by $\mathcal{M}$ the set of square-integrable $\mathcal{F}_t$-martingales, i.e., $M \in \mathcal{M}$ if M is a martingale, $M_0 = 0$ and $\sup_t \mathbb{E}M_t^2 < \infty$. $\mathcal{M}_c$ is the set of $M \in \mathcal{M}$ such that the sample path $t \mapsto M(t, \omega)$ is continuous for almost all ω. $\mathcal{M}^{loc}, \mathcal{M}_c^{loc}$ denote the set of processes locally in $\mathcal{M}, \mathcal{M}_c$. A process X is *càdlàg* if its sample paths are right-continuous with left-hand limits; we write $\Delta X_s = X_s - X_{s-}$.

The next section introduces the L_2 theory of stochastic integration, while Section 4.3 describes the Hilbert space structure of the set of square-integrable martingales, including the Davis–Varaiya results [13] The standard Brownian motion case is covered in Section 4.4, while Section 4.5 describes the very striking Jacod–Yor theorem relating martingale representation to convexity properties of the set of martingale measures.

In recent years, Lévy processes have become widely used in mathematical finance and elsewhere, and in Section 4.6 we summarize results of Nualart and Schoutens giving a basis, the so-called Teugels martingales, for square-integrable martingales of a certain class of Lévy processes. If the Lévy process has no diffusive component and a Lévy measure of finite support, then it reduces to a rather simple sort of *stochastic jump process*. But martingale representation theorems are available for jump processes in much greater generality; we summarize the theory in Section 4.7. Concluding remarks are given in Section 4.8.

4.2 The Battle of the Brackets

As is well known, the quadratic variation of the Brownian path W_t over the interval $[0,t]$ is equal to t, and the second-order term in the Itô formula arises from the 'multiplication table' entry $(dW_t)^2 = dt$. When we move to more general martingales such as $M \in \mathcal{M}$ there are two candidates to replace 'dt'. The first is the 'angular brackets' process $< M >_t$ introduced by Kunita and Watanabe [21], the existence of which is a direct application of the Meyer decomposition theorem. Indeed, for $M \in \mathcal{M}$ the process M_t^2 is a submartingale and $< M >_t$ is defined as the unique predictable increasing process such that $< M >_0 = 0$ and $M_t^2 - < M >_t$ is a martingale. For $M, N \in \mathcal{M}$ the *cross-variation process* $< M, N >_t$ is defined by polarization:

$$< M, N >_t = \frac{1}{4}\left(< M+N >_t - < M-N >_t\right).$$

(In particular, $< M, M >_t = < M >_t$.) The process $< M >$ defines a positive measure on the predictable σ-field $\mathcal{P}$ in $(0,\infty) \times \Omega$ by the recipe $< M > (F) = \mathbb{E}\int_{(0,\infty)} \mathbf{1}_F(t,\omega) d < M >_t$. We denote by $L_2(< M >)$ the corresponding L_2 space, i.e., the set of predictable processes ϕ satisfying $\mathbb{E}\int_0^\infty \phi_s^2 d < M >_s < \infty$. The stochastic integral $\int \phi dM$ is characterized in very neat fashion for $\phi \in L_2(< M >)$ as the unique element of $\mathcal{M}$ satisfying

$$< \textstyle\int \phi dM, N >_t = \displaystyle\int_0^t \phi_s d < M, N >_s, \quad t \geq 0 \tag{4.1}$$

for all $N \in \mathcal{M}$. Let $\mathcal{I}$ be the set of simple integrands, i.e., processes ϕ of the form $\phi_t(\omega) = \sum_{i=1}^n Z_i(\omega)\mathbf{1}_{]S_i,T_i]}(t,\omega)$ for stopping times $S_i \leq T_i$ and bounded $\mathcal{F}_{S_i}$-

measurable random variables Z_i. For these integrands the stochastic integral is defined in the obvious way as

$$\int \phi dM = \sum_{i=1}^{n} Z_i(M_{T_i} - M_{S_i})$$

and we have the Itô isometry

$$\mathbb{E}\left(\int \phi dM\right)^2 = \mathbb{E}\int_{\mathbb{R}_+} \phi_t^2 d < M >_t .$$

The integral may now be defined by continuity on the closure of $\mathcal{I}$ in $L_2(< \ M \ >)$, which is equal to $L_2(< M >)$ itself, and then (4.1) is satisfied.

In recent times the angular bracket process has generally been superseded by the 'square brackets' process $[M]_t$ characterized by the following theorem[1].

Theorem 1. *For $M \in \mathcal{M}$ there exists a unique increasing process $[M]_t$ such that (i) $M_0 = 0$, (ii) $M_t^2 - [M]_t$ is a uniformly integrable martingale and (iii) $\Delta[M]_t = (\Delta M_t)^2$ for $t \in (0, \infty)$.*

If $M \in \mathcal{M}_c$ then $[M] =< M >$. Any $M \in \mathcal{M}$ can be decomposed into $M = M^c + M^d$ where $M^c \in \mathcal{M}_c$ and M^d is 'purely discontinuous' (further details below). Then

$$[M]_t =< M^c >_t + \sum_{s\leq t}(\Delta M)^2.$$

If $M \notin \mathcal{M}_c$ then $S_t = \sum_{s\leq t}(\Delta M)^2$ is an increasing process and, trivially, a submartingale, so it has the Meyer decomposition $S_t = U_t + V_t$ where U_t is a martingale and V_t is a predictable increasing process, the so-called *dual predictable projection* of S_t. We have

$$< M >_t =< M^c >_t + V_t,$$

and hence $[M]_t - < M >_t = U_t$, a uniformly integrable martingale.

Stochastic integrals can now be defined *à la* Kunita–Watanabe, but based on the square brackets process. We define

$$[M, N] = \frac{1}{4}([M + N] - [M - N]).$$

The appropriate class of integrands is $L_2([M])$, the set of predictable processes ϕ satisfying $\mathbb{E}\int_0^\infty \phi_s^2 d[M]_s < \infty$.

Theorem 2. *For $M \in \mathcal{M}$ and $\phi \in L_2([M])$ there is a unique element $\int \phi dM \in \mathcal{M}$ such that $[\int \phi dM, N]_t = \int_0^t \phi_s d[M, N]_s$ for all $N \in \mathcal{M}$. Further, $\Delta(\int \phi dM)_t = \phi_t \Delta M_t$.*

[1] See Rogers and Williams [25], Section IV.26

When restricted (as here) to predictable integrands, the integrals defined by (4.1) and by Theorem 2 are the same. Indeed, they clearly coincide on the set $\mathcal{I}$ of simple integrands and a monotone class argument shows that $L_2(<M>) = L_2([M])$. The main reason for preferring $[M]$ to $<M>$ is universality: $[M]$ is well-defined for every local martingale M, but not every local martingale is locally square integrable as required for the definition of $<M>$. A further disadvantage of $<M>$ is that it is not invariant under mutually absolutely continuous measure change. See page 123 of Protter [24] for a discussion of these points.

In spite of the above, for a discussion limited to $\mathcal{M}^{loc}$ the angular brackets process has some appeal. For instance, as we see below, (strong) orthogonality of M and N is equivalent to $<M,N>= 0$. It seems much more intuitive to say that two objects M, N are orthogonal when some bilinear form is equal to zero than when $[M, N]$ is a uniformly integrable martingale, which is the equivalent statement couched in square bracket terms. For these reasons we prefer to use $<M>$ in the following sections.

4.3 $\mathcal{M}$ as a Hilbert Space

The martingale convergence theorem implies that each $M \in \mathcal{M}$ is closed, i.e., there is an $\mathcal{F}_\infty$-measurable random variable M_∞ such that $M_t \to M_\infty$ in L_2 and, for each t, $M_t = \mathbb{E}[M_\infty|\mathcal{F}_t]$. Thus there is a one-to-one correspondence between $\mathcal{M}$ and $L_2(\Omega, \mathcal{F}, P)$, so that $\mathcal{M}$ is a Hilbert space under the inner product $M \cdot N = \mathbb{E}[M_\infty N_\infty]$. We say that $\mathcal{H}$ is a *stable subspace* of $\mathcal{M}$ if $M \in \mathcal{H} \Rightarrow \int \phi dM \in \mathcal{H}$ for all $\phi \in L_2(<M>)$. If $\mathcal{H}$ is a stable subspace, then so is

$\mathcal{H}^\perp = \{Y \in \mathcal{M} : Y \perp X \text{ for all } X \in \mathcal{H}\}$. The *stable subspace generated by* M is $\mathcal{S}(M) = \{\int \phi dM : \phi \in L_2(<M>)\}$. It turns out that $N \perp \mathcal{S}(M) \Leftrightarrow <M,N>= 0$. More generally, the stable subspace $\mathcal{S}(\mathcal{A})$ generated by a subset $\mathcal{A} \subset \mathcal{M}$ is the smallest closed, stable subspace containing $\mathcal{A}$. The set of continuous martingales $\mathcal{M}_c \subset \mathcal{M}$ is a stable subspace. Its orthogonal complement $\mathcal{M}_d$ is the set of 'purely discontinuous' martingales.

The Hilbert space structure gives us a way of obtaining an abstract 'martingale representation theorem', stated as follows.

Theorem 3. *Suppose $L_2(\Omega, \mathcal{F}, P)$ is separable. Then there exists a sequence M_i, $i = 1, 2, \ldots$ in $\mathcal{M}$ such that $<M_i, M_j>= 0$ for $i \neq j$, and any $X \in L_2(\Omega, \mathcal{F}, P)$ can be represented as*

$$X = \sum_{i=1}^{\infty} \int_0^\infty \phi_i(s) dM_i(s), \tag{4.2}$$

for some sequence $\phi_i \in L_2(<M_i>)$.

The construction of ϕ_i, M_i in (4.2) is straightforward. Let Y_i, $i = 1, 2, \ldots$ be a countable dense subset of $L_2(\Omega, \mathcal{F}, P)$, and set $M_1 = Y_1$. Now let $M_2(\infty)$ be the projection of Y_2 onto $\mathcal{S}(M_1)^\perp$ and define $M_2(t) = \mathbb{E}[M_2(\infty)|\mathcal{F}_t]$. Then $\mathcal{S}(M_1) \perp \mathcal{S}(M_2)$. We now define $M_3(\infty)$ as the projection of Y_3 onto $(\mathcal{S}(M_1) \oplus \mathcal{S}(M_2))^\perp$.

Continuing in this way we obtain a sequence of mutually orthogonal subspaces $\mathcal{S}(M_i)$ such that

$$L_2(\Omega, \mathcal{F}, P) = \bigoplus_{i=1}^{\infty} \mathcal{S}(M_i).$$

The representation (4.2) follows.

Theorem 3 shows that, as long as $L_2(\Omega, \mathcal{F}_\infty, P)$ is separable, there is always a countable sequence $M_1, M_2, \ldots \subset \mathcal{M}$ such that $\mathcal{M} = \mathcal{S}(M_1, M_2, \ldots)$. The question of interest is whether there is a *finite* set $\mathcal{A} = (M_1, \ldots, M_k)$ such that $\mathcal{M} = \mathcal{S}(\mathcal{A})$ and, if so, what is the minimum number k. Such a set is said to have the *predictable representation property*. This property has acquired a new significance in recent times in connection with mathematical finance, where $\mathcal{A}$ models a set of price processes of traded financial assets, integrands ϕ_t are trading strategies and stochastic integrals represent the *gain from trade* obtained by using the corresponding strategy. If a set of assets $\mathcal{A}$ is traded and these assets have the predictable representation property, then the market is *complete*, implying that there are uniquely defined prices for all derivative securities. See, for example, Elliott and Kopp [16] for an explanation of these ideas.

Davis and Varaiya considered the characterization of k in the 1974 paper [13]. Recall that the angular bracket process $< M >$ is identified with a positive measure on the predictable σ-field $\mathcal{P}$ in $(0, \infty) \times \Omega$ by defining

$$< M > (F) = \mathbb{E} \int_{(0,\infty)} \mathbf{1}_F(t, \omega) d < M >_t . \tag{4.3}$$

The notation $< M > \succ < N >$, or $< M > \approx < N >$, signifies that the measure $< N >$ is absolutely continuous with respect to, or equivalent to, $< M >$. We obtained the following results.

Theorem 4. *Suppose $\mathcal{M} = \mathcal{S}(M_1, M_2, \ldots, M_k)$ where $k \leq \infty$ ($k = \infty$ denotes that the M_i sequence is countably infinite). Then there exists a sequence $N_1, \ldots, N_l$ in $\mathcal{M}$, with $l \leq k$ and $N_1 = M_1$, such that*

(i) $\mathcal{S}(N_1, \ldots, N_l) = \mathcal{S}(M_1, \ldots, M_k)$;

(ii) $\mathcal{S}(N_i) \perp \mathcal{S}(N_j)$, $j \neq i$; and

(iii)$< N_1 > \succ < N_2 > \succ \cdots$.

Theorem 5. *Suppose $\mathcal{M} = \mathcal{S}(M_1, \ldots, M_k) = \mathcal{S}(N_1, \ldots, N_l)$ and that*

(i) $\mathcal{S}(M_i) \perp \mathcal{S}(M_j)$ and $\mathcal{S}(N_i) \perp \mathcal{S}(N_j)$ for $i \neq j$;

(ii)$< M_1 > \succ < M_2 > \succ \cdots$ and $< N_1 > \succ < N_2 > \succ \cdots$.

Then $< M_i > \approx < N_i >$ for all i, and in particular $k = l$.

These theorems imply that there is a unique minimal cardinality for any set of martingales with the predictable representation property. We call this number the *multiplicity* of the filtration $\mathcal{F}_t$ (following earlier work on the gaussian case by Cramér [6]).

4.4 The Brownian Case

This is the classic case, solved by K. Itô [17]. We take $(\Omega, \mathcal{F}, (\mathcal{F})_t, P, (W_t))$ to be the canonical Wiener space, so that W_t is Brownian motion and $\mathcal{F}_t$ is the natural filtration of W_t. Of course, W_t has continuous sample paths and $< W >_t = t$. The Lévy representation theorem states that Brownian motion is the *only* martingale with these properties.

Theorem 6. $X \in L_2(\Omega, \mathcal{F}_\infty, P)$ *if and only if*

$$X = \mathbb{E}X + \int_0^\infty \phi_t dW_t,$$

where ϕ_t *is an adapted process satisfying* $\mathbb{E}\int_0^\infty \phi_t^2 dt < \infty$.

The most straightforward proof of this theorem is the one given by Øksendal [20]. For $n = 1, 2, \dots$ let $\mathcal{G}_n = \sigma\{W_{k/2^n}, k = 1, 2, \dots, 2^{2n}\}$. Then $\mathcal{G}_n$ is increasing and $\bigvee_1^\infty \mathcal{G}_n = \mathcal{F}_\infty$. It follows from this and the martingale convergence theorem that if $X \in L_2(\Omega, \mathcal{F}_\infty, P)$, then $X_n \to X$ in L_2 where $X_n = \mathbb{E}[X|\mathcal{G}_n]$. The theorem is therefore proved if we can 'represent' X_n, which takes the form $X_n = h(W_{t_1}, \dots, W_{t_m})$ for some Borel function $h : \mathbb{R}^m \to \mathbb{R}$. X_n can be approximated in L_2 in the standard way by random variables $\tilde{X}_n = \tilde{h}(W_{t_1}, \dots, W_{t_m})$ in which $\tilde{h}$ is a smooth function of compact support. A stochastic integral formula for $\tilde{X}_n$ can be written down in a fairly explicit way, just by using the Itô formula and elementary properties of the heat equation. See Davis [9] or Exercise 4.17 of Øksendal [20] for details of this construction.

A very neat alternative proof was devised by Dellacherie [14] (see also Davis [7]). The theorem is equivalent to the implication $X \in \mathcal{S}(W)^\perp \Rightarrow X = 0$ a.s. Suppose $X \perp \mathcal{S}(W)$, let $\tau_n = \inf\{t : |X_t| \geq 1/n\}$ and define

$$\Lambda_t^n = 1 + \frac{1}{2n} X_{t \wedge \tau_n}.$$

Since all martingales of the Brownian filtration are continuous,[2] $\Lambda_\infty^n > 0$ a.s. and we define a measure Q_n equivalent to P by $dQ_n/dP = \Lambda_\infty^n$. Now $\Lambda^n - 1 \in \mathcal{S}(W)^\perp$, so that $W\Lambda^n$ and is a P-martingale, implying that W is a Q_n-martingale and hence (by the Lévy theorem) a Q_n-Brownian motion. Thus Q_n and P coincide on $\mathcal{F}_\infty$, implying that $X_{\tau_n} = 0$ a.s. and therefore that $X = 0$ a.s.

4.5 The Jacod–Yor Theorem

In Theorems 4 and 5 we thought of the predictable representation property as being a characteristic of the filtration $\mathcal{F}_t$. Alternatively, we can think of this property

[2] We need to establish this property without appealing to the representation theorem! In [7], measure change arguments are used twice, first to establish continuity of martingales and then, as here, to get the representation property. See Section V.4 of Revuz and Yor's *Continuous Martingales and Brownian Motion* for another direct proof of the continuity property.

in relation to the *measure* P in the underlying probability triple $(\Omega, \mathcal{F}, P)$. The argument given at the end of the last section gives a hint as to why considering alternative measures might be a fruitful thing to do.

For $\mathcal{A} \subset \mathcal{M}$, denote by $\mathbb{M}(\mathcal{A})$ the set of probability measures Q on $(\Omega, \mathcal{F})$ such that each $M \in \mathcal{A}$ is a square-integrable Q-martingale. Clearly, $\mathbb{M}(\mathcal{A})$ is a convex set. $Q \in \mathbb{M}(\mathcal{A})$ is an *extreme point* if $Q = \lambda Q_1 + (1-\lambda)Q_2$ with $Q_1, Q_2 \in \mathbb{M}(\mathcal{A})$ implies $\lambda = 0$ or 1.

Theorem 7 (Jacod–Yor [19]). *Let $\mathcal{A}$ be a subset of $\mathcal{M}$ containing constant martingales. Then $\mathcal{S}(\mathcal{A}) = \mathcal{M}$ if and only if P is an extreme point of* $\mathbb{M}(\mathcal{A})$.

This is Theorem IV.57 of Protter [24]. The proof is too lengthy to describe in detail here, but we can show why extremality is a necessary condition. Indeed, suppose P is not an extreme point; then $P = \lambda Q_1 + (1-\lambda)Q_2$ for some $Q_1, Q_2 \in \mathbb{M}(\mathcal{A})$ and $\lambda \in]0,1[$. Let $L_t = \mathbb{E}[dQ_1/dP|\mathcal{F}_t]$. Then $1 = \lambda L_\infty + (1-\lambda)dQ_2/dP \geq \lambda L_\infty$, so $L_\infty \leq \lambda^{-1}$ a.s. Hence $\tilde{L}_t = L_t - L_0 \in \mathcal{M}$. If $X \in \mathcal{S}(\mathcal{A})$ then X is a Q_1-martingale, so for any $s < t$ and bounded $\mathcal{F}_s$-measurable H,

$$\mathbb{E}_P[X_t L_t H] = E_P[X_t L_\infty H] = \mathbb{E}_{Q_1}[X_t H] = \mathbb{E}_{Q_1}[X_s H] = \mathbb{E}_P[X_s L_s H],$$

so XL is a P-martingale. Hence $X\tilde{L}$ is a martingale, so that $< X, \tilde{L} >= 0$. Since X is arbitrary, $\tilde{L} \perp \mathcal{S}(\mathcal{A})$, so it cannot be the case that $\mathcal{S}(\mathcal{A}) = \mathcal{M}$. Note that this argument is very close to Dellacherie's proof of the Brownian representation theorem given above in Section 4.4.

Of course, P is an extreme point of $\mathbb{M}(\mathcal{A})$ if $\mathbb{M}(\mathcal{A}) = \{P\}$, and this is the way Theorem 7 is generally used in mathematical finance. The 'first fundamental theorem' of mathematical finance states (very roughly) that absence of arbitrage opportunities is equivalent to existence of an equivalent martingale measure (EMM), i.e., a measure Q under which each $M \in \mathcal{A}$ is a martingale, where $\mathcal{A}$ is the set of price processes of traded assets in the market model. The 'second fundamental theorem' states that the market is complete if there is a *unique* EMM. But this is (modulo technicalities) just an application of the Jacod–Yor theorem, since 'completeness' is tantamount to the predictable representation property. Thus the Jacod–Yor theorem is one of the cornerstones of modern finance theory.

4.6 Lévy Processes

Lévy processes have been around since — obviously — the original work of Paul Lévy in the 1930s and 1940s, but have recently been enjoying something of a renaissance, fueled in part by the need for asset price models in finance that go beyond the standard geometric Brownian motion model. The quickest introduction is still Section I.4 of Protter [24] (carried over from the 1990 first edition), but some excellent textbooks have recently appeared, including Applebaum [1], Bertoin [3], Sato [26] and Schoutens [27]. There is also an informative collection of papers edited by Barndorff-Nielsen *et al.* [2].

A process $X = (X_t, t \geq 0)$ is a *Lévy process* if it has stationary independent increments, $X_0 = 0$ and X_t is continuous in probability. The probability law of X is determined by the 1-dimensional distribution of X_t for any $t > 0$, and this has characteristic function

$$\mathbb{E}\left[e^{iuX_t}\right] = e^{t\psi(u)}$$

where $\psi(u)$ is the log characteristic function of an infinitely-divisible distribution. The Lévy-Khinchin formula shows that ψ must take the form

$$\psi(u) = \mathrm{i}au - \frac{1}{2}\sigma^2 u^2 + \int_{-\infty}^{\infty} \left(e^{\mathrm{i}ux} - 1 - \mathrm{i}ux\mathbf{1}_{|x|<1}\right) \nu(dx),$$

where a, σ are constants and the *Lévy measure* ν is a measure on $\mathbb{R}$ such that $\nu(\{0\}) = 0$ and

$$\int_{\mathbb{R}} (1 \wedge x^2)\nu(dx) < \infty. \tag{4.4}$$

If $\nu \equiv 0$ then X is Brownian motion with drift a and variance parameter σ^2. The interpretation of ν is that if $A \subset \mathbb{R}$ is bounded away from 0 and $N_A(t)$ denotes the counting process $N_A(t) = \sum_{s \leq t} \mathbf{1}_{(\Delta X_s \in A)}$, then N_A is a Poisson process with rate $\nu(A)$. The integrability condition on ν implies that the total jump rate is generally infinite and jumps occur at a dense set of times, although, for any $\epsilon > 0$, jumps of size greater than ϵ occur at isolated times. Protter [24] shows that every Lévy process has a càdlàg version. The sample paths have bounded variation if and only if $\sigma = 0$ and

$$\int_{\mathbb{R}} (1 \wedge |x|)\nu(dx) < \infty. \tag{4.5}$$

The L_2 theory of Lévy processes is explored in a beautiful little paper by Nualart and Schoutens [23], on which this section is mainly based. The condition on the Lévy measure is

$$\int_{\mathbb{R}\setminus(-\epsilon,\epsilon)} e^{\lambda|x|}\nu(dx) < \infty \quad \text{for some } \epsilon, \lambda > 0. \tag{4.6}$$

Condition (4.6) implies that X_t has moments of all orders, and that polynomials are dense in $(\mathbb{R}, \mu_t)$, where μ_t is the distribution of X_t. A convenient basis for martingale representation is provided by the so-called *Teugels martingales*, defined as follows. We set $X_t^{(1)} = X_t$ and for $i \geq 2$,

$$X_t^{(i)} = \sum_{0<s\leq t} (\Delta X_s)^i.$$

Then $\mathbb{E}X_t^{(i)} = m_i t$ where $m_1 = a$ and $m_i = \int_{\mathbb{R}} x^i \nu(dx)$ for $i \geq 2$. The Teugels martingales are

$$Y_t^{(i)} = X_t^{(i)} - m_i t, \quad i = 1, 2, \ldots,$$

the compensated power jump process of order i. Let $\mathcal{T}$ denote the set of linear combinations of the $Y^{(i)}$. The angular brackets processes associated with the Teugels martingales are

$$< Y^{(i)}, Y^{(j)} >_t = \left(m_{i+j} + \sigma^2 \mathbf{1}_{(i=j=1)}\right) t. \tag{4.7}$$

Let $\mathcal{R}$ be the set of polynomials on $\mathbb{R}$ endowed with the scalar product

$$\ll p, q \gamma = \int_{\mathbb{R}} p(x)q(x)x^2\nu(dx).$$

Then we see that $x^{i-1} \leftrightarrow Y^{(i)}$ is an inner product preserving map from $\mathcal{R}$ to $\mathcal{T}$, so any orthogonalization of $\{1, x, x^2, \ldots\}$ gives a set of strongly orthogonal martingales in $\mathcal{T}$. In particular we can find strongly orthogonal martingales $H^{(i)} \in \mathcal{T}, i = 1, 2 \ldots$ of the form

$$H^{(i)} = Y^{(i)} + a_{i,i-1}Y^{(i-1)} + \ldots + a_{i,1}Y^{(1)}.$$

In view of (4.7) the measures associated with the compensators $< H^{(i)} >$ by (4.3) are all proportional to the product measure $dt \times dP$ and hence these measures are all equivalent (as long as $H^{(i)} \neq 0$).

Theorem 8. *The set $\{H^{(1)}, H^{(2)}, \ldots\}$ has the predictable representation property, i.e., any $F \in L_2(\Omega, \mathcal{F}_\infty, P)$ has the representation*

$$F = \mathbb{E}F + \sum_{i=1}^{\infty} \int_0^{\infty} \phi_i(t) dH_t^{(i)}$$

for some predictable processes ϕ_i such that

$$\mathbb{E} \int_0^{\infty} \phi_i^2(t)dt < \infty.$$

The proof given in Nualart and Schoutens [23] proceeds by noting that polynomials of the form $X_{t_1}^{k_1}(X_{t_2} - X_{t_1})^{k_2} \ldots (X_{t_n} - X_{t_{n-1}})^{k_n}$ are dense[3] in $L_2(\Omega, \mathcal{F}_\infty, P)$, and obtaining a representation of these polynomials using stochastic calculus.

An interesting special case is as follows.

Corollary 1. *Suppose $\sigma = 0$ and that the Lévy measure ν has finite support $\{a_1, a_2, \ldots, a_n\}$. Then $\mathcal{A} = \{H^{(1)}, H^{(2)}, \ldots, H^{(n)}\}$ has the predictable representation property.*

This is equivalent to saying that, under the stated condition, $H^{(k)} \equiv 0$ for $k > n$. This fact is essentially due to non-singularity of the Vandermonde matrix

$$\begin{bmatrix} 1 & a_1 & a_1^2 & \ldots & a_1^{n-1} \\ 1 & a_2 & a_2^2 & \ldots & a_2^{n-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & a_n & a_n^2 & \ldots & a_n^{n-1} \end{bmatrix}.$$

It follows from Theorem 4 and Theorem 5 that n is the minimum number of martingales having the predictable representation property in this case.

[3] Incidentally, this shows that $L_2(\Omega, \mathcal{F}_\infty, P)$ is separable.

4.7 General Jump Processes

There is a simpler way to look at the case described above in Corollary 1. Indeed, we can write the process X_t as

$$X_t = a_1 N_1(t) + \cdots + a_n N_n(t),$$

where the processes $N_i(t)$, defined by $N_i(t) = \sum_{s \le t} \mathbf{1}_{(\varDelta X_t = a_i)}$, are independent Poisson processes with rates $\lambda_i = \nu(\{a_i\})$. We have

$$\mathcal{S}(H^{(1)}, H^{(2)}, \ldots, H^{(n)}) = \mathcal{S}(\tilde{N}_1, \ldots, \tilde{N}_n),$$

where $\tilde{N}_i$ is the compensated point process $\tilde{N}_i(t) = N_i(t) - \lambda_i t$, so the predictable representation property can equally well be expressed in terms of integrals with respects to the $\tilde{N}_i$ processes.

However, results of this sort are true in much greater generality: the representation of martingales of jump processes was investigated in a series of papers in the 1970s by, *inter alia*, Jacod [18], Boel, Varaiya and Wong [4], Chou and Meyer [5], Davis [8],[10] and Elliott [15]. In particular we do not need the Markov property, and can allow for processes taking values in much more general spaces. We follow the description in the Appendix of [10].

A *stochastic jump process* is a right-continuous piecewise-constant process X_t taking values in $\varXi \cup \{\varDelta\}$, where $\varXi$ is a Borel space and $\varDelta$ an isolated 'cemetery state'. We take a point $Z_0 \in \varXi$ and on some probability space $(\varOmega, \mathcal{F}, P)$ we define a countable sequence of pairs of random variables $(S_k, Z_k) \in \varUpsilon$, $k = 1, 2, \ldots$, where $\varUpsilon = \mathbb{R}_+ \times \varXi$. We then define $T_k = \sum_{i=1}^{k} S_i$ and $T_\infty = \lim_{t\to\infty} T_k$ and define the sample path X_t by

$$X_t = \begin{cases} Z_0 & 0 \le t < T_1 \\ Z_k & T_k \le t < T_{k+1} \\ \varDelta, & t \ge T_\infty \end{cases}.$$

The law of (X_t) can be specified by giving a family of conditional distributions $\mu_k : \varUpsilon^{k-1} \to \text{Prob}(\varUpsilon)$ (here $\varUpsilon^0 = \emptyset$). For simplicity of exposition, let us assume that $T_\infty = \infty$ a.s. We let $\mathcal{F}_t^X = \sigma\{X_s, 0 \le s \le t\}$ be the completed natural filtration.

For $A \in \mathcal{B}(\varXi)$ let

$$p(t, A) = \sum_{T_i \le t} \mathbf{1}_{(X_{T_i} \in A)},$$

and let $\tilde{p}(t, A)$ be the predictable compensator of p, easily defined in terms of the family of transition measures μ_k, such that $q(t \wedge T_k, A) = p(t \wedge T_k, A) - \tilde{p}(t \wedge T_k, A)$ is a martingale for each k, so $q(t, A)$ is a local martingale. Stochastic integrals with respect to q are defined *pathwise* by

$$M_t^g = \int_0^t g(s, x, \omega) q(ds, dx) = \int_0^t g(s, x, \omega) p(ds, dx) - \int_0^t g(s, x, \omega) \tilde{p}(ds, dx).$$

The appropriate class of integrands is

$$L_1^{loc}(p) = \left\{ g : \ g \text{ is predictable and } \int_0^\infty |g|\mathbf{1}_{t\le\tau_n} dp < \infty \right\}.$$

Here τ_n is a sequence of stopping times $\tau_n \uparrow \infty$ a.s. The martingale representation theorem is the following.

Theorem 9. *M_t is a local $\mathcal{F}_t^X$-martingale if and only if $M_t = M_t^g$ for all t a.s. for some $g \in L_1^{loc}(p)$.*

This is Theorem A5.5 of Davis [10]. The proof is a more-or-less bare hands calculation using methods initiated by Dellacherie and by Chou-Meyer [5]. An L_2 version is given by Elliott [15].

4.8 Concluding Remarks

Martingale representation has been a recurring theme in stochastic analysis ever since the pioneering work of K. Itô [17] for the Brownian filtration. The results have proved to be of key importance in several application areas, for example non-linear filtering and mathematical finance, and continue to be the inspiration for further developments, most particularly in connection with Malliavin's calculus on Wiener space (see Nualart [22]). We hope the reader will find this short survey useful in providing some background and context for this continually fascinating corner of stochastic analysis.

References

[1] D. Applebaum, *Lévy Processes and Stochastic Calculus*, Cambridge University Press, 2004.

[2] O. Barndorff-Nielsen, T. Mikosch and S.I. Resnick, Eds., *Lévy Processes: Theory and Applications*, Birkhäuser, 2001.

[3] J. Bertoin, *Lévy Processes*, Cambridge University Press, 1998.

[4] R. Boel, P. Varaiya, and E. Wong, Martingales on jump processes I: representation results, *SIAM J. Control & Optimization*, **13**:999–1021, 1975.

[5] C.S. Chou and P.-A. Meyer, Sur la répresentation des martingales comme intégrales stochastiques dans les processus ponctuels, In *Séminaire de Probabilités IX*, Lecture Notes in Mathematics **465**. Springer-Verlag, 1975.

[6] H. Cramér, Stochastic processes as curves in Hilbert space, *Th. Prob. Appls*, **5**:195–204, 1965.

[7] M.H.A. Davis, Martingales of Wiener and Poisson processes, *J. Lon. Math. Soc. (2)*, **13**:336–338, 1976.

[8] M.H.A. Davis, The representation of martingales of jump processes, *SIAM J. Control & Optimization*, **14**:623–638, 1976.

[9] M.H.A. Davis, The representation of functionals of diffusion processes as stochastic integrals, *Trans. Cam. Phil. Soc.*, **87**:157–166, 1980.

[10] M.H.A. Davis, *Markov Models and Optimization*, Chapman and Hall, 1993.

[11] M.H.A. Davis and P. Varaiya, Information states for linear stochastic systems, *J. Math. Anal. Appl.*, **37**:384–402, 1972.

[12] M.H.A. Davis and P. Varaiya, Dynamic programming conditions for partially-observed stochastic systems, *SIAM J. Control*, **11**:226–261, 1973.

[13] M.H.A. Davis and P. Varaiya, On the multiplicity of an increasing family of sigma-fields, *Ann. Prob.*, **2**:958–963, 1974.

[14] C. Dellacherie, Intégrales stochastiques par rapport aux processus de Wiener ou de Poisson, In *Séminaire de Probabilités VIII*, Lecture Notes in Mathematics **381**, Springer-Verlag, 1973. [Correction dans *SP IX*, LNM 465.]

[15] R.J. Elliott, Stochastic integrals for martingales of a jump process with partially accessible jump times, *Z. Wahrscheinlichkeitstheorie ver. Geb*, **36**:213–266, 1976.

[16] R.J. Elliott and P.E. Kopp, *Mathematics of Financial Markets*, Springer-Verlag, 2nd edition, 2004.

[17] K. Itô, Multiple Wiener integral, *J. Math. Soc. Japan*, **3**:157–169, 1951.

[18] J. Jacod, *Calcul Stochastique et Problèmes de Martingales*, Lecture Notes in Mathematics **714**, Springer-Verlag, 1979.

[19] J. Jacod and M. Yor, Etudes des solutions extrémales et répresentation intégrale des solutions pour certains problèmes de martingales, *Z. Wahrscheinlichkeitstheorie ver. Geb.*, **38**:83–125, 1977.

[20] B. Øksendal, *Stochastic Differential Equations*, Springer-Verlag, 6th edition, 2004.

[21] H. Kunita and S. Watanabe, On square integrable martingales, *Nagoya Math. J.*, **30**:209–245, 1967.

[22] D. Nualart, *The Malliavin Calculus and Related Topics*, Springer-Verlag, 1995.

[23] D. Nualart and W. Schoutens, Chaotic and predictable representations for Lévy processes, *Stoch. Proc. Appl.*, **90**:109–122, 2000.

[24] P.E. Protter, *Stochastic Integration and Differential Equations*, Springer-Verlag, 2nd edition, 2004.

[25] L.C.G. Rogers and D. Williams, *Diffusions, Markov Processes and Martingales*, Volume II, Cambridge University Press, 2nd edition, 2000.

[26] K. Sato, *Lévy Processes and Infinitely Divisible Distributions*, Cambridge University Press, 1999.

[27] W. Schoutens, *Lévy Processes in Finance: Pricing Financial Derivatives*, John Wiley, 2003.

5

Engineering Education: A Focus on Systems

Edward A. Lee

Department of Electrical Engineering and Computer Science
University of California at Berkeley
Berkeley, CA 94720 USA
eal@eecs.berkeley.edu

5.1 Introduction

Engineers have a major advantage over scientists. For the most part, the systems we analyze are of our own devising. It has not always been so. Not long ago, the principle objective of engineering was to coax physical materials to do our bidding by leveraging their intrinsic physical properties. The discipline was one of "applied science." Today, a great deal of engineering is about coaxing abstractions that we have invented. The abstractions provided by microprocessors, programming languages, operating systems, and computer networks are only loosely linked to the underlying physics of electronics.

The rapid improvements in the capabilities of electronics during the last half of the 20-th century are, in part, the reason for this separation. The physical constraints imposed by limited memory, processing speed, and communication bandwidth appeared to evaporate with each new generation of computers. What appeared to one generation as luxuriously inefficient abstractions became the bread and butter of the next generation. The separation of "computer science" from "electrical engineering" is both a consequence and a cause, fueling the separation and reflecting it at the same time.

At the same time, the systems science that was incubated in the study of electronic circuits (control systems, communications theory, and signal processing) has also become more abstract. Although these disciplines were created by true "electrical engineers" ("true" means that they were engaged with electrical systems), many of the practitioners today rarely encounter electricity directly. Their techniques are often realized in "embedded" software, ironically building on the abstractions that are only loosely connected to the electronics that their theory originally helped to create. The theories, however, have not adapted as well as one might hope to the world of software. Perhaps these theories remain too wedded to their physical heritage.

Computer scientists lament that the engineers who write embedded software use so few of the beautiful abstractions they have built. They write their code in C (or even in assembly code), using low-level (less abstract) mechanisms. They ignore advances in object-oriented design, memory management, operating systems, and programming languages, and instead directly interact with memory-mapped registers that set up timer interrupts, provide interrupt service routines, and build application-specific task schedulers. Wouldn't it be nice if they would just learn to use the modern technology, and set up instead an HTTP server in Java? Or a peer-to-peer network of embedded sensor and actuator components that discover each other's capabilities via JXTA? The problem is that the modern technology does not talk about the properties of the system that they have to control, such as timing.

On the other hand, the information technology revolution of the late 20-th century was greatly accelerated by advancing computing abstractions. The Internet and the Web are not principally electronic systems. They are conceptual frameworks. The financial, economic, and social systems built on top of them have transformed our cultural landscape. But there is a weakness. While the computing abstractions we have built are extremely well suited to the management of information, their very divorce from the physics makes them less well suited to the management of our physical environment. This is the key reason that these abstractions have had less impact in embedded software.[1]

It seems likely that embedded computing is the next transformational revolution. Although it may seem that computers are already everywhere, the real potential is vastly greater than what we have today. The National Research Council's report *Embedded Everywhere* [4] summarizes this view in the introduction:

> "Information technology (IT) is on the verge of another revolution. Driven by the increasing capabilities and ever declining costs of computing and communications devices, IT is being embedded into a growing range of physical devices linked together through networks and will become ever more pervasive as the component technologies become smaller, faster, and cheaper... These networked systems of embedded computers ... have the potential to change radically the way people interact with their environment by linking together a range of devices and sensors that will allow information to be collected, shared, and processed in unprecedented ways. ... The use of [these embedded computers] throughout society could well dwarf previous milestones in the information revolution."

Sensor networks and "smart dust" [7] are only just breaking out of being laboratory curiosities, but their successes to date imply that the electronics technology scales, and that leveraging advances in sensors, actuators, and wireless networking will make possible (and probable) a pervasiveness of computing that we can only dream of today.

I am convinced, however, that the embedded revolution will require a reexamination, and probably a reinvention of some of the core abstractions of computing and systems engineering. All effective abstractions hide properties of the underlying

[1] It is often assumed that real reason is that embedded software faces more severe resource constraints than general purpose software. But as I have pointed out, resource constraints have repeatedly evaporated with each new generation of computers, and yet the practice of embedded software has changed remarkably little.

systems, but the key to their effectiveness is that they hide the right properties. The divorce of computing from physics has to end for this embedded revolution to take hold.

It is not only the abstractions of computing that have to adapt. Embedded computing will also require a reexamination and reinvention of the core abstractions in the more physics-based engineering disciplines. The models that are used in civil, electrical, and mechanical engineering are deeply rooted in the interactions of physical devices, and poorly express the interactions of those physical devices with computing.

Consider a simple example. A physics-based model of a power distribution system will describe voltages and currents as a function of time, giving their dynamics as ordinary differential equations. The time variable, $t \in \mathbb{R}$, is universal. Its value in Schenectady is the same as its value in San Francisco. But the software embedded in the control system for the power network has tremendous difficulty maintaining a common time base across a distributed system. Even with technologies such as GPS (which provides atomic clock timing precision worldwide), building software that works tightly in concert over geographically distributed systems is extremely difficult. In fact, in the abstractions used to build the software, time is not a part of the ontology. No wonder the engineers who build this software are stuck working with very low-level mechanisms.

Another example is more technically difficult: dealing with random behavior. In standard computing abstractions, we have had the luxury of largely not having to worry about this. This is partly because electronics technology (with some algorithmic help from coding and communication theory) has delivered amazing reliability. Consider the fact that a 40 Gbyte hard disk can be copied flawlessly. This requires that the electronics process 320 billion bits without error. And operations like this occur by the millions on a daily basis. But when we switch our attention to embedded computers with energy scavenging and wireless communication, it is probably too much to expect such reliability. The computing abstractions will have to adapt.

The engineering of systems that are composed of both physical and computational components must be based on abstractions that embrace both physics and computation. There is huge potential for a new "systems science," and there are a few visionaries exploring it. But the cultural divide between computing and engineering is a major barrier to progress. We must break down that barrier.

5.2 Feedback Control, Hybrid Systems, and Beyond

A computational systems theory must, of course, build on both theories of computation and classical systems theories [3]. Ideally, it identifies the common foundations, like theories of composition of components. For example, classical feedback control theory, as illustrated in Fig. 5.1, builds on a key insight, dating back to the 1930s, that feedback systems can be effectively modeled self-referentially, using an abstraction of instantaneous feedback. At its roots, this principle rests on topological fixed point theory, the same foundation underlying recursion theory (a foundation

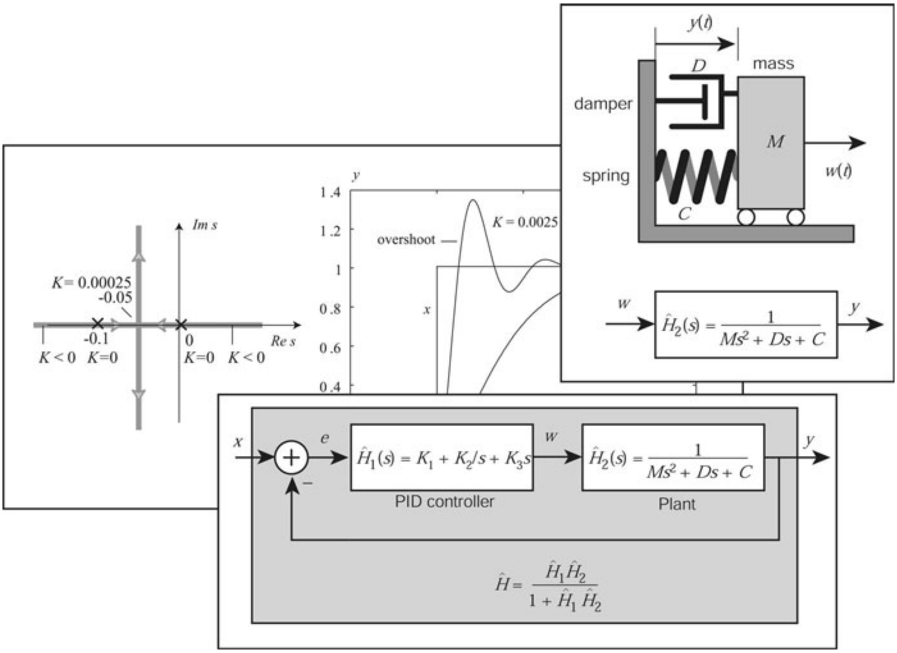

y(t)
D
mass
damper
M
spring
w(t)
C
w
$\hat{H}_2(s) = \frac{1}{Ms^2 + Ds + C}$
y
Im s
K= 0.00025
-0.05
-0.1
0
Re s
K < 0 K=0
K=0 K < 0
y
1.4
1.2
1
0.8
0.6
0.4
overshoot
K = 0.0025
x
x
e
w
y
$\hat{H}_1(s) = K_1 + K_2/s + K_3s$
PID controller
Plant
$\hat{H} = \frac{\hat{H}_1\hat{H}_2}{1 + \hat{H}_1\hat{H}_2}$

Fig. 5.1. Illustration of the mathematical tools of classical feedback control systems (from [14]).

for many modern programming languages) [6] and many theories of concurrency (e.g. the synchronous languages [2] and process networks [8]). It is extremely rare, however, for engineering students (or even engineering faculty) to be even aware of these commonalities. That these commonalities are not exploited in the curriculum is a consequence of the cultural divide that we created in the 20-th century between engineering and computer science.

On the engineering side, we often misrepresent to our students that the connection between the physical world and the world of software is simply a matter of discretizing time. As long as we respect the Nyquist sampling theorem, everything will be fine. Regrettably, software does not perform with the clock-tick regularity of discrete-time abstractions. And even if it did (or if we use tricks to achieve a reasonable approximation), the systems we build in software are far more complex than those we used to build with resistors, capacitors, and inductors. The linear-time-invariant abstraction that underlies so much of the pertinent systems theory is simply not applicable. No wonder engineers using embedded software are stuck with bench testing as their principal analysis tool.

The theory of hybrid systems (see for example [5, 10, 16]) is relatively recent example of a modern systems theory, one that combines computation with classical systems theory. It combines the continuous-time world (or its discretized versions) with the world of irregularly timed mode transitions. It provides analytical tools that

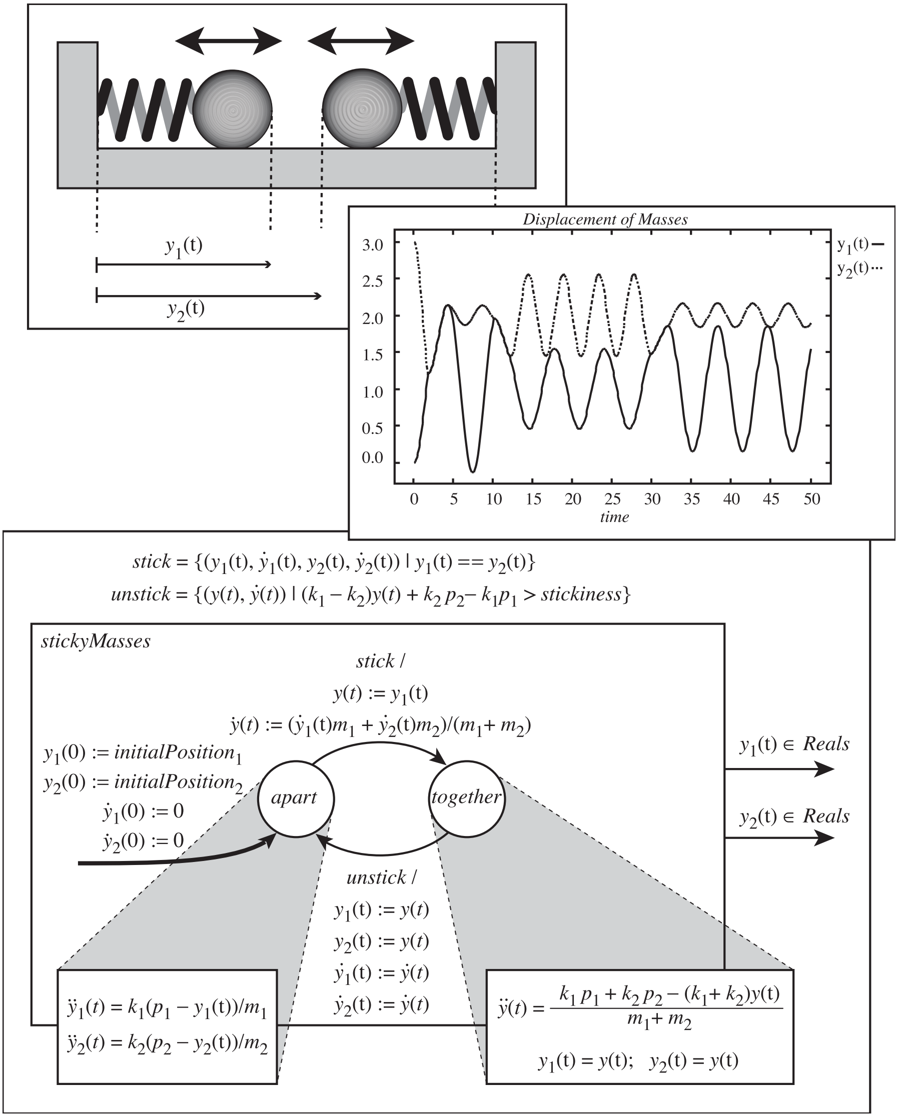
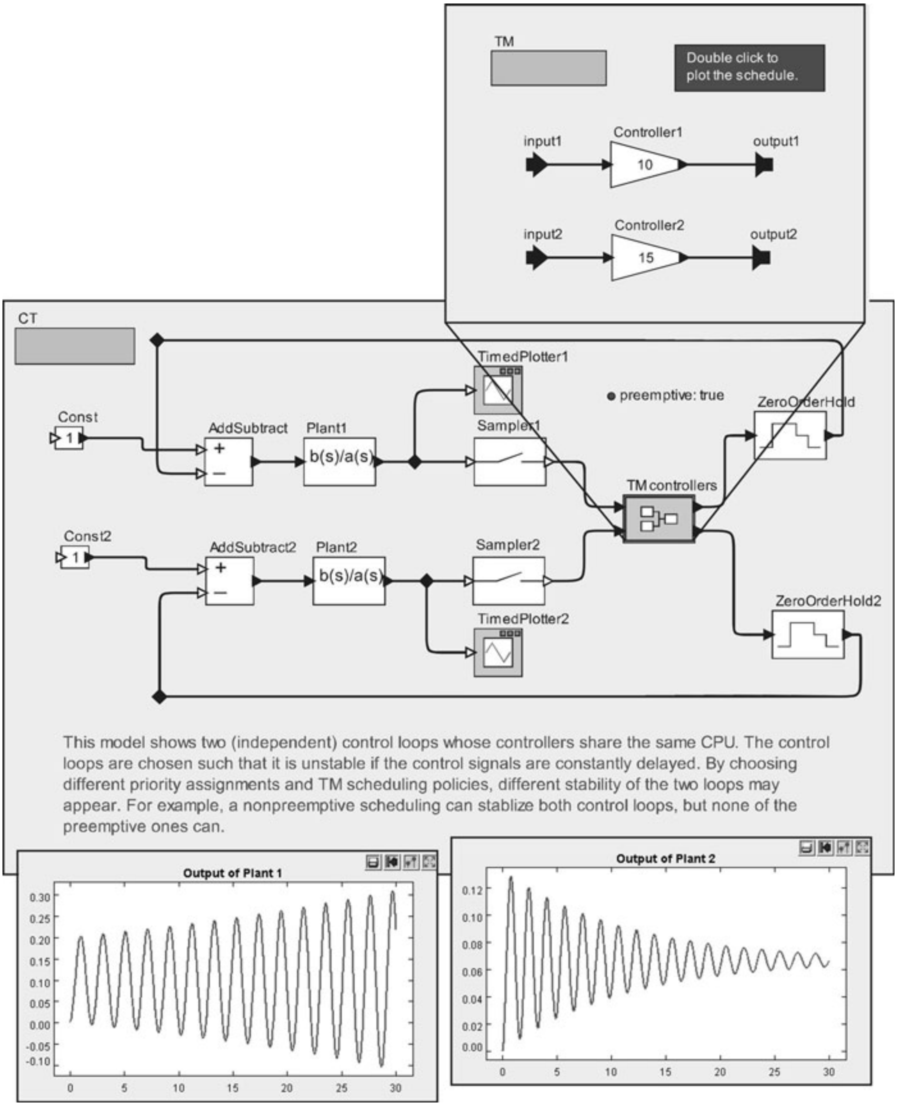

y_1(t)
y_2(t)
Displacement of Masses
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0
5
10
15
20
25
30
35
40
45
50
time
y_1(t) —
y_2(t) ···
$stick = \{(y_1(t), \dot{y}_1(t), y_2(t), \dot{y}_2(t)) \mid y_1(t) == y_2(t)\}$
$unstick = \{(y(t), \dot{y}(t)) \mid (k_1 - k_2)y(t) + k_2 p_2 - k_1 p_1 > stickiness\}$
stickyMasses
stick /
$y(t) := y_1(t)$
$\dot{y}(t) := (\dot{y}_1(t)m_1 + \dot{y}_2(t)m_2)/(m_1 + m_2)$
$y_1(0) := initialPosition_1$
$y_2(0) := initialPosition_2$
$\dot{y}_1(0) := 0$
$\dot{y}_2(0) := 0$
apart
together
unstick /
$y_1(t) := y(t)$
$y_2(t) := y(t)$
$\dot{y}_1(t) := \dot{y}(t)$
$\dot{y}_2(t) := \dot{y}(t)$
$y_1(t) \in Reals$
$y_2(t) \in Reals$
$\ddot{y}_1(t) = k_1(p_1 - y_1(t))/m_1$
$\ddot{y}_2(t) = k_2(p_2 - y_2(t))/m_2$
$\ddot{y}(t) = \frac{k_1 p_1 + k_2 p_2 - (k_1 + k_2)y(t)}{m_1 + m_2}$
$y_1(t) = y(t); \quad y_2(t) = y(t)$

Fig. 5.2. Illustration of the mathematical tools of hybrid systems (from [14]).

are rooted in both linear-time-invariant systems and automata (see Fig. 5.2). It leverages theories of computation to achieve decidability results (or, more commonly, undecidability results) [9], and theories of feedback control to study dynamics and stability. Much of the pioneering work in this area was carried out by teams that included both computer scientists and electrical engineers.



This model shows two (independent) control loops whose controllers share the same CPU. The control loops are chosen such that it is unstable if the control signals are constantly delayed. By choosing different priority assignments and TM scheduling policies, different stability of the two loops may appear. For example, a nonpreemptive scheduling can stablize both control loops, but none of the preemptive ones can.

Fig. 5.3. Model of two software-based controllers executing on a single computer under the control of a real-time operating system, where the controllers are attempting to each stabilize an unstable plant (after [15]).

However, the current intellectual formulation of hybrid systems has its limitations. It still relies on a model of time that poorly fits what software does. Consider a simple example, due to Jie Liu [15], where two software-based controllers execute on a single computer under the control of a real-time operating system (RTOS). A model of such a system is shown in Fig. 5.3. A typical RTOS will offer scheduling policy alternatives, such as preemptive or non-preemptive multitasking, and will support the assignment of priorities to tasks. Under the formulation in Fig. 5.3, if the scheduling policy is preemptive multitasking, only one of the two feedback loops can be made stable (which one depends on the relative priorities). Under non-preemptive multitasking, both can be made stable. This difference is extremely hard to explain using classical control theory. If such a simple system renders our analytical tools useless, then engineers are forced to reject either the implementation technology or the analytical tools. In the former case, an engineer might choose to not share the same computer for the two control loops in order to be able to rely on the analytical results. In the latter case, an engineer will bench test the system to verify stability, tweaking priorities and scheduling policies until the desired behavior is achieved experimentally. Neither outcome is particularly attractive.

When our analytical tools break down even for such small, localized systems, how can we expect them to perform for large-scale distributed systems? The lack of an effective temporal abstraction in software is a major limitation. The tight binding of a universal time continuum with control theory is an equally major limitation. The future of systems theory is going to have to offer better time and concurrency abstractions that yield to both formal analysis and distributed and concurrent software realizations.

5.3 A Focus on Systems

A few years ago, Pravin Varaiya, David Messerschmitt and I led an effort at Berkeley that started down the road of updating the curriculum in the EECS department. We began with our outdated introductory curriculum in EE, where an “introduction to electrical engineering” was principally about passive analog circuits. The rationale for the changes is described in [11], where we cite the considerable work of others that influenced our thinking. A truly long term (and highly speculative) vision is laid out in [12]. The first concrete outcome of this work was a new introductory course on systems [13] and a supporting textbook [14]. Despite this modest progress, the vision remains incomplete and unfulfilled. Academic institutions have considerable inertia.

A unifying theme in these efforts is an increased focus on systems rather than technologies. From [11],

> “First, we must prepare students to select abstractions, not just technologies. Second, just as designs can be built on top of higher level abstractions, so can courses.”

Selecting abstractions requires being able to reason about the properties of those abstractions. All too often, engineering abstractions are presented as immutable facts ("this is how computers work," or "this differential equation describes that feedback circuit") rather than as human ideas ("this is how VonNeumann proposed that we control automatic machines," or "ignoring the intrinsic randomness and latency in this circuit, Black proposed that we could idealize its behavior in this way"). When we present these ideas as immutable facts, we are doing it out of a genuine belief that the methods are useful to engineers. But we are failing to convey that in a rapidly changing technological climate, engineers must be prepared to think critically about the engineering methods, not just about the engineering designs. When we teach modeling, we must also teach meta-modeling, where we discuss the modeling choices.

5.4 Conclusion

Abelson and Sussman describe computer science as "procedural epistemology" [1]. Indeed, 20th century computing was about procedure as knowledge. I believe that 21st century computing will transform into a system science that subsumes procedure, but also embraces concurrency, time, randomness, and physicality. 21st century computing will be an epistemology of concurrent interacting components. And the highly valued engineering education will be that which focuses on systems rather than on technologies.

References

[1] H. Abelson and G.J. Sussman, *Structure and Interpretation of Computer Programs*, MIT Press, second edition, 1996.

[2] A. Benveniste and G. Berry, The synchronous approach to reactive and real-time systems, *Proceedings of the IEEE*, **79** (9):1270–1282, 1991.

[3] R. Boute, Integrating formal methods by unifying abstractions, In E. Boiten, J. Derrick and G. Smith, Eds., *Fourth International Conference on Integrated Formal Methods*, LNCS **2999**:441–460, Springer-Verlag, Canterbury, Kent, England, 2004.

[4] C.S. Committee on Networked Systems of Embedded Computers, D. o. E. Telecommunications Board, and N.R.C. Physical Sciences, *Embedded, Everywhere—A Research Agenda for Networked Systems of Embedded Computers*, National Academy Press, Washington DC, 2001.

[5] A. Deshpande and P. Varaiya, Information structures for control and verification of hybrid systems, In *Proc. American Control Conference*, Seattle, 2642–2647, 1995.

[6] J. Hartley Rogers, *Theory of Recursive Functions and Effective Computability*, McGraw-Hill, 1967.

[7] T. Hoffman, Smart dust - mighty motes for medicine, manufacturing, the military and more, *Computerworld*, March 24 2003.

[8] G. Kahn, The semantics of a simple language for parallel programming, In *Proc. of the IFIP Congress 74*, North-Holland Publishing Co., 1974.

[9] P. Kopke, T. Henzinger, A. Puri, and P. Varaiya, What's decidable about hybrid automata? In *27th Annual ACM Symposioum on Theory of Computing (STOCS)*, 372–382, 1995.

[10] M. Kourjanski and P. Varaiya, Stability of hybrid systems, Hybrid Systems, LNCS **1066**:413–423, Springer-Verlag, 1995.

[11] E.A. Lee and D.G. Messerschmitt, Engineering an education for the future, *IEEE Computer Magazine*, **31** (1), 1998.

[12] E.A. Lee and D.G. Messerschmitt, A highest education in the year 2049, *Proceedings of the IEEE*, **87** (9), 1999.

[13] E.A. Lee and P. Varaiya, Introducing signals and systems—the Berkeley approach, In *First Signal Processing Education Workshop*, Hunt, Texas, 2000.

[14] E.A. Lee and P. Varaiya, *Structure and Interpretation of Signals and Systems*, Addison Wesley, 2003.

[15] J. Liu, Responsible Frameworks for Heterogeneous Modeling and Design of Embedded Systems, Ph.D. Thesis, Technical Memorandum UCB/ERL M01/41, December 2001.

[16] A. Puri and P. Varaiya, Verification of hybrid systems using abstractions, In *Hybrid Systems II*, Proc. Hybrid Systems Workshop, LNCS **999**:359–369, Springer-Verlag, 1994.

6

New Directions in System Design Automation

Akash R. Deshpande

Founder and Chief Technology Officer
Teja Technologies, Inc.
akash@teja.com

Summary. I take this opportunity with great pleasure to thank Prof. Pravin Varaiya for his guidance over the past fifteen years not only in my academic research but also at Teja Technologies, Inc. In this chapter I have outlined some of the emerging themes in system design particularly for network equipment. Factors such as the proprietary nature of many of the developments, the rapid pace of change in the field, and also the desire to keep out material that may appear promotional of commercial interests have required this chapter to be kept at a fairly general level.

My doctoral thesis work with Prof. Varaiya dealt with the modeling, analysis and control of hybrid systems—i.e., systems which combined continuous and discrete state dynamics [1]. Subsequently, as a member of a research team at California PATH Laboratory of the UC-Berkeley, directed by Prof. Varaiya, I contributed to the development of a hybrid system specification and simulation system called SHIFT [2]. After forming Teja Technologies, Inc., where Prof. Varaiya was on the Board of Directors for several years, I continued the core work on a commercial basis with emphasis on high-performance execution applied to fast path network applications.

6.1 Architectures

Improvements in silicon process technology (130nm to 90nm to 65nm) and progressively challenging price-performance demands are driving the development of new techniques in system design and implementation. The density of gates in silicon now easily enables the integration of multiple functions on a single "system-on-a-chip," or of multiple processor cores in a variety of interconnects, or a combination of software programmable and hardwired logic elements in application-specific, configurable architectures. But the availability of such an enabling capability does not by itself indicate how systems should be designed. On the other hand, system requirements in terms of throughput and latency performance, power consumption, device area and cost continue to get more stringent. Because of the wide range of system functions, and their diverse definitions of performance, there is not yet a generalized guiding design principle for attaining the price-performance metrics across various domains. Over the last few years, several industry projects have been conducted in packet processing and signal processing, and new projects are emerging in server processing, that now start to show a pattern that can be used for general system design automation.

In this chapter, we shall consider systems for which throughput performance and not latency performance is the primary concern. In case latency is the primary concern, then the system developer must determine the parallel paths in the application's algorithms and exploit them to the fullest in the implementation. Since these are likely to be different from one application to the next, it may be hard to find general techniques for designing such systems. As one example of such a general principle, latency-minimizing systems are typically implemented to take in one data "sample" and process it fully before starting on the next sample. Exploitation of parallelism is greatly aided by the high degree of silicon integration since more circuits (= more parallel paths) can be implemented, thereby increasing the system performance (and also device area). It is our belief that existing approaches in electronic design automation (EDA) are largely adequate in this problem domain.

When latency performance requirement is relaxed, a different approach is opened for system design. In this approach, multiple "samples" are processed simultaneously, thereby gaining systemic parallelism independent of the specific application algorithms that are run on the samples. Typically, most packet processing applications and a large class of signal processing applications are, within limits, relatively insensitive to latency performance, and hence benefit from such a treatment.

Fig. 6.1 uses a simple example to illustrate the throughput performance benefits of processing multiple packets simultaneously. Fig. 6.1(a) shows a single packet processing block, Fig. 6.1(b) shows two blocks operating in parallel and Fig. 6.1(c) shows two blocks operating in a pipeline. In all cases, P packets arrive into the system per second. Each packet requires I instructions of processing, considering each block to be a processor. (Equivalently, any block could be implemented as hardwired logic, with each packet requiring a certain number of logic operations—without loss of generality, we will consider the processor case).

In case 1.a, the single processor must execute $P * I$ instructions per second. In case 1.b, each processor receives every other packet for processing. Hence (ignoring second-order effects) it needs to run at only half the clock rate of the processor in case 1.a. In case 1.c, the first processor executes the first half of the instructions in the program and the second processor executes the second half. Since each processor executes only half the instructions per packet, again it needs to run at half the clock rate of the processor in case 1.a.

In addition to reducing the processor clock rate, such parallel or pipelined architectures provide better memory and I/O latency hiding since stalls in some processors can overlap with computation in the other processors. Such latency hiding is enhanced when each processor can support multiple hardware contexts.

While scenarios described above are intentionally simplistic, typical architectures have several (tens or hundreds) of processor or logic blocks, additional logic for arbitration of access to shared resources, internal communication paths between blocks, caching and other local data storage schemes, coherency maintenance mechanisms, and specialized hardware logic elements. Despite the complications intro-

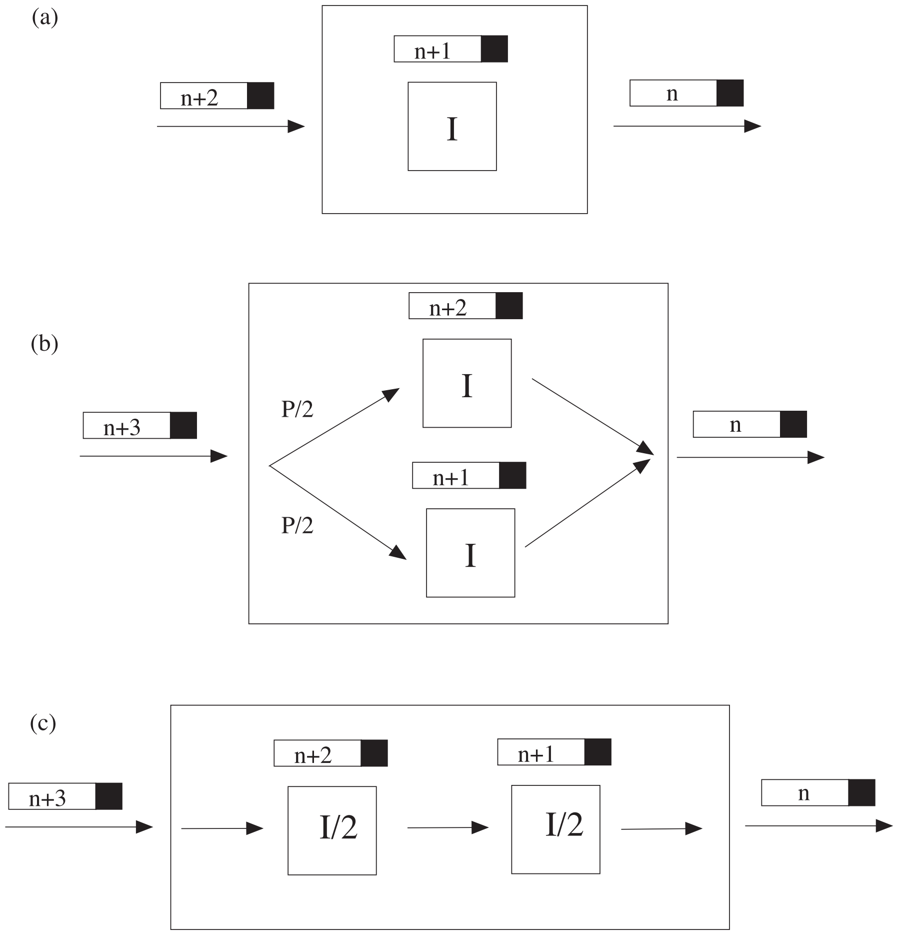


Fig. 6.1. Packet processing blocks. (a) Single packet processing block, (b) Parallel packet processing blocks, (c) Pipelined packet processing blocks

duced by these factors in both device and system design, the benefits in price and performance far outweigh the drawbacks, and the trend is towards increasing proliferation of such architectures.

The two architectures used in the example—parallel and pipelined—appear to hold the most promise going forward. The parallel architecture seems suited for "end-point" systems—i.e., systems that are the ultimate producers or consumers of packetized data. The pipeline architecture seems suited for "mid-point" systems—i.e., systems that are the forwarders of packetized data. Accordingly, symmetric, multi-threaded, multi-core processors are being employed in computer server systems, and pipeline architectures are being employed in switches, routers and other network equipment. We note that while it is unlikely that pipeline architectures will form the basis of computer server systems, there is no architectural barrier for parallel architectures to be used in switches and routers. Since servers are intended for general purpose processing, the parallel architectures will continue to exhibit features that are beneficial across multiple applications. It remains to be seen whether standard, off-the-shelf, general purpose pipeline architecture components shall survive in the long run. On the other hand, since a given embedded system has a fairly specific application, there will be continuing need to develop application-specific pipeline architectures.

At first sight it appears obvious that the parallel architecture must be used symmetrically. Typically a symmetric multiprocessing operating system is run on the parallel architecture, and then the various applications are run on the operating system. In some server systems, multiple virtual machines are run on partitions of the parallel architecture to provide a reliable environment for running multiple applications. But it has been shown convincingly that such symmetric uses of the architecture do not yield the best overall system performance. Performance is enhanced by reserving a portion of the computing resources for offloading the common functions of all applications such as, for example, layers 2–4 of packet processing and stored data access. The portion of the resources devoted to the offload function needs to be easily configurable in order to balance the overall system operation. In effect, the parallel architecture is converted into a two-stage pipeline between the offload processing and the application processing stages. Overall system performance gain by this technique by a factor of 3–5 has been demonstrated [3].

Coming now to pipeline architectures, we first notice that a general purpose processor is typically required beside the “fast path” pipeline in order to manage the control information used by the pipeline. While the pipeline efficiently executes application-specific functions, the control processor(s) run an operating system and general applications on top.

Thus, at some level, there is a unification of the two architectures, with some computing resources devoted to an operating system running general purpose applications and others reserved for offloading the fast path functions.

6.2 System Design Automation

Because of the application-independent nature of the systemic characteristics of throughput performance systems, and because of the unified nature of the target architectures, it becomes possible to develop general approaches for system design automation. The design flow of such an automation scheme has the following elements:

- specification of the system architecture,
- specification of the application algorithms and data structures,
- specification of the mapping of the application to the architecture,
- generation of the system based on these specifications,
- inspection of performance results of the generated system, and

- iteration over these steps until the desired price-performance tradeoffs are met.

Without loss of generality, additional layers can be implemented on top of this baseline design flow to support higher productivity features such as object-orientation and automated iteration using some sort of constrained optimization scheme.

For pragmatic reasons such as training, reuse and legacy code base, it becomes necessary to choose the ANSI C programming language (and its associated programming model) for the various specifications, especially of the algorithms and data structures. It is important to keep any new concepts to a minimum, and to provide mechanisms that enable these concepts to be applied to a wide range of system functions and implementations. We now describe some of the key concepts in these specifications.

For architecture specification, there are the following concepts: architecture, processor, memory, logic block, bus, OS and process. An architecture is a container for these elements (including other architectures). Buses are used to connect elements. An OS runs on processor(s) and a process runs on an OS. Process, (free) processors and logic blocks are targets for mapping algorithms, and memories are targets for storing static data as well as managing dynamic data. Each of these elements can have a wide range of types and detailed properties based on specific implementations.

The following program fragment illustrates the definition of the Intel IXP2800 network processor. The purpose of this fragment and others that follow is to give a flavor of the specifications using concrete examples. Their detailed explanations are beyond the scope of this chapter. All data types and functions beginning with the `teja_` prefix illustrate the architecture specification concepts.

```
teja_architecture_t
create_ixp2800_architecture(teja_architecture_t container, const char *name)
{
  teja_architecture_t ixp2800;
  teja_processor_t xscale, ue[16];
  teja_memory_t scratchpad, rbuf, tbuf, shared[16];
  teja_memory_t local[16], local_shadow[16];
  teja_bus_t msf_bus, pci_bus, sram_bus, dram_bus;
  teja_bus_t ue_bus[16], slowport_bus;

  int i;
  char buf[256];
  char buf1[32];

  ixp2800 = teja_architecture_new(container, name, IXP2800);

  xscale = teja_processor_new(ixp2800, "xscale", XSCALE);

  for(i=0; i<16; i++) {
    sprintf(buf, "ue%i", i);
    sprintf(buf1, "%i", i);
    ue[i] = teja_processor_new(ixp2800, buf, IXPMEV2);
    teja_processor_set_property(ue[i], IXPMEV2_UENG_ID, buf1);
  }

  scratchpad = teja_memory_new(ixp2800, "scratchpad", IXPSCRATCHPAD);

  rbuf = teja_memory_new(ixp2800, "rbuf", IXPRBUF);

  tbuf = teja_memory_new(ixp2800, "tbuf", IXPTBUF);
  for(i=0; i<16; i++) {
    sprintf(buf, "shared%i", i);
    shared[i] = teja_memory_new(ixp2800, buf, IXPSHAREDREG);

    sprintf(buf, "localmem%i", i);
    local[i] = teja_memory_new(ixp2800, buf, IXPLOCALMEM);
    ...
  }

  ...

  pci_bus = teja_bus_new(ixp2800, "pci_bus", PCI, EXPORTED_BUS);

  sram_bus = teja_bus_new(ixp2800, "sram_bus", SRAM, EXPORTED_BUS);

  dram_bus = teja_bus_new(ixp2800, "dram_bus", DRAM, EXPORTED_BUS);


  teja_processor_connect_default(xscale, sram_bus);
  ...
  teja_memory_connect_default(scratchpad, sram_bus);
  ...

  for(i=0; i<16; i++) {
    ...
    teja_memory_connect_default(local_shadow[i], ue_bus[i]);
  }

  return ixp2800;
}
```

The use of this architecture in the Intel IXDP2801 board design is shown in the following program fragment.

```
teja_architecture_t
create_ixdp2801_architecture(teja_architecture_t container,
                             const char *name)
{
  teja_architecture_t ixp2800, ixdp2801;
  teja_bus_t pci_bus, sram_bus, dram_bus;

  ixdp2801 = teja_architecture_new(container, name, IXDP2801);

  ixp2800 = create_ixp2800_architecture(ixdp2801, "ixp2800");

  teja_architecture_set_property(ixp2800, IXP2800_DRAM_CLOCK_FREQ, "100.0");

  ...

  teja_architecture_connect(ixp2800, "pci_bus", pci_bus);
  teja_architecture_connect(ixp2800, "sram_bus", sram_bus);
  teja_architecture_connect(ixp2800, "dram_bus", dram_bus);

  ...

}
```

The use of this board with the Linux operating system running an application process is illustrated by the following program fragment.

```
void  packet_classifier_config() {
  teja_architecture_t pca, ixdp2801, ixp;
  teja_processor_t ue;
  teja_os_t linux;
  teja_process_t init;
  char buf[20];
  int i;

  pca = teja_architecture_new(NULL, "pca", USER_DEFINED);

  ixdp2801 = create_ixdp2801_architecture(pca, "ixdp2801");

  for (i=0; i<16; i++) {
    sprintf(buf, "ixdp2801.ixp2800.ue%i", i);
    ue = teja_processor_lookup(pca, buf);
    teja_processor_set_property(ue, IXPMEV2_IN_USE_CTX, "4");
  }

  linux = teja_os_new(pca, "linux", MVLINUX);
  init = teja_process_new(pca, "init");
}
```

Algorithms and data structures are specified using the ANSI C language supplied with two special features (expressed within the C syntax). The first is an API for making explicit the coordination of parallel programs. This API, known as the late-binding API, provides the abstractions of mutual exclusion, synchronized queue, asynchronous communication channel, dynamic memory pools and the waiting for events. The use of this API in the application simply indicates the coordination points in the distributed logic. The API is bound to an implementation later in the design flow in the mapping specification. The second feature is the specification of time, event and logic-driven state machine structures within a sequential program by providing a pragma that marks statement labels as states, with the corresponding interpretation that control flows to that labeled statement constitute state transitions of the automaton.

The following program fragment illustrates a packet classifying algorithm specified using the late binding API. A fragment of the alternative specification using state machine notation is also provided. All the data types and functions beginning with the teja_ prefix illustrate the late binding API concepts.

```
extern teja_channel_t channel;
extern teja_memorypool_t packet_pool;

extern struct max_header header_cache;
extern struct statistics stats;
extern struct exceptions exc;

void classifier()
{
  int total_pkts = 0;
  int len = 0;
  int *packet = NULL;

  int evt;
  struct packet_descriptor *pd;

  int opt_len;

  while (1) {

    // start to rx_status
    teja_wait(-1, -1, &evt, (void*) &pd, channel);

    if (pd != NULL) {
      packet = pd->pkt_ptr;
      len = pd->size;

      // rx_status to rx_success
      if (packet != NULL) {
        total_pkts += 1;
        teja_memcpy(&header_cache, packet, sizeof(struct max_header));

        // rx_success to ip
        if (header_cache.protocol_type == ASSIGNED_IP_NO) {
          opt_len = (header_cache.ver_hdr_len & 0x0f) - 5;

          // ip to tcp
          if (header_cache.protocol == ASSIGNED_TCP_NO) {
            statistics_received(&stats, TCP_TYPE);
            // tcp to start
            teja_memorypool_put_node(packet_pool, packet);
          }

          // ip to icmp
          else if (header_cache.protocol == ASSIGNED_ICMP_NO) {
            statistics_received(&stats, ICMP_TYPE);
            // icmp to start
            teja_memorypool_put_node(packet_pool, packet);
          }

          // ip to udp
          else if (header_cache.protocol == ASSIGNED_UDP_NO) {
            statistics_received(&stats, UDP_TYPE);
            // udp to start
            teja_memorypool_put_node(packet_pool, packet);
          }
        }

        // rx_success to arp
        else if (header_cache.protocol_type == ASSIGNED_ARP_NO) {
          statistics_received(&stats, ARP_TYPE);
          // arp to start
          teja_memorypool_put_node(packet_pool, packet);
        }

        // rx_success to start
        else {
          exceptions_inc_drop_count(&exc);
          teja_memorypool_put_node(packet_pool, packet);
        }
      } // rx_status to start
    } // start to start - implicit in original
  }
}

void alternative_classifier() {
 _Pragma("state") start:
  teja_wait(-1, -1, &evt, (void*) &pd, channel);
  if (pd != NULL) {
    packet = pd->pkt_ptr;
    len = pd->size;
    goto rx_status;
  }

 _Pragma("state") rx_status:
  if (packet != NULL) {
    total_pkts += 1;
    teja_memcpy(&header_cache, packet, sizeof(struct max_header));
    // cut through to next pipeline stage
    teja_send(channel, PACKET_EVENT, pd,
      sizeof(struct packet_descriptor), packet_descriptor_pool);
    goto rx_success;
  }
 ...
}
```

The mapping specification combines the application and the architecture into a system: functions map to processes, processors or logic blocks, data structures map to memories, and the late-binding API maps to the detailed properties of the target architecture.

The following program fragment illustrates the mapping specification.

```
void packet_classifier_map()
{
  teja_os_map("lin", "pca.ixdp2801.ixp2800.xscale");

  teja_process_map_os("init", "lin");

  teja_run_on_process("setup", "init", INIT);

  teja_run_on_processor("generator",
                        "pca.ixdp2801.ixp2800.ue0",
                        GENERATOR);

  teja_producer_connect(GENERATOR, "channel");
  teja_consumer_connect(CLASSIFIER, "channel");

  teja_memorypool_map_sram_list("packet_descriptor_pool",
    1, 100, sizeof(struct packet_descriptor), 64);
  ...
  teja_channel_map_next_neighbor_ring("channel",
                                      "signal_based",
                                      0,
                                      0,
                                      "qdrsram");

  teja_variable_map_memory("arp_packet", "pca.ixdp2801.sram2");
  ...
  teja_variable_map_memory("stats", "pca.ixdp2801.ixp2800.scratchpad");
  ...
  teja_variable_map_memory("header_cache",
                           "pca.ixdp2801.ixp2800.localmem1");

  teja_init_function("init_packet_classifier",
                     "pca.ixdp2801.ixp2800.xscale");
}
```

The system generation step (implemented, for example, as the Teja C compiler `tejacc`) employs a novel approach compared to traditional compiler and operating system tools or to synthesis tools. First `tejacc` compiles the architecture specification source files as a standard C program and executes it. The effect of this execution is to customize (the running copy of) `tejacc` for the target architecture. Then this customized `tejacc` parses and analyzes the application source files in the context of the target architecture. This provides the compiler with a whole system view for validation and optimization, including the analysis and unrolling of the state machines and the inlining of functions and other constant optimizations. Finally, `tejacc` analyzes the mapping specification in preparation for system generation. To aid this step, two other platform components are used. One is a library of architectural infrastructure elements as implementations of the late-binding API provided for processor as well as logic-based parallel and pipelined architectures, and the other is a set of target-specific compiler back-ends that support the generation of processor-based microcode or logic-based HDL code. Thus, the system generation supports, on one hand, a contained set of concepts for architecture and algorithm specification and targets a rich set of implementations, on the other hand, through target-specific support packages, while attaining high performance in all cases using global system optimization techniques.

The performance feedback on the generated system is keyed to the conceptual framework treated as a tightly-coupled distributed processing network. For each thread of execution the cycle counts through various called functions and state transitions as well as the busy/idle duty cycles are reported. For each coordination element such as a queue, mutex or channel, the loading in terms of number of stored messages or requests and their arrival and departure statistics is reported. A time series of these reports can be analyzed using network techniques to discern bottlenecks and derive insight into a remapping of the application or a rearchitecting of the system.

6.3 Results and Future Directions

We have implemented the system design automation scheme described in this chapter and applied it to multiple target applications and architectures with excellent performance and productivity results. The architectures targeted span parallel and pipeline structures of both fixed and configurable types such as parallel multi-threaded, multi-core server processors, pipelined multi-threaded, multi-core network processors and configurable FPGAs with combined parallel and pipelined architectures of both software programmable and hardwired elements. The applications span layers 2–4 of the OSI network model with some emerging applications all the way up to layer 7, with performance levels ranging from 1–10 gigabits per second for minimum size Ethernet frames at attractive resource efficiencies.

In addition to the themes presented in this chapter, research directions include increased automation of software-hardware partitioning and software partitioning across multiple hardware resources, improvement in hardware logic generation, and the integration with test and formal verification platforms.

References

[1] A. Deshpande, Control of Hybrid Systems, Ph.D. Dissertation, University of California, Berkeley, 1994.

[2] A. Deshpande, A. Gollu and L. Semenzato, The shift programming language for dynamic networks of hybrid systems, *IEEE Trans. Automatic Control*, special issue on Hybrid Systems, April, 1998.

[3] Teja Technologies Inc., Software Performance Considerations for Multi-core Processors, 2004. http://www.teja.com/content/teja-multiprocessor-perf.pdf.

Part III

Networks

7

Causal Coding and Feedback in Gaussian Sensor Networks

Michael Gastpar

University of California, Berkeley
gastpar@berkeley.edu

Summary. Varaiya and Walrand found an elegant insight regarding the use of feedback in a causal coding context: While generally useful, feedback becomes useless when the channel is sufficiently symmetric. The goal of this note is to extend this insight to scenarios inspired by sensor networks. Specifically, two such scenarios are considered: a situation with a single sensor but where the source is observed through a noisy channel, and a genuine network scenario where all source and noise distributions are assumed to be Gaussian. For the latter, it is shown that feedback is useless if source and channel bandwidth are equal, but that, if the latter is larger, feedback is strictly useful. Varaiya and Walrand establish their results via dynamic programming arguments. It is unclear to date whether such arguments can be extended to the distributed scenario considered in the present chapter. Instead, our results are established via information-theoretic bounds.

7.1 Introduction

Causal coding and decoding is a desirable feature in a number of sensor network applications. Consider for example an earthquake monitoring sensor network in the framework of an emergency management system. Clearly, the process of communicating the sensed data to the central collection unit (i.e., their encoding and decoding) should introduce as little delay as possible, calling essentially for causal coding. Another example is the case of sensor networks with very simple (i.e., cheap) sensor nodes that may have little or no space for storage. Causal coding permits us to efficiently take into account such a limitation.

What is the use of feedback in a causal coding/decoding context? Varaiya and Walrand [20, 21] have developed a perspective and the necessary tools for analyzing causal encoding/decoding in a point-to-point communication system. One of their key results is that feedback is strictly useful in general causal coding problems, but that it becomes useless if the channel is sufficiently symmetric. This is by contrast to the case of arbitrary (not necessarily causal) encoding and decoding, unconstrained both in delay and in complexity, where it is well known that for discrete memoryless communication channels, feedback does not permit an increase in capacity [4,

p. 212]. (It can, however, improve the reliability, i.e., enhance the decay of the error probability as a function of the duration of the transmission [3].)

In this chapter, we extend the results of [20, 21] in two directions, both inspired by sensor network problems. First, we consider the scenario where the source is not observed directly by the encoder, but rather through a noisy channel, modeling the noise introduced by the sensing device. As we show, the same dynamic programming arguments as in [20, 21] can be used to establish the usefulness of feedback.

Then, we study the case of multiple sensors that have to process their respective observed data in a distributed fashion before communicating it to a central data collection point. It is unclear to date whether dynamic programming arguments can be extended to the distributed scenario considered in this chapter. Instead, we restrict attention to one particular Gaussian sensor network model and use a different set of tools. In particular, using information-theoretic arguments, it is again possible to make claims about the usefulness of feedback.

There are two main contributions for the considered Gaussian sensor network scenario. First, it is shown that when source and channel bandwidth are equal, feedback is useless in a causal coding context. Second, it is shown that when the channel bandwidth exceeds the source bandwidth, feedback is strictly useful.

These results, in some sense, confirm the basic intuition furnished by Varaiya and Walrand in [20, 21]: when the communication channel has an appropriate symmetry, feedback is useless. Specifically, in our example, when source and channel bandwidth are equal, the resulting overall situation can be seen as "sufficiently symmetric" for feedback to become useless. Conversely, in the absence of symmetry, [20, 21] provides examples where feedback is strictly useful in a causal coding context. Again, this is reflected by our findings in the sense that when source and channel bandwidths differ, feedback enhances performance.

The rest of the chapter is organized as follows. In Section 7.2, the remote source-channel communication problem is considered. After setting up the problem and discussing known information-theoretic bounds, we investigate the problem of *causal* encoding and decoding. The optimum schemes can be characterized using the methods and arguments of [20, 21]. Related work has been presented in [1].

In Section 7.3, we provide detailed definitions and assumptions for the specific Gaussian sensor network investigated in the present chapter.

While the arguments of [20, 21] can be extended easily to the *remote* source-channel coding problem (as we show in Section 7.2.4), there does not seem to be an extension to the distributed processing problem (as defined in Section 7.3). Using a different set of tools, in Sections 7.4 and 7.5, we present the two main results of the chapter, illustrating that feedback is useless in one case and useful in the other.

7.2 The Remote Source-Channel Communication Problem

7.2.1 Notations and conventions

In this chapter, we will denote random variables by upper case letters, such as X, and their realizations by lower case letters, such as x. The alphabet (or range) of x is

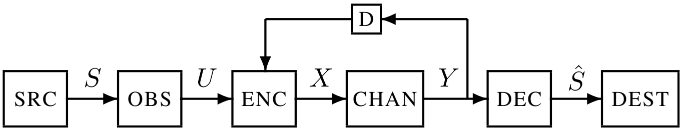


Fig. 7.1. The remote source-channel communication scenario: A source (SRC) is observed through a noisy process (OBS) by an encoder (ENC) whose task is to produce a signal to be transmitted across the channel (CHAN) to a decoder (DEC). The decoder must provide the destination (DEST) with the best possible estimate of the original source. Moreover, the encoder may also know the channel output signal, at a delay (D) of one time unit.

denoted by $\mathcal{X}$, and the probability mass function (or probability density function if $\mathcal{X}$ is continuous-valued) will be denoted by

$$p_X(x). \tag{7.1}$$

The expectation operator will be denoted by $E[\cdot]$.

Sequences of random variables will be denoted by $\{X[i]\}_{i=1}^n$, where i is thought of as the (discrete) time index. Occasionally, we will use the shorthand $X^n = \{X[i]\}_{i=1}^n$.

7.2.2 Model definition and problem statement

In this section, we study the communication system illustrated in Fig. 7.1. It is almost the standard source-channel communication system, except that the encoder does not get to observe the source directly. Rather, it merely gets a noisy version of the source outputs. This could be termed *remote* source observation, and we therefore refer to the resulting communication problem as the remote source-channel communication scenario. Our consideration is motivated by the fact that the sensor reading is rarely the quantity that one is interested in. Rather, the sensor network is typically expected to reveal an underlying state of nature, which often can be observed only partially and subject to measurement noise in the sensing device.

More precisely, for the scope of this chapter, we consider a (discrete-time) stationary ergodic source characterized by a sequence of random variables $\{S[i]\}_{i\geq 1}$, where i denotes the (discrete) time index. The probability mass (or density) function of the sequence of random variables is assumed to be fixed and known. The source outputs are observed through a *noisy* observation process by the encoder, characterized in a probabilistic manner[1] by a conditional distribution $p(u|s)$. We consider block codes of length n. That is, upon observing a sequence $\{U[i]\}_{i=1}^n = \{u[i]\}_{i=1}^n$, the encoder must produce a codeword, i.e., a sequence of n channel input symbols,

$$\{x[i]\}_{i=1}^n = f_n(\{u[i]\}_{i=1}^n). \tag{7.2}$$

[1] For simplicity, our considerations are restricted to the case where the noisy channel through which the source is observed is *memoryless*.

More generally, as illustrated in Fig. 7.1, we also consider the case of (perfect) feedback. That is, beyond (7.2), the channel input signal may also depend on past channel outputs, as follows:

$$x[i] = f_{n,i}(\{u[j]\}_{j=1}^{n}, \{y[k]\}_{k=1}^{i-1}). \tag{7.3}$$

In our communication model, there is a price or cost associated with each pair of channel input and output symbols, measured by a *channel cost*[2] function,

$$\rho : \mathcal{X} \times \mathcal{Y} \rightarrow R_{+}, \tag{7.4}$$

where we use R_{+} to denote the non-negative real numbers. In terms of this cost, the coding function of Equation (7.2) must be selected in such a way as to satisfy a constraint of the form

$$\frac{1}{n}\sum_{i=1}^{n} E\left[\rho(X[i], Y[i])\right] \leq P_n. \tag{7.5}$$

A usual choice of a channel cost function is $\rho(x) = |x|^2$, permitting us to associate the expected cost with the power consumption of the communication system.

The channel is assumed to be memoryless and characterized by a conditional distribution $p_{Y|X}(y|x)$, that is,

$$p(y[n]|x[n], y^{n-1}) = p_{Y|X}(y[n]|x[n]). \tag{7.6}$$

Based on the observed channel output sequence y^n, the decoder produces an estimate $\{\hat{S}[i]\}_{i=1}^{n}$ of the underlying source sequence, according to a mapping

$$\{\hat{s}[i]\}_{i=1}^{n} = g_n(\{y[i]\}_{i=1}^{n}). \tag{7.7}$$

The overall coding/decoding effort is then assessed by the *distortion* between the actual source output sequence, and its corresponding reconstruction at the decoder. To this end, the model specifies a *distortion measure*

$$d : \mathcal{S} \times \hat{\mathcal{S}} \rightarrow R_{+}. \tag{7.8}$$

The resulting distortion is assessed in terms of its average,

$$D_n = \frac{1}{n}\sum_{i=1}^{n} E\left[d(S[i], \hat{S}[i]\right], \tag{7.9}$$

where the expectation is evaluated over the joint distribution of the pair of sequences $(\{S[i]\}_{i=1}^{n}, \{\hat{S}[i]\}_{i=1}^{n})$.

[2] This terminology, while customary in the communication theory literature, see e.g. [6], is unfortunately in slight contradiction with [20, 21], where the term “cost” denotes what we will call “distortion” below. Therefore, we will be explicit and call ρ the “channel cost function” throughout.

The success of a particular block coding scheme (of block length n) is measured entirely in terms of the cost-distortion pair (P_n, D_n). Optimal pairs are then optimal *trade-offs* between cost and distortion. More specifically, a pair (P_n, D_n) represents an optimal system if at fixed cost P_n, there is no other code with distortion $\tilde{D}_n < D_n$, and at fixed distortion D_n, there is no other code with cost $\tilde{P}_n < P_n$. In this chapter, we are interested in the optimum performance *irrespective of the complexity of the encoding and decoding*, in the limit as $n \to \infty$. We consider in parallel two settings:

(a) The case where the encoding and decoding mappings are entirely unconstrained.
(b) The case where only causal encoding and decoding is allowed (as defined in [20, 21]).

7.2.3 Information-theoretic bounds

In this subsection, we suppose that *arbitrary* encoding and decoding is allowed, in the sense that the encoder in Eq. (7.2) and the decoder in Eq. (7.7) may be arbitrary mappings. The information-theoretic approach to this problem, developed by Shannon [19], is to determine the ultimate limits on (P_n, D_n), as $n \to \infty$. This is sometimes referred to as the OPTA (optimum performance theoretically attainable), see [2, p.156].

For the remote source-channel communication scenario, the problem formulation and the corresponding solution have been found in [7]. An extended account can be found in [2, p.78, p.124ff.]. For the purposes of this chapter, we need the following result:

Theorem 1. *The optimum trade-off between cost and distortion (OPTA) for the remote source-channel communication scenario with a memoryless channel is characterized by*

$$R_{remote}(D) = C(P), \tag{7.10}$$

where $R_{remote}(D)$ denote the remote rate-distortion function for the source, and $C(P)$ the Shannon capacity of the communication channel. This is true irrespective of whether feedback from the channel output to the encoder is available.

The proof of this theorem is briefly outlined in Appendix A.

7.2.4 Causal coding for a first-order Markov source

In this section, we derive the analogues of the results of [21] for the communication scenario illustrated in Fig. 7.1 and defined in Section 7.2.2. That is, we now restrict the coding scheme to be causal: The encoder map in Eq. (7.3) must satisfy

$$x[i] = f_{n,i}(\{u[j]\}_{j=1}^{i}, \{y[j]\}_{j=1}^{i-1}), \tag{7.11}$$

i.e., the ith component of the channel input may only be a function of $\{s[j]\}_{i=1}^{i}$. Similarly, the decoder map in Eq. (7.7) must satisfy

$$\hat{s}[i] = g_{n,i}(\{y[j]\}_{j=1}^{i}). \tag{7.12}$$

This setup can be illustrated using a simple extension of the instructive example in [21, Sec. III.B], permitting us to conclude that feedback is *strictly* useful in a causal coding context, in some cases.

As in [21, Sec. III.B], this example permits us to conclude that in a causal encoding-decoding context, feedback is strictly useful, by contrast to the case of *arbitrary* (hence, not necessarily causal) encoding-decoding (Theorem 1).

By analogy to [21, Thm. 1], we obtain the following proposition:

Proposition 1. *There is an optimal causal code f^* of the form*

$$x[i] = f_{n,i}^{*}(u[i], \{y[j]\}_{j=1}^{i-1}). \tag{7.13}$$

This proposition can be proved along the lines of [21, Thm. 1].

7.2.5 Summary — two notions of optimality

In this section, we provided the basic problem setup of this chapter, namely, the remote source-channel communication problem. Two optimality criteria were discussed: On the one hand, information-theoretic optimality, where the encoding and decoding can be done with arbitrary complexity and delay. On the other hand, we studied optimal *causal* encoding and decoding schemes.

It is interesting to point out that in certain special cases, these two notions coincide in the sense that, even with arbitrary complexity and delay, it is not feasible to obtain a performance superior to that of the best causal coding scheme. Such cases can be identified along the lines of [12].

7.3 The Considered Gaussian Sensor Network Scenario

In this chapter, we study a particular sensor network model under which a Gaussian source is observed in noise by M sensors. These sensors communicate over an additive white Gaussian multiple-access channel. It is customary in the communication theory literature to consider a constraint on the power *transmitted* by each node, reflecting the limited battery power at the nodes. However, in our sensor network model, the signals transmitted by the sensor nodes will typically be correlated. For such cases, regulatory bodies (such as the United States Federal Communications Commission (FCC)) are more likely to impose power constraints on the system *as a whole*.[3] As shown elsewhere [8, 9], such a change in perspective leads to significantly different insights and conclusions in a communication network context.

The sensor network model studied in this chapter is illustrated in Figs. 7.2 and 7.3 and described in the sequel.

[3] Such a perspective is suggested by the current formulation of the Code of Federal Regulations (CFR) Title 47, Part 15, Section 15.31(h). See http://www.gpoaccess.gov/cfr/index.html

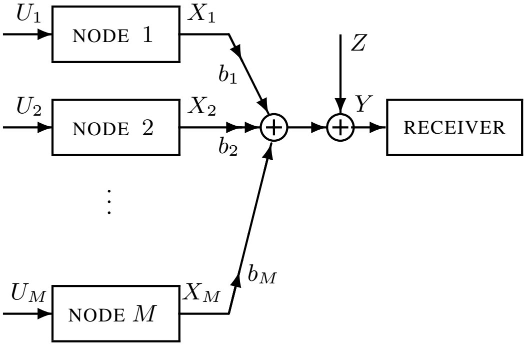


Fig. 7.2. The additive white Gaussian multiple-access channel considered in this chapter.

The communication channel in our sensor network is modeled by the standard additive white Gaussian multiple-access channel as defined e.g. in [4, Sec.14.1.2] and shown in Fig. 7.2. The signals transmitted by the M sensor nodes are complex-valued sequences $\{X_m[i]\}_{i\geq 1}$, for $m = 1, 2, \ldots, M$, selected appropriately by each node. The destination receives the signal

$$Y[i] = Z[i] + \sum_{m=1}^{M} X_m[i], \tag{7.14}$$

where $\{Z[i]\}_{i\geq 1}$ is a sequence of independent and identically distributed (i.i.d.) circularly complex Gaussian random variables of mean zero and variance σ_Z^2.

In the point-to-point case discussed in Section 7.2, the coding schemes had to satisfy *channel input cost* constraints of the form (7.4)–(7.5). Similarly, here, we also consider the communication channel of Fig. 7.2 subject to constraints. Usually (see, e.g., [4, Sec.14.1.2]), the power transmitted by the nodes is constrained:

$$\frac{1}{n}\sum_{i=1}^{n} E\left[|X_m[i]|^2\right] \leq P_m, \tag{7.15}$$

for $m = 1, 2, \ldots, M$. This reflects well the physical limitations of transmitting nodes (finite power etc.). However, when the sequences $\{X_m[i]\}_{i=1}^{n}$, for $m = 1, 2, \ldots, M$, are not selected independently of each other, we argue that it becomes more meaningful to consider a constraint on the *received* power, that is,

$$\frac{1}{n}\sum_{i=1}^{n} E\left[\left|\sum_{m=1}^{M} b_m X_m[i]\right|^2\right] \leq Q. \tag{7.16}$$

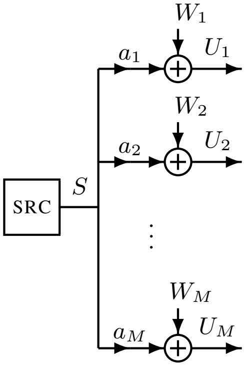


Fig. 7.3. The source structure in the sensor network considered in this chapter: The source information U_m at node m is a noisy version of a common underlying source S, and the overall goal is to reconstruct S to within mean-squared error distortion.

The results provided in this chapter concern the latter case.[4]

The source structure for the sensor network considered in this chapter is illustrated in Fig. 7.3. The underlying data of interest is denoted by $S^n = \{S[i]\}_{i=1}^n$, which we assume to be a sequence of i.i.d. circularly symmetric Gaussian random variables of mean zero and variance σ_S^2.

Sensor m observes a sequence $U_m^n = \{U_m[i]\}_{i=1}^n$ which depends on the source according to a conditional distribution $p(u_1^n, u_2^n, \ldots, u_M^n | s^n)$. For the scope of this chapter, we assume this to be given by

$$U_m[i] = \alpha_m S[i] + W_m[i], \tag{7.17}$$

where $W_m[i]$ are i.i.d. random variables (over i and m) of mean zero and variance σ_W^2.

7.4 Single Channel Use: Feedback Is Useless

Let us now analyze the optimum performance in the sensor network defined in Section 7.3. In order to gain intuition for this problem, recall the following three key facts:

1. Consider the case $M = 1$, and neglect the observation noises (that is, set $\sigma_W^2 = 0$). Then, following [14], it is easy to find the optimum *causal* code: it turns out that multiplying U_1 by an appropriate constant to meet the power constraint is, in fact, the optimum code in an information-theoretic sense, hence, *a fortiori*, the optimum causal code.

[4] For the case of transmit power constraints of the form of Equation (7.15), slightly weaker results were established in [10].

2. Continuing to neglect the noise in the source observation process (i.e., $\sigma_W^2 = 0$), the information-theoretically optimum communication strategy is to make all signals equal, potentially up to scaling (to account for the fact that the coefficients b_m are different). This is sometimes referred to as beam-forming.
3. Neglecting now the noise on the channel, i.e., $\sigma_Z^2 = 0$, it is also well-known that the minimum mean-squared error (MMSE) estimate of S based on the noisy observations $U_1, \ldots, U_M$ is simply a suitable linear combination of the latter. Note that the considered multiple-access channel precisely permits us to form such a linear combination.

These three insights suggest that a good *causal* strategy might be for each sensor to apply an appropriate scaling factor to its respective observation U_m, and to transmit this without further coding. Clearly, however, the three insights above are not sufficient to establish that it is the *best* causal code. In particular, it is unclear whether this will indeed be the best strategy to simultaneously suppress the measurement *and* the communication noises. The following theorem asserts that this is nevertheless the case:

Theorem 2. *For the source structure defined in this section and the additive white Gaussian MAC under a received-power constraint, the optimum performance for causal coding is*

$$D = \frac{\sigma_S^2}{1 + \frac{A\sigma_S^2}{\sigma_W^2}} + \frac{\sigma_S^2}{1 + \frac{\sigma_W^2}{A\sigma_S^2}} \cdot \frac{1}{1 + \frac{Q}{\sigma_Z^2}}, \tag{7.18}$$

where $A = \sum_{m=1}^{M} |a_m|^2$. Moreover, this is the optimum performance in the sense that no other coding scheme, causal or not, can perform better.

The proof of this theorem is given in Appendix B. As discussed above, the optimum causal coding scheme is simple. To establish optimality, in this case, we are able to show that the performance according to Theorem 2 is, in fact, the information-theoretic optimum, hence, cannot be improved upon by removing the causality constraint.

The second question is whether feedback can improve on this performance. While the capacity of memoryless channels cannot be increased by feedback, it is known that in memoryless networks, feedback generally does increase capacity. In particular, this is true for the additive white Gaussian multiple-access channel (under a transmit-power constraint), see [17]. For the same channel, but under a received-power constraint, it can be shown that feedback cannot increase capacity [9]. For the considered sensor network, it follows directly from the proof of Theorem 2 that feedback does not permit us to enhance performance. That is,

Corollary 1. *Feedback is useless.*

The proof argument is given at the end of the converse proof in Appendix B.

Remark 1. Notice that we assume that the coefficients b_m, $m = 1, 2, \ldots, M$, in Fig. 7.2 are assumed to be deterministic (nonrandom) constants.

7.5 Two Channel Uses: Feedback Is Useful

The preceding section concluded that feedback is useless for the considered sensor network situation. In this section, we consider a slight variation on the original setup, and show that this leads to a different conclusion.

In particular, the variation is that the channel can be used *twice* for each source sample. Equivalently, this can be understood as the scenario where the (temporal) bandwidth of the source is half the bandwidth of the communication channel. In comparison to Section 7.4, there is additional communication capability in the setup of this section, and the question becomes that of optimally exploiting this benefit.

In the absence of feedback, one way of exploiting the potential of two channel uses per source sample in a causal coding/decoding context is to simply *repeat* the source simple twice. It is easy to verify that for such repetition coding, the resulting distortion is found to be[5]

$$D = \frac{\sigma_S^2}{1 + \frac{A\sigma_S^2}{\sigma_W^2}} + \frac{\sigma_S^2}{1 + \frac{\sigma_W^2}{A\sigma_S^2}} \cdot \frac{1}{1 + \frac{2Q}{\sigma_Z^2}}, \tag{7.19}$$

where $A = \sum_{m=1}^{M} |a_m|^2$. The general question of whether this can be improved upon (in the absence of feedback) appears to be unanswered, even if the coding and decoding are allowed to incur arbitrary delay and be arbitrarily complex.

Instead, we now consider the scenario where causal, noiseless feedback is available. For the standard point-to-point communication problem, simple feedback strategies have been suggested that permit us to exploit additional channel bandwidth [5, 11, 15, 18]. The strategy is to send, in the second channel use, merely the innovation between the source value that needs to be communicated, and the receiver's current estimate. In extension of this work, the strategy can also be shown to perform optimally in the point-to-point version of the remote source-channel communication scenario, as studied in Section 7.2.

Combining this insight with that of Theorem 2, it is perhaps not surprising that such a "send the innovations only" strategy also works in a distributed setting such as the one studied in this chapter. Somewhat less immediate is the insight that the strategy, in fact, achieves the information-theoretic optimum, as asserted in the following theorem.

Theorem 3. *When noiseless, causal feedback is available, the optimum performance for causal coding is*

$$D = \frac{\sigma_S^2}{1 + \frac{A\sigma_S^2}{\sigma_W^2}} + \frac{\sigma_S^2}{1 + \frac{\sigma_W^2}{A\sigma_S^2}} \left(\frac{1}{1 + \frac{Q}{\sigma_Z^2}} \right)^2, \tag{7.20}$$

[5] It is interesting to note that while suboptimal in general, repetition coding by far outperforms a strategy in which the source is compressed *optimally* in a distributed fashion, and the resulting compression indices are communicated across the multiple-access channel using capacity-achieving codes. See [13] for further detail.

where $A = \sum_{m=1}^{M} |a_m|^2$*. Moreover, this is the optimum performance in the sense that no other coding scheme, causal or not, can perform better.*

The proof of this theorem is given in Appendix C.

For the main conclusion of this chapter, namely, the role of feedback in a causal coding/decoding context, the remaining task is to show that the performance of Theorem 3 cannot be achieved without feedback.

This is difficult because the information-theoretically optimum performance without feedback is *not* known to date: While the information-theoretic source coding problem for the source illustrated in Fig. 7.3 is solved [16], and the capacity of the additive white Gaussian multiple-access channel is known, there is no separation theorem for the overall communication system: it is suboptimal to combine these two codes.

Using a different approach, we can establish the following proposition, proved in Appendix D.

Proposition 2. *For the sensor network considered in this chapter, when there are two channel uses per source sample, the smallest distortion achievable using a causal encoder/decoder pair without feedback is strictly larger than the distortion achievable* with *feedback, given in Theorem 3.*

7.6 Extensions

7.6.1 Collaboration between the sensors

While we have focused in this chapter on the simple communication structure illustrated in Fig. 7.2, it is easy to see that the results apply unchanged to the case where arbitrary (constrained or unconstrained) cooperation between the sensor nodes is available. This is addressed in part in [13].

7.6.2 Arbitrary ratio between source symbols and channel uses

The results and insights presented in Section 7.5 can be extended to the case where for each k source samples, n channel uses are available, where $k \leq n$. In extension of the analysis in [5, 15, 18], the coding scheme is modified to transmit suitable linear combinations of the k source samples that are jointly processed. Innovations are then furnished just like in the proof of Theorem 3.

It is interesting to note that in the converse case, when $k > n$, it is unclear what a successful strategy will look like, even in the simple point-to-point case studied in [5, 15, 18].

7.6.3 Noisy feedback

Another relevant extension concerns *noisy* feedback. From an information-theoretic perspective, even in the point-to-point case, noisy feedback has not been widely studied. The reason is simply that for memoryless channels, feedback does not permit us to boost the performance in the first place. It does, of course, enable simple schemes to attain optimum performance, but when the feedback becomes noisy, these schemes usually break down, and it becomes more attractive to simply disregard the feedback.

The situation is fundamentally different for the sensor network considered in this chapter since the optimum performance without feedback is *unknown*, and in fact, it is unclear what coding scheme to use, other than repetition coding. Hence, feedback, even if noisy, can be exploited to improve the performance over repetition coding.

The performance of the scheme used to establish Theorem 3 under noisy feedback can be analyzed accordingly. In particular, suppose that instead of $Y_1[i]$, a signal $Y_1[i]+V[i]$ is fed back, where $\{V[i]\}$ is a sequence of i.i.d. circularly complex Gaussian random variables of mean zero and variance σ_V^2. Clearly, it is a simple matter to evaluate the effect of this degradation on the scheme of Theorem 3.

7.7 Conclusions

This chapter further investigates the significance of feedback in a causal coding context. It has been shown by Varaiya and Walrand [20, 21] that while feedback is useless if the channel structure is sufficiently symmetric, it can be strictly useful when such symmetry is absent.

This chapter extends their study to a simple Gaussian sensor network case, and reveals a similar behavior: In the symmetric case (which, in our setup, is the case where source and channel bandwidth are equal), feedback turns out to be useless, whereas in the asymmetric case (when the channel bandwidth exceeds the source bandwidth), feedback is strictly useful.

The elegant dynamic programming approach of [20, 21] appears to be hard to extend to a distributed setting such as the one considered here. Therefore, we established our results using a different set of tools.

Acknowledgment. Inspiring discussions with Prof. J. Walrand and Prof. M. Vetterli are gratefully acknowledged. The work in this chapter was supported in part by the National Science Foundation under award CCF-0347298 (CAREER).

A The Remote Source-Channel Communication Problem

In extension of the standard point-to-point joint source-channel communication problem, the key to analyze the communication scenario of Fig. 7.1 turns out to be the remote rate-distortion function [2, 6, 7]:

Definition 1. *Define the remote rate-distortion function for a source S, characterized by a probability distribution $p(s)$, an observation process $p(u|s)$, and a distortion measure $d(s, \hat{s})$ as follows:*

$$R_{remote}(D) = \min I(U; \hat{S}) \tag{7.21}$$

where the minimization is performed over all conditional distributions $p(\hat{s}|u)$ *for which* $E[d(S,\hat{S})] \leq D$.

With this, we now prove Theorem 1.

Proof of Theorem 1. Converse: Consider any source-channel code of block length n, achieving a distortion D. This code induces a distribution $p(\hat{s}^n|u^n)$. It can be shown that for this distribution,

$$I(U^n;\hat{S}^n) \geq nR_{remote}(D). \tag{7.22}$$

By the data processing inequality,

$$I(U^n;\hat{S}^n) \leq I(U^n;Y^n). \tag{7.23}$$

Furthermore, we can expand

$$I(U^n;Y^n) = H(Y^n) - H(Y^n|U^n) \leq H(Y^n) - H(Y^n|X^n,U^n), \tag{7.24}$$

and by the fact that the channel is memoryless,

$$H(Y^n|X^n,U^n) = \sum_{i=1}^{n} H(Y_i|X_i). \tag{7.25}$$

Finally, since

$$H(Y^n) \leq \sum_{i=1}^{n} H(Y_i), \tag{7.26}$$

we conclude that

$$nR_{remote}(D) \leq nC. \tag{7.27}$$

Achievability follows straightforwardly from the operational significance of the remote rate-distortion function and of capacity.

B Proof of Theorem 2

Converse: To obtain an upper bound to the best possible performance, we determine the performance for the "idealized" scenario where all the encoders in Figs. 7.2 and 7.3 are merged into one single device, turning the overall network effectively into a (memoryless) point-to-point (remote) source-channel communication system. For the latter, the optimum performance is described by Theorem 1. In order to evaluate the theorem, we have to determine the remote rate-distortion function for the scenario of Fig. 7.3. Collect all the noisy observation into the vector $U = (U_1, \ldots, U_M)$. Then, we can write

$$D_{remote}(R) = \min E|S - \hat{S}|^2 \tag{7.28}$$

where the minimum is over all conditional distributions $p(\hat{s}|u)$ for which $I(U; \hat{S}) \leq R$. Rewriting this, we find

$$\begin{aligned} D_{remote}(R) &= \min_{p(\hat{s}|u): I(U;\hat{S}) \leq R} E|S - f(U) + f(U) - \hat{S}|^2 \\ &\leq \min_{p(\hat{s}|u): I(U;\hat{S}) \leq R} E|S - f(U)|^2 + E|f(U) - \hat{S}|^2, \end{aligned}$$

with equality if and only if

$$E(S - f(U))(f(U) - \hat{S})^H = 0. \tag{7.29}$$

Let us choose $f(U)$ to be the minimum mean-squared error (MMSE) estimate of S based on U. By the orthogonality principle, Condition (7.29) is satisfied, and the remaining minimization problem is

$$\min_{p(\hat{s}|u): I(U;\hat{S}) \leq R} E|f(U) - \hat{S}|^2,$$

which can be evaluated by standard arguments to yield $Var(f(U))2^{-R}$. Combining terms, we find

$$D_{remote}(R) = \frac{\sigma_S^2 \sigma_W^2}{\sum_{m=1}^M |a_m|^2 \sigma_S^2 + \sigma_W^2} \left(1 + \frac{\sigma_S^2}{\sigma_W^2} \sum_{m=1}^M |a_m|^2 2^{-R} \right). \tag{7.30}$$

The rate available to communicate in the “idealized” scenario is easily determined to be

$$C = \log_2 \left(1 + \frac{Q}{\sigma_Z^2} \right). \tag{7.31}$$

Combining (7.30) with (7.31) yields the claimed formula.

For Corollary 1, simply note that feedback clearly leaves the remote rate-distortion function, Equation (7.30), unchanged, but it also does not affect the capacity of the additive white Gaussian (vector) channel in the “idealized” scenario, Equation (7.31), see e.g. [4, p.256].

Achievability: Let us analyze the following coding scheme: During time slot i, sensor m transmits

$$X_m[i] = \frac{a_m^*}{b_m} \underbrace{\sqrt{\frac{Q}{\sigma_S^2 \left(\sum_{m=1}^M |a_m|^2 \right)^2 + \sigma_W^2 \sum_{m=1}^M |a_m|^2}}}_{=\alpha} U_m[i], \tag{7.32}$$

where a_m^* denotes the complex conjugate of a_m. It is easily verified that this satisfies the imposed constraint on the received sequence $\{Y[i]\}$, which takes the shape

$$Y[i] = \alpha \sum_{m=1}^{M} a_m^* U_m[i] + Z[i]. \tag{7.33}$$

Note that this implies that the joint distribution of $S[i]$ and $Y[i]$ is a multivariate Gaussian distribution. To simplify notation, let us drop the time indices and merely write S and Y instead of $S[i]$ and $Y[i]$, respectively. The distortion incurred by estimating S from Y is well known to be determined by the formula

$$D = E\left[|S|^2\right] - \frac{E\left[SY^*\right] E\left[YS^*\right]}{E\left[|Y|^2\right]}, \tag{7.34}$$

where, from the model definition,

$$E\left[|S|^2\right] = \sigma_S^2, \tag{7.35}$$

and from the received-power constraint,

$$E\left[|Y|^2\right] = Q + \sigma_Z^2. \tag{7.36}$$

Furthermore, the term $E\left[SY^*\right]$ can be evaluated as follows:

$$\begin{aligned} E\left[SY^*\right] &= \alpha \sum_{m=1}^{M} a_m E\left[SU_m^*\right] \\ &= \alpha \sum_{m=1}^{M} a_m a_m^* E\left[SS^*\right] \\ &= \alpha \sum_{m=1}^{M} |a_m|^2 \sigma_S^2. \end{aligned} \tag{7.37}$$

Noting that $E\left[YS^*\right] = \left(E\left[SY^*\right]\right)^*$, the distortion can be evaluated as follows:

$$\begin{aligned} D &= \sigma_S^2 - \frac{\sigma_S^4 \left(\sum_{m=1}^M |a_m|^2\right)^2}{\sigma_S^2 \left(\sum_{m=1}^M |a_m|^2\right)^2 + \sigma_W^2 \sum_{m=1}^M |a_m|^2} \frac{Q}{Q + \sigma_Z^2} \\ &= \sigma_S^2 - \frac{\sigma_S^4 \sum_{m=1}^M |a_m|^2}{\sigma_W^2 + \sigma_S^2 \sum_{m=1}^M |a_m|^2} \frac{Q}{Q + \sigma_Z^2} \\ &= \frac{\sigma_S^2 \sigma_W^2}{\sigma_W^2 + \sigma_S^2 \sum_{m=1}^M |a_m|^2} \left(1 + \frac{\sigma_S^2}{\sigma_W^2} \left(\sum_{m=1}^{M} |a_m|^2\right) \left(1 - \frac{Q}{Q + \sigma_Z^2}\right)\right) \\ &= \frac{\sigma_S^2 \sigma_W^2}{\sigma_W^2 + \sigma_S^2 \sum_{m=1}^M |a_m|^2} \left(1 + \frac{\sigma_S^2}{\sigma_W^2} \left(\sum_{m=1}^{M} |a_m|^2\right) \frac{\sigma_Z^2}{Q + \sigma_Z^2}\right) \\ &= \frac{\sigma_S^2 \sigma_W^2}{\sigma_W^2 + \sigma_S^2 \sum_{m=1}^M |a_m|^2} \left(1 + \frac{\sigma_S^2}{\sigma_W^2 \left(1 + \frac{Q}{\sigma_Z^2}\right)} \sum_{m=1}^{M} |a_m|^2\right), \end{aligned} \tag{7.38}$$

which is the claimed formula.

C Proof of Theorem 3

Converse: Let us proceed exactly along the lines of the proof of Theorem 2, presented in Appendix B. The only change occurs in the *rate* available to communicate, which simply is

$$C = 2\log_2\left(1+\frac{Q}{\sigma_Z^2}\right). \tag{7.39}$$

Combining (7.30) with (7.39) yields the claimed formula.

Achievability: There are now, for each source sample, two channel uses available. In the first channel use, we have

$$Y_1[i] = \underbrace{\sqrt{\frac{Q}{\sigma_S^2\left(\sum_{m=1}^M |a_m|^2\right)^2 + \sigma_W^2\sum_{m=1}^M |a_m|^2}}}_{=\alpha}\sum_{m=1}^M a_m^* U_m[i] + Z[i]. \tag{7.40}$$

In the second channel use, the transmitters select their signals in such a way as to make the received signal

$$Y_2[i] = \beta\left(\sum_{m=1}^M a_m^* U_m[i] - \gamma Y_1[i]\right) + Z_2[i], \tag{7.41}$$

where γ is selected such as to ensure

$$E\left[Y_1^*[i]Y_2[i]\right] = 0. \tag{7.42}$$

Equivalently, γ is selected such that $\gamma Y_1[i]$ is the minimum mean-squared error estimate of $\sum_{m=1}^M a_m^* U_m[i]$ based on $Y_1[i]$, that is

$$\begin{aligned}\gamma &= \frac{E\left[\left(\sum_{m=1}^M a_m^* U_m\right)Y_1^*\right]}{E\left[|Y_1|^2\right]}\\ &= \frac{\alpha^* E\left[\left(\sum_{m=1}^M a_m^* U_m\right)\left(\sum_{m=1}^M a_m^* U_m\right)^*\right]}{E\left[|Y_1|^2\right]}\\ &= \frac{\alpha^* E\left[\left|\sum_{m=1}^M a_m^* U_m\right|^2\right]}{E\left[|Y_1|^2\right]}\end{aligned} \tag{7.43}$$

which can be expressed as $\gamma = \frac{Q}{\alpha(Q+\sigma_Z^2)}$. Hence,

$$\begin{aligned}Y_2[i] &= \beta\left(\sum_{m=1}^M a_m^* U_m[i] - \frac{Q}{Q+\sigma_Z^2}\frac{1}{\alpha}\left(\alpha\sum_{m=1}^M a_m^* U_m[i] + Z_1[i]\right)\right) + Z_2[i]\\ &= \beta\left(\frac{\sigma_Z^2}{Q+\sigma_Z^2}\sum_{m=1}^M a_m^* U_m[i] - \frac{Q}{\alpha(Q+\sigma_Z^2)}Z_1[i]\right) + Z_2[i].\end{aligned} \tag{7.44}$$

It remains to determine β, which follows from ensuring that $E\left[|Y_2[i]|^2\right] \leq Q$. That is,

$$E\left[|Y_2[i]|^2\right] = |\beta|^2 \left(\left(\frac{\sigma_Z^2}{Q+\sigma_Z^2} \right)^2 \frac{Q}{|\alpha|^2} + \frac{1}{|\alpha|^2} \left(\frac{Q}{Q+\sigma_Z^2} \right)^2 \sigma_Z^2 \right) + \sigma_Z^2, \quad (7.45)$$

which yields $\frac{|\beta|^2}{|\alpha|^2} = \frac{Q+\sigma_Z^2}{\sigma_Z^2}$.

Finally, we have to evaluate the distortion incurred when estimating $S[i]$ from $Y_1[i]$ and $Y_2[i]$. Again, from standard results about multivariate Gaussians, the distortion can be expressed as

$$\begin{aligned} D &= \sigma_S^2 - \left(E[SY_1^*] \; E[SY_2^*] \right) \begin{pmatrix} E[|Y_1|^2] & E[Y_1Y_2^*] \\ E[Y_1^*Y_2] & E[|Y_2|^2] \end{pmatrix}^{-1} \begin{pmatrix} E[Y_1S^*] \\ E[Y_2S^*] \end{pmatrix} \\ &= D - \frac{E[SY_1^*]\,E[Y_1S^*]}{E[|Y_1|^2]} - \frac{E[SY_2^*]\,E[Y_2S^*]}{E[|Y_2|^2]}, \end{aligned} \quad (7.46)$$

since our code construction ensures Equation (7.42).

Again, $E\left[|Y_2|^2\right] = Q$ by construction, and the last term to evaluate is

$$\begin{aligned} E[SY_2^*] &= E\left[S \left(\beta \frac{\sigma_Z^2}{Q+\sigma_Z^2} \sum_{m=1}^{M} a_m^* U_m[i] \right)^* \right] \\ &= \beta^* \frac{\sigma_Z^2}{Q+\sigma_Z^2} \left(\sum_{m=1}^{M} a_m E[SU_m^*[i]] \right) \\ &= \beta^* \frac{\sigma_Z^2}{Q+\sigma_Z^2} \left(\sum_{m=1}^{M} |a_m|^2 \sigma_S^2 \right). \end{aligned} \quad (7.47)$$

Finally,

$$\frac{E[SY_2^*]\,E[Y_2S^*]}{E[|Y_2|^2]} = \frac{|\beta|^2 \sigma_Z^4}{(Q+\sigma_Z^2)^3} \left(\sum_{m=1}^{M} |a_m|^2 \sigma_S^2 \right) \sigma_S^4. \quad (7.48)$$

Combining terms yields the claimed formula.

D Proof of Proposition 2

The main point of this proposition can be established by noting that under a *causality* constraint both at the encoder *and* at the decoder, an equivalent problem statement is the following: A Gaussian random variable U can be arbitrarily transformed into the pair $X_1 = f_1(U), X_2 = f_2(U)$, for two arbitrary functions $f_1(\cdot)$ and $f_2(\cdot)$, respectively, satisfying $E[|f_1(U)|^2] \leq P$ and $E[|f_2(U)|^2] \leq P$. Based on $Y_1 = X_1 + Z_1$ and $Y_2 = X_2 + Z_2$, the goal is to estimate U. The resulting mean-squared error is

$$E\left[U - E\left[U|Y_1, Y_2\right]\right] = E\left[\left|U - E\left[U|f_1(U) + Z_1, f_2(U) + Z_2\right]\right|^2\right]. \tag{7.49}$$

The functions $f_1(\cdot)$ and $f_2(\cdot)$ must be selected so as to minimize (7.49).

To bound this distortion, we may use the following lemma (which is a rather direct consequence of the maximum entropy theorem, stating that $h(U - \hat{U}) \leq \log(2\pi e Var(U - \hat{U}))$:

Lemma 1. *If U is Gaussian with mean zero and variance σ_U^2, then for any $\hat{U} \neq U$,*

$$\sigma_U^2 2^{-I(U;\hat{U})} \leq E\left[\left|U - \hat{U}\right|^2\right]. \tag{7.50}$$

Furthermore, using $g(\cdot)$ to denote the estimator function (i.e., the conditional mean of U given $f_1(U) + Z_1$ and $f_2(U) + Z_2$),

$$\begin{aligned} I(U;\hat{U}) &= I(U; g(f_1(U) + Z_1, f_2(U) + Z_2)) \\ &\leq I(U; f_1(U) + Z_1, f_2(U) + Z_2) \end{aligned} \tag{7.51}$$

by the data processing inequality. Hence, consider the problem

$$\max I(U; f_1(U) + Z_1, f_2(U) + Z_2), \tag{7.52}$$

where the $\max$ is taken over all functions $f_1(\cdot)$ and $f_2(\cdot)$ satisfying $E[|f_1(U)|^2] \leq P$ and $E[|f_2(U)|^2] \leq P$.

The distortion achieved in Theorem 3 corresponds to the case when $f_1(U) + Z_1$ and $f_2(U)+Z_2$ are independent and Gaussian, which uniquely maximizes the mutual information in (7.52), and achieves Lemma 1 with equality. However, it follows from standard arguments about multivariate Gaussian distributions that there do not exist (deterministic) functions $f_1(\cdot)$ and $f_2(\cdot)$ for which $f_1(U)$ and $f_2(U)$ are independent Gaussian random variables. Hence, the resulting mutual information $I(U; f_1(U) + Z_1, f_2(U)+Z_2)$ must be strictly smaller, and thus, following Lemma 1, the resulting distortion must be strictly larger than the distortion of Theorem 3. [6]

References

[1] R. Bansal and T. Başar, Simultaneous design of measurement and control strategies for stochastic systems with feedback, *Automatica*, **25** (5):679–694, 1989.

[2] T. Berger, *Rate Distortion Theory: A Mathematical Basis for Data Compression*, Prentice-Hall, Englewood Cliffs, NJ, 1971.

[3] M.V. Burnashev, Data transmission over a discrete channel with feedback, random transmission time, *Problemy Peredachi Informatsii*, **12** (4):10–30, 1976.

[6] While this argument shows that the performance of Theorem 3 cannot be achieved *exactly* without feedback, it leaves open the question of how closely it can be approached. Improved bounds for causal codes may be found via the techniques proposed in [22].

[4] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, Wiley, New York, 1991.

[5] T.J. Cruise, Achievement of rate-distortion bound over additive white noise channel utilizing a noiseless feedback link, *Proc. IEEE (Letters)*, **55**:1102–1103, April 1967.

[6] I. Csiszár and J. Körner, *Information Theory: Coding Theory for Discrete Memoryless Systems*, Academic Press, New York, 1981.

[7] R.L. Dobrushin and B.S. Tsybakov, Information transmission with additional noise, *IRE Transactions on Information Theory*, IT–**18**:S293–S304, 1962.

[8] M. Gastpar, Gaussian multiple access channels under received-power constraints, In *Proc. IEEE Information Theory Workshop*, San Antonio, TX, October 2004.

[9] M. Gastpar, On capacity under received-signal constraints, In *Proc. 42nd Annual Allerton Conference on Communication, Control, and Computing*, Monticello, IL, October 2004.

[10] M. Gastpar, On the role of feedback in large sensor networks, In *Proc. International Zurich Seminar*, Zurich, Switzerland, February 2004.

[11] M. Gastpar and B. Rimoldi, Source-channel communication with feedback, In *Proc. 2002 IEEE Information Theory Workshop*, Paris, France, April 2003, invited paper.

[12] M. Gastpar, B. Rimoldi, and M. Vetterli, To code, or not to code: Lossy source-channel communication revisited, *IEEE Transactions on Information Theory*, **49** (5):1147–1158, May 2003.

[13] M. Gastpar and M. Vetterli, Power, spatio-temporal bandwidth, and distortion in large sensor networks, *IEEE Journal on Selected Areas in Communications (Special Issue on Self-Organizing Distributive Collaborative Sensor Networks)*, 2005, in press.

[14] T.J. Goblick, Theoretical limitations on the transmission of data from analog sources, *IEEE Transactions on Information Theory*, IT–**11** (4):558–567, October 1965.

[15] T. Kailath, An applications of Shannon's rate-distortion theory to analog communication over feedback channels, In *Proc. Princeton Symposium on System Science*, Princeton, NJ, March 1967.

[16] Y. Oohama, The rate-distortion function for the quadratic Gaussian CEO problem, *IEEE Transactions on Information Theory*, IT–**44** (3):1057–1070, May 1998.

[17] L.H. Ozarow, The capacity of the white Gaussian multiple access channel with feedback, *IEEE Transactions on Information Theory*, IT–**30** (4):623–629, July 1984.

[18] J.P.M. Schalkwijk and L.I. Bluestein, Transmission of analog waveforms through channels with feedback, *IEEE Transactions on Information Theory*, IT–**13**:617–619, October 1967.

[19] C.E. Shannon, A mathematical theory of communication, *Bell Sys. Tech. Journal*, **27**:379–423, 623–656, 1948.

[20] P. Varaiya and J.C. Walrand, Optimal causal coding-decoding problems, *Systems and Control Letters*, **3**:189–192, September 1983.

[21] J.C. Walrand and P. Varaiya, Optimal causal coding-decoding problems, *IEEE Transactions on Information Theory*, IT–**29** (6):814 – 820, November 1983.

[22] J. Ziv and M. Zakai, On functionals satisfying a data-processing theorem, *IEEE Transactions on Information Theory*, IT–**19** (3):275–283, May 1973.

8

Cross-layer Design of Control over Wireless Networks

Xiangheng Liu and Andrea Goldsmith [*]

Department of Electrical Engineering, Stanford University, Stanford, CA 94305-9515, USA
liuxh@wsl.stanford.edu, andrea@wsl.stanford.edu

Summary. We consider a set of networked controllers where multiple control systems coexist with their control loops closed over a shared wireless network that induces random delays and packet losses. This system requires a joint design of the wireless network and the controllers, where the design objective is to optimize the control performance. This performance is a complex function of the controller design and the network parameters, such as throughput, packet delay and packet loss probability. Random delays and packet losses in the feedback loop impose new challenges on the optimal controller design. We first investigate controller design with randomly dropped packets. We prove the separation of estimation and control under certain assumptions of the network and show that the Kalman filter can be modified to generate the optimal state estimate when part or all of the observation is lost. The wireless network needs to provide a sufficient throughput for each of the sensor measurements in order to guarantee the stability of the Kalman filter. We then focus on the wireless network design for this controller. The goal of optimizing the control performance imposes implicit tradeoffs on the wireless network design as opposed to the explicit tradeoffs typical in wireless data and voice applications. Specifically, the tradeoffs between network throughput, time delay and packet loss probability are intricate and implicit in the control performance index, which complicates network optimization. We show that this optimization requires a cross-layer design framework, and propose such a framework for a broad class of networked control applications. We then illustrate this framework by a cross-layer optimization of the link layer, MAC layer, and sample period selection in a double inverted pendulum system.

8.1 Introduction

Distributed control over wireless networks has many compelling applications, including automated highway systems [1], automated factories, and smart homes and appliances. The deployment of wireless networks enables new control applications and allows fully mobile operation and flexible installation, while reducing maintenance costs. Building a distributed control system supported by a wireless network

[*] This work is supported by ABB through the Stanford Networking Research Center and by NSF under grant 0120912.

is a challenging task that requires a new design approach to both systems. Many of the design challenges are similar to those in control over the Internet.

Control systems and communication networks are typically designed using very different principles. Traditional control theory requires the feedback data to be accurate, timely and lossless. Conversely, random delay and packet loss are generally accepted in communication network design and, indeed, very hard to avoid. This delay and loss is much more pronounced in wireless networks than in wired networks due to limited spectrum and power, time-varying channel gains, and interference.

Joint design of control and communication is two-fold: the controller design needs to be robust and adaptive to the communication faults such as random delays and packet losses, while the network should be designed with the goal of optimizing the end-to-end control performance. Furthermore, there is a tradeoff between communication and controller performance. From the control perspective, the more knowledge the controller has about the system, the better the control performance is. Additional knowledge about the system is obtained by increasing the number of sensors or sending sensor measurements more frequently. However, this increases the communication burden on the network and the network may become congested. The congestion results in longer delays and more packet losses, which degrade the control performance. Therefore, a joint design of the network and the controller is necessary. Joint design of control and communication has received little attention due to its inherent challenges and interdisciplinary nature. An example of such a joint design can be found in [2], where the controller synthesis and communication rate allocation is solved jointly with an iterative method.

We consider a networked control system as in Fig. 8.1. Multiple control systems coexist and their feedback loops are closed over a shared wireless network. Each control system has a centralized controller. Both the sensor measurements and the control commands need to be communicated wirelessly. Our results in this chapter are closely related to our previous work [3] [4] [5] [6]. In [3] we studied the optimal Kalman filtering updates in the presence of random partial observation losses and the convergence properties as a function of packet loss probabilities. The communication design trade-offs in the link layer and the medium access control (MAC) layer were evaluated in [4] and [5], respectively. Cross-layer design issues of the wireless network for distributed control applications are discussed in [6]. Other related work includes [7] [8] [9] [10].

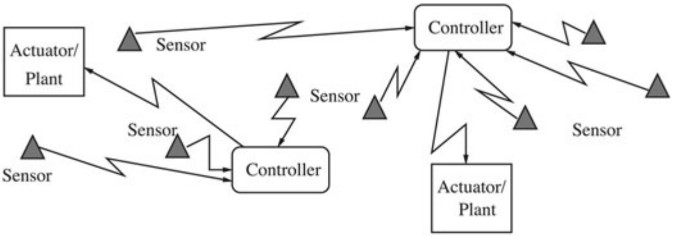


Fig. 8.1. Networked control over wireless.

We cast the joint control and communication design problem in a broader framework of cross-layer design. Cross-layer network design has recently been applied to many applications, such as video over wireless [11] and sensor networks with energy constraints [12]. Different aspects of cross-layer design in wireless ad hoc networks are considered in [13] [14] [15]. We use a cross-layer framework for the joint control and communication problem as it allows each layer of the network protocol stack to be optimized relative to the end-to-end control performance. We will specifically investigate the interaction of the physical layer design, the MAC protocol choice, and the controller sampling period within our cross-layer design framework.

The remainder of this chapter is organized as follows. In Section 8.2 we present the LQG optimal controller design in the presence of random delays and packet losses by proving the separation principle and finding the modified Kalman filter updates with partial observation losses. In Section 8.3 we explain the layered structure of data networks and the cross-layer network design framework for networked control applications. In Section 8.4 we describe our wireless network model. In Section 8.5 we illustrate our iterative cross-layer design of the link layer, MAC layer, and sample period selection with a double inverted pendulum system. Our conclusions and discussion are given in Section 8.6.

8.2 Control System Model and Optimal LQG Control

We assume all the plants in our model are continuous-time linear time-invariant systems and we represent the n^{th} system with the following state space equations:

$$\begin{cases} \dot{\mathbf{x}}^{<n>}(t) = A^{<n>}\mathbf{x}^{<n>}(t) + B^{<n>}\mathbf{u}^{<n>}(t) + \mathbf{w}^{<n>}(t), \\ \mathbf{y}^{<n>}(t) = C^{<n>}\mathbf{x}^{<n>}(t) + \mathbf{v_s}^{<n>}(t). \end{cases}$$

Here $\mathbf{x}^{<n>}(t)$ is the system state, $\mathbf{w}^{<n>}(t)$ is the disturbance acting on the plant, $\mathbf{u}^{<n>}(t)$ is the control force, $\mathbf{y}^{<n>}(t)$ is the measured output, and $\mathbf{v_s}^{<n>}(t)$ is the measurement noise. All boldface variables are vectors.

There are many control performance measures that can be considered and the impact of imperfect communication for different measures can be different. We consider a linear quadratic cost function as our performance measure. Specifically, we want to minimize

$$J_{LQG} = \sum_{n=1}^{M} \lim_{t\to\infty} \mathrm{E}\, \mathbf{x}'^{<n>}(t) Q_x^{<n>} \mathbf{x}^{<n>}(t) + \mathbf{u}'^{<n>}(t) Q_u^{<n>} \mathbf{u}^{<n>}(t),$$

where the weight matrix $Q_x^{<n>} \geq 0$ and $Q_u^{<n>} > 0$, and $\mathbf{x}'$ is the transpose of $\mathbf{x}$. We can tune the system performance by choosing different $Q_x^{<n>}$ and $Q_u^{<n>}$. Note that the controllers are discrete-time and the control output is transformed to a continuous-time signal via a zero-order hold. Thus the closed-loop system is a sample-data system. Minimizing the linear quadratic cost function we consider is equivalent to minimizing the generalized H_2 norm (often abbreviated as the H_2

norm) with proper transformation [16]. Since all the systems have the same state-space representations, we drop the superscript $< n >$ except when needed for clarification.

Different control systems share one wireless network and interact with each other only through the network sharing. The network design determines the packet delay distribution and packet loss probability that affect the control performance. If the delay distribution and packet loss probability are known, we can decouple the design and analysis of different control systems. In the rest of this section, we assume that the delay distribution and packet loss probability are known, and focus on the design and performance evaluation of a single closed-loop system.

In the rest of this section, we first state our assumptions, under which the separation principle of estimation and control is proved even in the presence of random delays and packet losses. The state feedback controller with random control packet losses is derived and the modified Kalman filter updates are shown to be optimal for state estimation with random observation losses. Lastly we summarize our control design and show how to use Markovian Jump Linear System (MJLS) techniques to evaluate the system performance.

8.2.1 Assumptions and timing issues

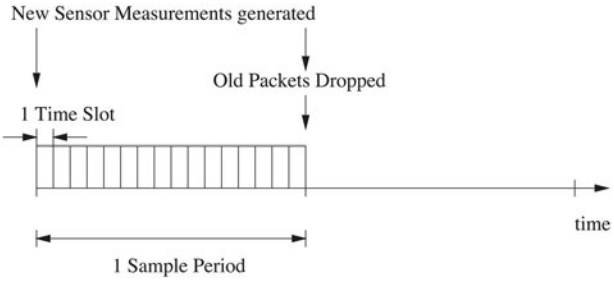


Fig. 8.2. Timing diagram.

We illustrate our timing assumptions in Fig. 8.2. Time is evenly slotted and there are multiple time slots within one sample period. New sensor measurements are generated at the beginning of each sample period and old packets are dropped when new packets are generated. A packet is declared lost if it has not been received by the end of the sample period. Within one time slot, exactly one data packet can be transmitted. The packet may be corrupted during the transmission due to noise and interference. Thus the receiver may not be able to decode the packet. Retransmissions are allowed when there are enough time slots in one sample period. In this chapter, we ignore the propagation time and processing time,[2] and only consider re-

[2] The propagation delay and processing delay are roughly constant. We can also model this delay into the plant dynamics.

transmission delays. Due to the slotted nature, the delay only takes discrete values that are in multiples of the length of a time slot. Note that all delays are bounded by one sample period since packets are dropped at the end of every sample period.

We assume that neither the sensors nor the actuator have computational capabilities. Thus the sensors can only send the measured data, but not a function of the measured data. For the ease of analysis, we assume that the actuator updates at most once per sample period. If no control command is received in a sample period, the actuator continues to use the previous control command until a new one is received.

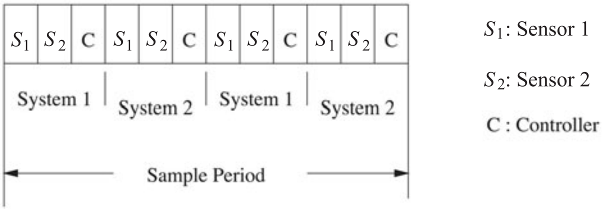


Fig. 8.3. Timing illustration with TDMA.

Depending on the controller design, the control command may depend on the time delay and the sensor measurements available to the controller. The controller can either wait till the receipt of all sensor measurements before any transmission to the actuator, or, in some cases, the controller can choose to send a control command whenever the channel is available. [3] For example, in a Time Division Multiple Access (TDMA) network, each sensor and controller takes turns to transmit in a predetermined order as shown in Fig. 8.3. In this figure, two control systems share the network via TDMA. Each system has two sensors and one controller. If the controller does not transmit in its given time slot, the time slot is wasted anyway. Thus the controller should send a predicted control command based on the current information. In the event of no further control commands received at the actuator, this control command can be used. In this scenario, the actuator may receive more than one control command in a sample period, but we assume that the actuator updates at most once. So the actuator needs to set up certain rules to decide which control command it uses to update. This is also part of the controller design and our design is explained in Subsection 8.2.4.

We also assume that the actuator sends an acknowledgment (ACK) packet to the controller when a control command is successfully received by the actuator. The ACK packets are often small and thus require little network resource. We assume that the ACK from the actuator to the controller experiences no loss with negligible delay. This assumption of timely and lossless ACK on the control command is an

[3] We could also have the controller wait a certain amount of time before it transmits to the actuator.

important one that allows the separation of estimation and control as we shall see in the next subsection.

8.2.2 Separation of estimation and control

The separation principle was shown to be suboptimal in decentralized control by a well-known counterexample [17], where different distributed controllers have access to different information sets. It was argued in [18] that the separation principle does not hold in centralized control when the controller does not know if the previous control commands are received or lost. However, the separation principle has been proven to be optimal under certain assumptions in networked control systems. Gupta et. al. [19] proved the optimality of the separation of estimation and control when the sensor and the controller communicates over a packet-dropping link. The authors assumed no packet losses from the controller to the actuator. Nilsson [20] proved the separation principle with random bounded delays in the feedback loop assuming no packet losses.

Under our assumptions, the controllers do know if a particular control command is received by the actuator and at what time it is executed. In this subsection, we will prove the optimality of the separation principle when both random delays and packet losses are taken into account.

We first review Nilsson's result on the optimal state feedback controller with random bounded delays. We then generalize the result to account for packet losses from the controller to the actuator. Lastly we study the problem of output feedback control with random packet losses in the sensor measurements and we prove the optimality of the separation of estimation and control.

Let τ_k denote the delay of the control command in the k^{th} sample period. This delay is defined to be the time from the beginning of the k^{th} sample period to the time that the control command is executed by the actuator. The sample period is h and $0 < \tau_k \leq h$. We assume that τ_k is available to the controller and will justify this assumption in Subsection 8.2.4. By discretizing the plant dynamics, we get

$$\begin{cases} \mathbf{x}_{k+1} &= \varPhi \mathbf{x}_k + \varGamma_0(\tau_k)\mathbf{u}_k + \varGamma_1(\tau_k)\mathbf{u}_{k-1} + \mathbf{w}_k, \\ \mathbf{y}_k &= C\mathbf{x}_k + \mathbf{v}_{sk}, \end{cases} \tag{8.1}$$

where $\varPhi = e^{Ah}$, $\varGamma_0(\tau_k) = \int_0^{h-\tau_k} e^{As} ds B$, and $\varGamma_1(\tau_k) = \int_{h-\tau_k}^{h} e^{As} ds B$.

At time N we want to minimize a linear quadratic cost function

$$J_N = \mathrm{E}\left\{ \sum_{k=0}^{N} \mathbf{u}_k' Q_u \mathbf{u}_k + \sum_{k=0}^{N} \mathbf{x}_k' Q_x \mathbf{x}_k + \mathbf{x}_{N+1}' Q_N \mathbf{x}_{N+1} \right\}.$$

With state feedback, we have $\mathbf{y}_k = \mathbf{x}_k$ and we first assume no packet losses anywhere in the feedback loop. The optimal state feedback controller is found in Theorem 5.1 in [20]. We restate it here in our setting. The control law that minimizes our cost function is given by

$$\mathbf{u}_k(\tau_k) = -L_k(\tau_k)\begin{bmatrix}\mathbf{x}_k\\ \mathbf{u}_{k-1}\end{bmatrix}, \tag{8.2}$$

where

$$\begin{aligned}
&L_k(\tau_k) = (Q_u + \tilde{S}^{22}_{k+1}(\tau_k))^{-1}\left[\tilde{S}^{21}_{k+1}(\tau_k)\ \tilde{S}^{23}_{k+1}(\tau_k)\right],\\
&\tilde{S}_{k+1}(\tau_k) = G'(\tau_k)S_{k+1}G(\tau_k),\\
&G(\tau_k) = \begin{bmatrix}\Phi\ \Gamma_0(\tau_k)\ \Gamma_1(\tau_k),\\ 0\quad I\qquad 0\end{bmatrix},\\
&S_k = \mathrm{E}_{\tau_k}\left\{F_1'(\tau_k)\begin{bmatrix}Q_x & 0\\ 0 & Q_u\end{bmatrix}F_1(\tau_k) + F_2'(\tau_k)\tilde{S}_{k+1}(\tau_k)F_2(\tau_k)\right\},\\
&F_1(\tau_k) = \begin{bmatrix}I\quad 0\\ -L_k(\tau_k)\end{bmatrix},\\
&F_2(\tau_k) = \begin{bmatrix}I\quad 0\\ -L_k(\tau_k)\\ 0\quad I\end{bmatrix},\\
&S_N = \begin{bmatrix}Q_N & 0\\ 0 & 0\end{bmatrix}.
\end{aligned}$$

Here $\tilde{S}^{ij}_{k+1}(\tau_k)$ is the block (i,j) of the symmetric matrix $\tilde{S}_{k+1}(\tau_k)$.

Now let us consider the state feedback problem with the control command sent to the actuator over a wireless link. Thus control command packets can be lost. We let $\mathbf{u}_k = \mathbf{u}_{k-1}$ when no control command is received in the k^{th} sample period. We want to find the optimal controller when the control command is received with delay τ_k where $0 < \tau_k \leq h$.

The optimal controller can be solved by extending the state space of τ_k in the previous theorem. When a control command is lost, we let $\tau_k = \infty$. Correspondingly, we have $F_1(\infty) = \begin{bmatrix}I & 0\\ 0 & I\end{bmatrix}$ and $F_2(\infty) = \begin{bmatrix}I & 0\\ 0 & I\\ 0 & I\end{bmatrix}$. Since control packets are declared lost at the end of the sample period, we can think of it as a control update of $\mathbf{u}_k = \mathbf{u}_{k-1}$ with a delay h. Thus $\tilde{S}_{k+1}(\infty) = \tilde{S}_{k+1}(h)$. This leads to the following theorem.

Theorem 1. *When the control command is lost with probability p_l, the optimal state feedback controller is given by:*

$$\bar{\mathbf{u}}_k(\tau_k) = -L_k(\tau_k)\begin{bmatrix}\mathbf{x}_k\\ \bar{\mathbf{u}}_{k-1}\end{bmatrix}, \tag{8.3}$$

for $0 < \tau_k \leq h$, where h is the sample period and

$$\begin{aligned}
&L_k(\tau_k) = (Q_u + \tilde{S}^{22}_{k+1}(\tau_k))^{-1}\left[\tilde{S}^{21}_{k+1}(\tau_k)\ \tilde{S}^{23}_{k+1}(\tau_k)\right],\\
&\tilde{S}_{k+1}(\tau_k) = G'(\tau_k)S_{k+1}G(\tau_k),\\
&G(\tau_k) = \begin{bmatrix}\Phi\ \Gamma_0(\tau_k)\ \Gamma_1(\tau_k),\\ 0\quad I\qquad 0\end{bmatrix},
\end{aligned}$$

$$
\begin{aligned}
S_k &= \mathrm{E}_{\tau_k,\tau_k\le h}\left\{F_1'(\tau_k)\begin{bmatrix} Q_x & 0\\ 0 & Q_u\end{bmatrix}F_1(\tau_k)+F_2'(\tau_k)\tilde{S}_{k+1}(\tau_k)F_2(\tau_k)\right\}\\
&\quad + p_l * \left(F_1'(\infty)\begin{bmatrix} Q_x & 0\\ 0 & Q_u\end{bmatrix}F_1(\infty)+F_2'(\infty)\tilde{S}_{k+1}(h)F_2(\infty)\right),\\
F_1(\tau_k) &= \begin{bmatrix} I & 0\\ -L_k(\tau_k) & \end{bmatrix},\quad F_1(\infty)=\begin{bmatrix} I & 0\\ 0 & I\end{bmatrix},\\
F_2(\tau_k) &= \begin{bmatrix} I & 0\\ -L_k(\tau_k) & \\ 0 & I\end{bmatrix},\quad F_2(\infty)=\begin{bmatrix} I & 0\\ 0 & I\\ 0 & I\end{bmatrix},\\
S_N &= \begin{bmatrix} Q_N & 0\\ 0 & 0\end{bmatrix}.
\end{aligned}
$$

Note that the iteration may not converge when the probability of loss, p_l, is too large. This is because the controller cannot update often enough due to packet losses. An upper bound on p_l that guarantees stability can be found via standard MJLS stability criteria [21].

We now consider the general output feedback case where both sensor measurements and control commands need to be sent over a shared wireless network. Thus both the sensor measurements and the control commands can be lost randomly. Let I^k be the information set that is available to the controller at the time of computing the control command in the k^{th} sample period. Note that I^k is a random set that depends on the packet dropping patterns in the current and previous sample periods. The maximum possible set of I^k is $\{\mathbf{y}_0,\mathbf{y}_1,\ldots,\mathbf{y}_k,\mathbf{u}_1,\ldots,\mathbf{u}_{k-1}\}$. Based on our assumption, the controllers always know the past controls $\{\mathbf{u}_1,\ldots,\mathbf{u}_{k-1}\}$ and the corresponding execution times $\{\tau_1,\ldots,\tau_{k-1}\}$, but the controllers may only have a subset knowledge of the observation $\{\mathbf{y}_0,\mathbf{y}_1,\ldots,\mathbf{y}_k\}$. In the next theorem, we prove the separation principle is optimal for the output feedback problem with random packet losses. Hence the separation principle still holds even in the presence of packet losses.

Theorem 2. *For the output feedback control problem, the optimal controller that generates a control command to be updated at time $kh+\tau_k$ with information set I^k is*

$$
\mathbf{u}_k(\tau_k,I_k) = -L_k(\tau_k)\begin{bmatrix}\mathrm{E}[\mathbf{x}_k|I^k]\\ \mathbf{u}_{k-1}\end{bmatrix}, \tag{8.4}
$$

where L_k is calculated as in Theorem 1 and $\mathrm{E}[\mathbf{x}_k|I^k]$ *is the state estimate given the information set I^k.*

Proof. We prove the separation by writing the iteration of the quadratic cost function as

$$
\begin{aligned}
&J_{k+1} = J_k + \mathrm{E}\mathbf{w}_k' S_{k+1}^{11}\mathbf{w}_k +\\
&\mathrm{E}_{\tau_k,\tau_k\le h}\mathrm{E}_{I_k|\tau_k}(\mathbf{u}_k(\tau_k,I_k)-\bar{\mathbf{u}}_k(\tau_k))'(Q_u+\tilde{S}_{k+1}^{22}(\tau_k))(\mathbf{u}_k(\tau_k,I_k)-\bar{\mathbf{u}}_k(\tau_k))
\end{aligned}
$$

where

$$
\bar{\mathbf{u}}_k(\tau_k) = -L_k(\tau_k)\begin{bmatrix}\mathbf{x}_k\\ \mathbf{u}_{k-1}\end{bmatrix}
$$

is the optimal control with perfect state information $\mathbf{x}_k$.

To minimize J_{k+1}, the optimal control $\mathbf{u}_k(\tau_k, I_k)$ is the minimum mean square estimate of $\bar{\mathbf{u}}_k(\tau_k)$. Recall that I_k is the information set available for calculating $\mathbf{u}_k(\tau_k, I_k)$. Since $\bar{\mathbf{u}}_k(\tau_k)$ is a linear function of $\mathbf{x}_k$ and $\mathbf{u}_{k-1}$, and $\mathbf{u}_{k-1}$ is known, the minimum mean square estimate of $\bar{\mathbf{u}}_k(\tau_k)$ is

$$\mathbf{u}_k(\tau_k, I_k) = \mathrm{E}[\bar{\mathbf{u}}_k(\tau_k)|I^k] = -L_k(\tau_k)\begin{bmatrix}\mathrm{E}[\mathbf{x}_k|I^k] \\ \mathbf{u}_{k-1}\end{bmatrix},$$

where $\mathrm{E}[\mathbf{x}_k|I^k]$ is the minimum mean square estimate of the state variable given the information set I^k. Note that this holds regardless of the information set I^k, which is probabilistic due to the random packet losses of the sensor measurements. □

This proof extends the separation principle to the case of partial observation losses and control command losses. The optimal controller can be separated into two cascaded parts: the state estimator and the state feedback controller. The state feedback controller can be computed using Theorem 1. The optimal state estimator needs further investigation. In the next subsection, we show how the Kalman filter can be modified to adapt to sensor measurement losses.

8.2.3 Kalman filtering in the presence of partial observation losses

In this subsection, we show how to use a modified Kalman filter to calculate $\mathrm{E}[\mathbf{x}_k|I^k]$ when partial observation losses are possible. For illustration purposes, we assume that the observation vector $\mathbf{y}_\mathrm{k}$ is divided into two parts $[\mathbf{y}_{1,\mathrm{k}}; \mathbf{y}_{2,\mathrm{k}}]$ and each part is encoded and sent separately. All the results can be similarly extended to the case where the observation is composed of more than two parts. Since the past controls and their execution times are known at the state estimator, it is sufficient to consider the system with the following dynamics:

$$\begin{aligned}\mathbf{x}_{k+1} &= \Phi\mathbf{x}_k + \mathbf{w}_k, \\ \begin{bmatrix}\mathbf{y}_{1,k} \\ \mathbf{y}_{2,k}\end{bmatrix} &= \begin{bmatrix}C_1 \\ C_2\end{bmatrix}\mathbf{x}_k + \begin{bmatrix}\mathbf{v}_{1,k} \\ \mathbf{v}_{2,k}\end{bmatrix},\end{aligned} \tag{8.5}$$

where $\mathbf{x}_\mathrm{k} \in \mathcal{R}^\mathrm{n}, \mathbf{y}_{1,\mathrm{k}}, \mathbf{v}_{1,\mathrm{k}} \in \mathcal{R}^{\mathrm{m}_1}$, and $\mathbf{y}_{2,\mathrm{k}}, \mathbf{v}_{2,\mathrm{k}} \in \mathcal{R}^{\mathrm{m}_2}$. The system matrices are of the appropriate dimensions. The covariance matrices of $\mathbf{v}_{1,\mathrm{k}}$ and $\mathbf{v}_{2,\mathrm{k}}$ are R_{11} and R_{22} respectively. Comparing with the system dynamics in Equation (8.1), we have $\mathbf{y}_\mathrm{k} = \begin{bmatrix}\mathbf{y}_{1,k} \\ \mathbf{y}_{2,k}\end{bmatrix}$, $\mathrm{C} = \begin{bmatrix}C_1 \\ C_2\end{bmatrix}$. Note that the covariance matrix of $\mathbf{v}$ is $\mathrm{R} = \begin{bmatrix}R_{11} & R_{12} \\ R_{21} & R_{22}\end{bmatrix}$ and the covariance matrix of $\mathbf{w}$ is denoted as Q. We assume the system (Φ, C) is observable, thus the Kalman filter converges without sensor measurement losses.

The measurement outputs $\mathbf{y}_{1,k}$, $\mathbf{y}_{2,k}$ are encoded separately and sent over different wireless channels in time step k. We use $\gamma_{i,k}$ to indicate whether $\mathbf{y}_{i,k}$ is received correctly in time step k. We assume $\gamma_{1,k}$ and $\gamma_{2,k}$ are i.i.d. Bernoulli random variables for all k with $\mathrm{Pr}(\gamma_{1,k} = 1) = \lambda_1$ and $\mathrm{Pr}(\gamma_{2,k} = 1) = \lambda_2$. Note that λ_1 and

λ_2 represent the percentage of the sensor measurement packets that are correctly received. Also note that λ_1 and λ_2 are proportional to the link throughput from sensor 1 to the Kalman filter and the link throughput from sensor 2 to the Kalman filter, respectively. We refer to the pair (λ_1, λ_2) as the network throughput, which depends on the channel gains, the network traffic and the network resource allocation (such as power, time slots, etc.).

We also assume that $\gamma_{1,k}$ and $\gamma_{2,l}$ are independent for every k and l. Thus, $\mathbf{y}_{1,k}$ and $\mathbf{y}_{2,k}$ can be independently lost or received. When an observation is lost, it is equivalent to receiving the measurement with an infinite noise variance. The measurement noise $\mathbf{v}_{i,k}$ is assumed to have the following conditional probability density distribution:

$$p(\mathbf{v}_{i,k}|\gamma_{i,k}) \sim \begin{cases} \mathcal{N}(0, R_{ii}) \text{ if } \gamma_{i,k} = 1, \\ \mathcal{N}(0, \sigma_i^2 I) \text{ if } \gamma_{i,k} = 0, \end{cases} \tag{8.6}$$

where we take $\sigma_{\mathrm{i}}^2 \to \infty$ when the observation $\mathbf{y}_{i,k}$ is lost. We assume that the cross-correlation terms R_{12} and R_{21} do not change as a function of $\gamma_{1,k}$ and $\gamma_{2,k}$. In fact, the measurement noise at different sensors are often uncorrelated.

Let $\gamma_k = [\gamma_{1,k}; \gamma_{2,k}]$, $\gamma_0^k = \{\gamma_0, \ldots, \gamma_k\}$, and $\mathbf{y}_0^k = \{\mathbf{y}_0, \ldots, \mathbf{y}_k\}$. We define

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} &\equiv E[\mathbf{x}_k|\mathbf{y}_0^k, \gamma_0^k], \\ P_{k|k} &\equiv E[(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k})(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k})^{'}|\gamma_0^k], \\ \hat{\mathbf{x}}_{k+1|k} &\equiv E[\mathbf{x}_{k+1}|\mathbf{y}_0^k, \gamma_0^k], \\ P_{k+1|k} &\equiv E[(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k})(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k})^{'}|\gamma_0^k]. \end{aligned}$$

The time update of the Kalman filter is independent of the observation process and thus stays the same as in the Kalman filter with no packet losses,

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k} &= \Phi\hat{\mathbf{x}}_{k|k}, \\ P_{k+1|k} &= \Phi P_{k|k}\Phi' + Q. \end{aligned} \tag{8.7}$$

But the measurement update is now stochastic since the received measurements now depend on the random variables $\gamma_{1,k}$ and $\gamma_{2,k}$.

When $\gamma_{1,k} = 1, \gamma_{2,k} = 1$, the complete observation measurements are received. Thus, the measurement update is the same as in the Kalman filter with no packet losses.

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k+1} &= \hat{\mathbf{x}}_{k+1|k} + P_{k+1|k}C'[CP_{k+1|k}C' + R]^{-1}(\mathbf{y}_{k+1} - C\hat{\mathbf{x}}_{k+1|k}), \\ P_{k+1|k+1} &= P_{k+1|k} - P_{k+1|k}C'[CP_{k+1|k}C' + R]^{-1}CP_{k+1|k}. \end{aligned} \tag{8.8}$$

When $\gamma_{1,k} = 0, \gamma_{2,k} = 0$, the optimal measurement update is to run one step open loop. This also corresponds to the case of no observation in [22]:

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k+1} &= \hat{\mathbf{x}}_{k+1|k}, \\ P_{k+1|k+1} &= P_{k+1|k}. \end{aligned} \tag{8.9}$$

When $\gamma_{1,k} = 1, \gamma_{2,k} = 0$, only $\mathbf{y}_{1,k}$ is received by the Kalman filter. The corresponding measurement noise covariance matrix is now

$$\tilde{R} = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & \sigma_2^2 I \end{bmatrix} = \mathrm{R} + \begin{bmatrix} 0 & 0 \\ 0 & \sigma_2^2 I - R_{22} \end{bmatrix}. \tag{8.10}$$

With the observation $\mathbf{y}_{1,k}$ only, the Kalman filter updates assuming the noise covariance is $\tilde{R}$:

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k+1} &= \hat{\mathbf{x}}_{k+1|k} + P_{k+1|k}C'[CP_{k+1|k}C' + \tilde{R}]^{-1}(\mathbf{y}_{k+1} - C\hat{\mathbf{x}}_{k+1|k}), \\ P_{k+1|k+1} &= P_{k+1|k} - P_{k+1|k}C'[CP_{k+1|k}C' + \tilde{R}]^{-1}CP_{k+1|k}. \end{aligned} \tag{8.11}$$

Note that

$$\begin{aligned} & C'[CXC' + \tilde{R}]^{-1}C \\ &= C'\left(CXC' + \mathrm{R} + \begin{bmatrix} 0 & 0 \\ 0 & \sigma_2^2 I - R_{22} \end{bmatrix}\right)^{-1} C \\ &\overset{(a)}{=} C'\left(CXC' + \mathrm{R} + \begin{bmatrix} 0 & 0 \\ 0 & \sigma_2^2 I \end{bmatrix}\right)^{-1} C \\ &\overset{(b)}{=} C'\begin{bmatrix} \mathcal{M}_{11} - \mathcal{M}_{12}\mathcal{M}_{22}^{-1}\mathcal{M}_{21} & 0 \\ 0 & 0 \end{bmatrix} C \\ &\overset{(c)}{=} C'\begin{bmatrix} (C_1XC_1^{'} + R_{11})^{-1} & 0 \\ 0 & 0 \end{bmatrix} C \\ &\overset{(d)}{=} C_1^{'}[C_1XC_1^{'} + R_{11}]^{-1}C_1, \end{aligned}$$

where

$$[CXC' + \mathrm{R}]^{-1} = \begin{bmatrix} \mathcal{M}_{11} & \mathcal{M}_{12} \\ \mathcal{M}_{21} & \mathcal{M}_{22} \end{bmatrix},$$

and (a) follows since $\sigma_2^2 \to \infty$, (b) is derived by using the low rank adjustment of the matrix inversion formula [23] and taking $\sigma_2 \to \infty$, (c) is due to the alternative formula of the inverse of a partitioned matrix [23], and (d) is derived by simple multiplication of the partitioned matrices.

Therefore, for $\gamma_{1,k} = 1$ and $\gamma_{2,k} = 0$, the measurement update is

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k+1} &= \hat{\mathbf{x}}_{k+1|k} + P_{k+1|k}C_1^{'}[C_1P_{k+1|k}C_1^{'} + R_{11}]^{-1}(\mathbf{y}_{1,k+1} - C_1\hat{\mathbf{x}}_{k+1|k}), \\ P_{k+1|k+1} &= P_{k+1|k} - P_{k+1|k}C_1^{'}[C_1P_{k+1|k}C_1^{'} + R_{11}]^{-1}C_1P_{k+1|k}. \end{aligned} \tag{8.12}$$

Note this is equivalent to the classical Kalman filter measurement update if $\mathbf{y}_1$ were the only observation. This is somewhat surprising because it seems that the Kalman filter does not distinguish between a packet loss and a non-existing sensor measurement. On the other hand, the result can be expected because the Kalman filter is now stochastic and it only depends on the current packet arrivals.

Similarly, when $\gamma_{1,\mathrm{k}} = 0, \gamma_{2,\mathrm{k}} = 1$, the Kalman filter updates as if $\mathbf{y}_2$ were the only observation:

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k+1} &= \hat{\mathbf{x}}_{k+1|k} + P_{k+1|k}C_2^{'}[C_2P_{k+1|k}C_2^{'} + R_{22}]^{-1}(\mathbf{y}_{2,k+1} - C_2\hat{\mathbf{x}}_{k+1|k}), \\ P_{k+1|k+1} &= P_{k+1|k} - P_{k+1|k}C_2^{'}[C_2P_{k+1|k}C_2^{'} + R_{22}]^{-1}C_2P_{k+1|k}. \end{aligned} \tag{8.13}$$

Let $P_k = P_{k|k-1}$. Combining (8.7) (8.8) (8.9) (8.12) (8.13), we get:

$$\begin{aligned} P_{k+1} &= \Phi P_k\Phi' + Q - \gamma_{1,k}\gamma_{2,k}\Phi P_kC'(CP_kC' + R)^{-1}CP_k\Phi' \\ &\quad -\gamma_{1,k}(1 - \gamma_{2,k})\Phi P_kC_1^{'}(C_1P_kC_1^{'} + R_{11})^{-1}C_1P_k\Phi' \\ &\quad -(1 - \gamma_{1,k})\gamma_{2,k}\Phi P_kC_2^{'}(C_2P_kC_2^{'} + R_{22})^{-1}C_2P_k\Phi'. \end{aligned} \tag{8.14}$$

As we have shown, the Kalman filter updates become stochastic when part or all of the observation measurements can be lost randomly. Due to the stochastic nature, we no longer have a unique deterministic error covariance matrix in the steady state. We define

$$\begin{aligned} g_{\lambda_1\lambda_2}(X) &= \Phi X\Phi' + Q - \lambda_1\lambda_2\Phi XC'(CXC' + R)^{-1}CX\Phi' \\ &-\lambda_1(1-\lambda_2)\Phi XC_1^{'}(C_1XC_1^{'} + R_{11})^{-1}C_1X\Phi' \\ &-(1-\lambda_1)\lambda_2\Phi XC_2^{'}(C_2XC_2^{'} + R_{22})^{-1}C_2X\Phi', \end{aligned} \tag{8.15}$$

as a useful shorthand since

$$\mathrm{E}[P_{k+1}|P_k] = g_{\lambda_1\lambda_2}(P_k), \tag{8.16}$$

and

$$\mathrm{E}[P_{k+1}] = \mathrm{E}[g_{\lambda_1\lambda_2}(P_k)]. \tag{8.17}$$

We now state the main theorems of our results on Kalman filtering in the presence of partial observation losses. The detailed development and proofs can be found in [3].

The convergence of the iteration $\bar{P}_{k+1} = g_{\lambda_1\lambda_2}(\bar{P}_k)$ guarantees the boundedness of P_k for any k. The first theorem establishes the condition under which the iteration $\bar{P}_{k+1} = g_{\lambda_1\lambda_2}(\bar{P}_k)$ converges. It also proves the uniqueness of the solution when it does converge.

Theorem 3. *Suppose* $\exists$ *a matrix* $P > 0$ *such that* $P > g_{\lambda_1\lambda_2}(P)$. *Then:*
(a) $\forall \mathrm{P}_0 \geq 0$, *the iteration* $\bar{P}_{k+1} = g_{\lambda_1\lambda_2}(\bar{P}_k)$ *converges and*

$$\lim_{k\to\infty} \bar{P}_k = \lim_{k\to\infty} g^k_{\lambda_1,\lambda_2}(\bar{P}_0) = \bar{P} \tag{8.18}$$

independent of initial condition $\bar{P}_0$*; and*
(b) $\bar{P}$ *is the unique positive semidefinite solution of* $\bar{P}_{k+1} = g_{\lambda_1\lambda_2}(\bar{P}_k)$.

The steady state error covariance matrix $\bar{P}$ can be computed efficiently by solving a semidefinite program (SDP). It can be shown that $\bar{P}$ is an upper bound of the expected error covariance. The next theorem shows the existence of a stability region boundary such that the expected error covariance matrix goes to infinity if the throughput pair is less than the rates specified by the boundary.

Theorem 4. *Assume* (Φ, Q) *is controllable and* (Φ, C) *is observable. Fix* $0 \leq \lambda_1 \leq 1$. *If* $\bar{P}_{k+1} = g_{\lambda_1\lambda_2}(\bar{P}_k)$ *is unstable for* $\lambda_2 = 0$ *while stable for* $\lambda_2 = 1$, *then* $\exists\ \lambda_2^c$ *with* $0 \leq \lambda_2^c \leq 1$ *such that*

$$\lim_{t\to\infty} \mathrm{E}[P_k] = \infty \ \textit{for} \ 0 \leq \lambda_2 \leq \lambda_2^c,$$

and there exists a positive semidefinite matrix $M_{P_0} > 0$ *as a function of the initial condition* $P_0 \geq 0$ *such that*

$$\mathrm{E}[P_k] \leq M_{P_0}\ \forall k \ \textit{for} \ \lambda_2^c < \lambda_2 \leq 1.$$

If $\bar{P}_{k+1} = g_{\lambda_1\lambda_2}(\bar{P}_k)$ *is unstable for the given* λ_1 *when* $\lambda_2 = 1$, *then* $\lambda_2^c = 1$. *If* $\bar{P}_{k+1} = g_{\lambda_1\lambda_2}(\bar{P}_k)$ *is stable for the given* λ_1 *when* $\lambda_2 = 0$, *then* $\lambda_2^c = 0$. *We will get the same stability region boundary if we fix* λ_2 *and vary* λ_1.

Even though this boundary cannot always be found, the upper and lower bounds of the stability region boundary can always be found by solving a feasibility problem of a Linear Matrix Inequality (LMI).

8.2.4 Controller design

The previous subsection shows that the optimal LQG controller with random packet losses has two cascaded parts: the Kalman filter and the state feedback controller. The Kalman filter calculates the minimum mean square error state estimate based on received sensor measurements. When all the sensor measurements are received, the classical steady state Kalman filter is used. When none of the sensor measurements are received, we can have the Kalman filter run a one step forward open loop, and this also gives the optimal state estimate. When only part of the sensor measurements are received, the Kalman filter updates as if the received measurements are the only measurements taken. Note that the optimal Kalman filter is a function of the previous error covariance matrix, which depends on the whole history of packet losses. This makes the computation of Kalman filter gains highly complex. We calculate the steady state error covariance matrix of the iteration $\bar{P}_{k+1} = g_{\lambda_1 \lambda_2}(\bar{P}_k)$. We use this covariance matrix $\bar{P}$ to calculate the Kalman filter gains. Recall that $\bar{P}$ is an upperbound of the expected error covariance matrix. Thus the filter does not output the optimal state estimate but the performance degradation is minimal. The Kalman filter gains now only depend on the sensor measurement losses in the current time slot and thus are much easier to compute.

The state feedback controller is a function of the total time delay in the feedback loop. Thus, it is time varying. The total time delay is from the time when measurements are taken to the time when the actuator updates with the received control command. We assume that the control command is calculated based on the available sensor measurement information right before the transmission to the actuator. Recall that we assume there is a reliable ACK for every transmission. Therefore, the controller knows the time delay of the control command if the next transmission is successful. If a control command is lost, its value does not affect the control system. Therefore it is reasonable to assume that the controller knows the delay τ_k at the time of computing $\mathbf{u}_k$.

Upon receiving a control command, the actuator needs to decide if it should update with the received control command. We assume that there is an indicator bit in the control command packet that tells the actuator whether the control command is computed based on full measurement information or partial information. When the actuator receives a control command that is computed based on full observation information, the actuator updates itself and does not update until the next sample period. In case of receiving a control command based on partial measurement information, the actuator will hold the command and only update with this command if no further control command is received by the end of the sample period. Such an update only occurs at the end of the sample period. When no control command is received, the actuator continues to use its previous command until a new one is received in the next sample period. Note that this is just our design and we are not optimizing this part of the controller. We hope to study this problem in the future.

8.2.5 Performance evaluation

We evaluate the system performance by modeling the closed loop system as a MJLS. Define the augmented system state vector, $\hat{\mathbf{x}}(k) = [\tilde{\mathbf{x}}(k); \hat{\mathbf{x}}(k|k-1); \mathbf{y}(k-1); \mathbf{u}(k-1)]$ and the joint noise vector $\hat{\mathbf{w}}(k) = [\tilde{\mathbf{w}}(k); \mathbf{v}(k)]$, where $\mathbf{v}(k) = \mathbf{v_s}(k) + \mathbf{v_q}(k)$ and $\mathbf{v_s}(k)$ is the measurement noise and $\mathbf{v_q}(k)$ is the quantization noise. Note that $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{w}}$ are the discretized state and disturbance and $\hat{\mathbf{x}}(k|k-1)$ is the Kalman filter state estimate. We choose the Markovian state $r = (D, s_c, s_m)$ where D is the time delay in the control command, s_c indicates the control command loss and the sensor measurement information available to the controller at the time of command calculation, and s_m indicates the sensor measurement loss at the estimator:

$$s_c = \begin{cases} 0 \text{ control command is based on full observation,} \\ 1 \text{ control command is based on } \mathbf{y_1} \text{ only,} \\ 2 \text{ control command is based on } \mathbf{y_2} \text{ only,} \\ 3 \text{ control command is based on no new measurements,} \\ 4 \text{ control command is lost.} \end{cases}$$

$$s_m = \begin{cases} 0 \text{ no sensor measurement losses,} \\ 1 \text{ sensor measurements } \mathbf{y_1} \text{ received but } \mathbf{y_2} \text{ lost,} \\ 2 \text{ sensor measurements } \mathbf{y_2} \text{ received but } \mathbf{y_1} \text{ lost,} \\ 3 \text{ all sensor measurements are lost.} \end{cases}$$

Note that for all $D < T$, where T is the sample period, we always have $s_c = s_m = 0$ while when $D = T$, we can have $s_c = 0$ and $s_m = 0$, or $s_c = 1$ and $s_m = 0, 1$, or $s_c = 2$ and $s_m = 0, 2$, or $s_c = 3$ and $s_m = 0, 1, 2, 3$, or $s_c = 4$ and $s_m = 0, 1, 2, 3$.[4] Therefore we have $L + 12$ Markovian states, where L is the number of time slots in one sample period. We can write the system in the form of a MJLS as $\hat{\mathbf{x}}(k+1) = F_r \hat{\mathbf{x}}(k) + G_r \hat{\mathbf{w}}(k)$ for $r = 1, 2, \ldots, L+12$. The system matrices F_r, G_r can be easily derived. Let $\Sigma_x(k) = \mathrm{E}\, \hat{\mathbf{x}}(k)\hat{\mathbf{x}}(k)'$, then

$$\Sigma_x(k+1) = \sum_{r=1}^{L+12} q_r F_r \Sigma_x(k) F_r' + \sum_{r=1}^{L+12} q_r G_r G_r',$$

where q_r is the probability that the MJLS is in state r. As $k \to \infty$, it can be shown [20] that a unique steady-state covariance matrix $\Sigma_x = \lim_{k\to\infty} \Sigma_x(k)$ exists when the recursion is stable. We can now evaluate the linear quadratic cost function since $J_{LQG} = \mathrm{Trace}\left(\begin{bmatrix} Q\ 0\ 0\ 0 \end{bmatrix} \Sigma_x\right) + \mathrm{Trace}\left(\begin{bmatrix} 0\ 0\ 0\ R \end{bmatrix} \Sigma_x\right)$.

We have found the optimal controller that adapts to the random delays and packet losses in the feedback loop. In the following sections, we will study the network design issues in order to minimize the performance degradation. In the next section, we first motivate why a cross-layer framework is necessary.

[4] The delay distribution that we use to calculate the state feedback controller is exactly the distribution of D when a control command is received. Note that $Pr(D = T)$ sums up nine different probabilities.

8.3 Cross-layer Network Design for Distributed Control

A layered network architecture is central to most data network designs. Layering provides design modularity that facilitates standardization and implementation. An international standard of an Open System Interconnection (OSI) model includes seven layers from top to bottom: the Application layer, the Presentation layer, the Session Layer, the Transport layer, the Network Layer, the medium access control (MAC) layer, and the Physical layer. Traditionally, each layer is designed separately with control messages passing between adjacent layers. The idea of cross-layer design is to jointly design these different layers. Cross-layer design can imply a joint design across all network layers simultaneously, which is highly complex. Alternatively, it can entail choosing parameters or protocols at different network layers from existing designs in a joint fashion, which is our design approach. The goal of cross-layer design is to provide the best possible end-to-end performance of the application. Application examples include voice, video, web browsing, and high speed data transfer. Cross-layer design has shown significant performance benefits for applications with hard delay constraints, such as video [11]. In joint control and network design, the application is control.

We consider a simpler four-layer architecture for cross-layer design to illustrate the benefits in control applications. These layers are shown in Fig 8.4. The physical layer defines a point-to-point communication link. The MAC layer defines how the channel is shared among multiple transmitters. The network layer implements routing and flow control for the network. The application layer supports distributed control. Therefore, we consider the control system design as parameters in the application layer of our four layer network model.

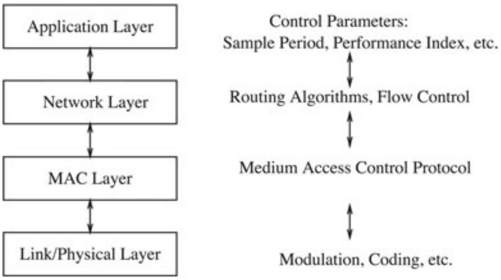


Fig. 8.4. Layered structure of wireless network.

The goal of the Network, MAC, and Link layers is to optimize control performance. This performance is a complicated function of the packet delay distribution, the probability of packet loss and the data resolution associated with the network. Note that the average delay, which is often used as a performance metric in other wireless systems, is a useless metric for control applications since the closed loop system performance depends on the full delay distribution, not just on the average delay. The link design, the MAC protocol, and the routing algorithm jointly affect the delay distribution and the packet loss probability. The sample period of the control system is considered as a parameter of the application layer. The sample period determines how often new packets are generated and when the old packets are dropped. Thus the sample period affects the network traffic, which in turn affects the delay distribution and packet losses. Therefore, it is important to design the parameters of the MAC layer, link layer, and controller jointly. However, it is difficult to quantify the impact of a network design on the control performance analytically for all control systems. This is because the control performance is an implicit and intricate function of the network parameters. We use a numerical example with classical inverted pendulum systems to illustrate our framework for cross-layer design and its associated performance gains.

Cross layer network design over all layers is a very challenging problem. In particular, it is difficult to simultaneously optimize all the layers, which motivated the OSI model in the first place. Thus, we study a suboptimal iterative method for cross-layer optimization over a subset of the network layers: the link layer, the MAC layer and the application layer (sample period selection). In particular, to jointly design the MAC protocol, the link design and select the optimal sample period, we first fix a sample period and a MAC protocol and choose the best link layer design. For this link design and the sample period, we choose the best MAC protocol. The third step is to optimize the sample period for the chosen link and MAC protocol design. We then iterate the algorithm until it converges. It is important to point out that we optimize the controller design for each network design choice. Even though this network optimization is only based on just a few protocol parameters, it can yield significant performance gains and insight as we show in Section 8.5. In the next section, we describe our wireless network model.

8.4 The Wireless Network Model

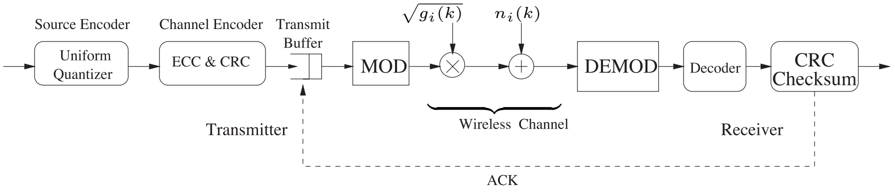


Fig. 8.5. Wireless communication link model.

8.4.1 Wireless channel model

We consider a discrete time channel with stationary, ergodic, slowly time-varying gain $\sqrt{g_i(k)}$ and additive white Gaussian noise (AWGN) $n_i(k)$, where the subscript i refers to the i^{th} link and k refers to the k^{th} time instant. We assume the channel gains are static. This is justified by the assumption of very slow fading where the channel coherence time (the time over which the channel remains roughly constant) is long enough so that the control system converges to steady-state within a coherence time interval. We assume that the channel power gain $g_i(k)$ is independent of the channel input and the transmission power P_i does not change as the channel gain varies.

8.4.2 Wireless link model

Different link layer design choices (coding, modulation, etc.) lead to different performance in terms of data rate and probability of error [24]. We assume a simple class of communication link designs as shown in Fig. 8.5. The figure shows the wireless link from a sensor to a controller. We assume the same link model for all the wireless links including the links from controllers to the actuators.

Each transmitter is assigned a unique ID number and this ID number is attached to the data (sensor measurement or control command) that needs to be sent. Assume we have M transmitters, then each ID number consists of $\lceil \log_2 M \rceil$ bits. At the transmitter, the data is first quantized and converted into a binary bit stream via a uniform quantizer. The bit stream, piggy-backed by the sender ID number, goes through the channel encoder that uses BCH codes for error correction and then a CRC (Cyclic Redundancy Check) for error detection. The effect of undetected errors can be disastrous in control applications since the actuator will use an erroneous control command as the correct one. We use a 16-bit CRC for which the probability of undetected errors is roughly 2^{-16}, which is less than 0.01%. We thus ignore the effects of undecoded errors since the probability is negligible. We use either BPSK or QPSK modulation at the transmitter. At the receiver, we assume coherent detection of the PSK signals. The BCH decoder can correct some transmission errors depending on the number of error correcting bits. After error correction, the receiver performs the CRC checksum for error detection. When no error is detected, the receiver sends an ACK back to the transmitter. If the transmitter receives the ACK, it clears its transmit buffer and does not transmit until a new packet arrives. Note that retransmissions are allowed if there are extra time slots. We assume that the transmit buffer only has a capacity of one data packet. Thus a packet will be discarded[5] if it has not been successfully received by the end of the sample period. Therefore, if a packet is successfully received, the packet delay is bounded by one sample period.

From the control perspective, the relevant communication parameters are data rate, time delay and probability of packet loss. Thus we can simplify the link model as in Fig. 8.6. This simplified model is sufficient to calculate all the communication

[5] In a control system, a new measurement is always more valuable than old measurements. Each transmitter only needs to send the newest data available.

parameters that may affect the control performance. The covariance of the quantization noise $\mathbf{v}_{q,i}$ is a function of the number of data bits representing the signal, and thus depends on the data rate. Both the time delay distribution and the probability of packet loss are determined by the MAC protocols, total number of retransmissions, and probability of successful transmission p_s. The probability of successful transmission p_s for each packet can be easily calculated given the link design, wireless channel gain and transmit power.

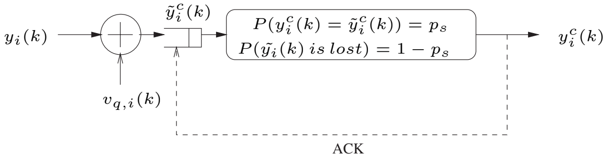

$y_i(k)$
$\tilde{y}_i^c(k)$
$P(y_i^c(k) = \tilde{y}_i^c(k)) = p_s$
$P(\tilde{y}_i(k)\ is\ lost) = 1 - p_s$
$y_i^c(k)$
$v_{q,i}(k)$
ACK

Fig. 8.6. Simplified model of the communication link.

8.4.3 MAC protocols

A common transmission scheduling protocol is TDMA (Time Division Multiple Access). TDMA is a collision-free protocol in which time slots are assigned in advance and never changed. We consider fixed TDMA and assume that time slots are divided evenly among all the transmitter/receiver pairs. Since the time slots are pre-assigned, a time slot can be wasted if the pre-assigned transmitter no longer has a packet to send.

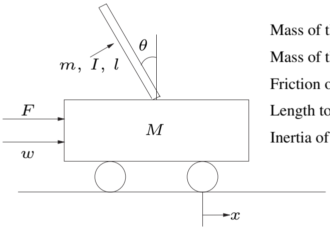
We also consider a contention based protocol Random Access (RA) with ACK. The ACK is a small packet that is sent back to the transmitter upon a successful transmission. We assume no spatial reuse and any two simultaneous transmissions will collide and cause packet losses.[6] With RA, each transmitter attempts to grab the channel independently with a probability of p at any given time slot. With ACK, the transmitter does not send redundant packets for the information that is already successfully decoded.

8.5 Numerical Example

The cart with an inverted pendulum, shown in Fig. 8.7, is controlled with a force, F, to cancel the random disturbance w and maintain the pendulum in an upright position. We use x to denote the cart position coordinate and θ as the pendulum angle from vertical.

For this example, we assume two identical inverted pendulum plants with the parameter choices as listed in Fig. 8.7. The state of the system is chosen as $[x(t), \dot{x}(t),$

[6] In practice the receiver may be able to decode one of the messages even with interference from another. This capability is called the capture effect [25].



Mass of the cart: $M = 0.5$ kg
Mass of the pendulum $m = 0.2$ kg
Friction of the cart $b = 0.1$ N/m/sec
Length to pendulum center of mass $l = 0.3$ m
Inertia of the pendulum $I = 0.006$ kg $*$ m^2

Fig. 8.7. Inverted pendulum and cart.

$\theta(t), \dot{\theta}(t)]$. The system dynamics are not linear in θ. We assume the pendulum does not move more than a few degrees away from the vertical and linearize the system dynamics about $\theta = 0$. We can thus get the standard linear model for the inverted pendulum. We would like to minimize the linear quadratic cost function with $Q_x^{<1>}(1,1) = 1.5, Q_x^{<1>}(3,3) = 2.5, Q_u^{<1>} = 1$, and $Q_x^{<2>}(1,1) = 1, Q_x^{<2>}(3,3) = 100, Q_u^{<2>} = 1$, where all the unspecified elements in $Q_x^{<i>}$ are zero. These weight matrices $Q_x^{<1>}, Q_x^{<2>}$ and $Q_u^{<1>}, Q_u^{<2>}$ are chosen to reflect the different priorities of different systems and different signals. In this example, the second system weighs more on θ since its main goal is to keep θ small while the first system gives roughly equal emphasis on x and θ. The measurement noise $\mathbf{v_s}(k)$ is assumed to be Gaussian with zero mean and covariance matrix $R = [10^{-4}, 0; 0, 10^{-6}]$.

8.5.1 Link layer resource allocation and design tradeoffs

We first show how different link designs affect the control performance. We assume TDMA in which the parameters for each system are transmitted sequentially as illustrated in Fig. 8.3 earlier in the chapter. Since we have 6 transmitter/receiver pairs, the ID field for each packet is 3 bits. We also require a minimum of 4 bits to represent each measurement/control command and approximate the quantization noise as a Gaussian random variable. We use two modulation schemes: BPSK and QPSK. QPSK provides twice the data rate of BPSK but QPSK incurs a larger probability of bit error for a fixed transmission power and bandwidth. We consider three different frame sizes: 24 bits, 32 bits and 48 bits. Note that there is a 16 bit CRC in each frame and we also use one bit guard time between each transmission. We use BCH codes for error correction. With a 32-bit frame, we can use (15, 11) and (15, 7) codes, where the first number is the total number of coded bits in the codeword and the second is the number of information bits. The code rate is defined as the fraction of the number of message bits and the total number of bits in the code. With a 48-bit frame, we can use (31, 26), (31, 21), (31, 16) and (31, 11) codes. We also consider the cases where no error correction coding is used. We represent these cases by (7, 7), (15, 15) and (31, 31) for 24-bit, 32-bit and 48-bit frames, respectively. In this example, we assume 96 Ksymbols/sec, thus the data rate is 96 Kbps for BPSK and 192 Kbps for QPSK. The transmission power is 10 mW and the noise density is $\frac{N_0}{2} = 10^{-8}\ W/Hz$. We consider a static channel gain $g = 0.2$.

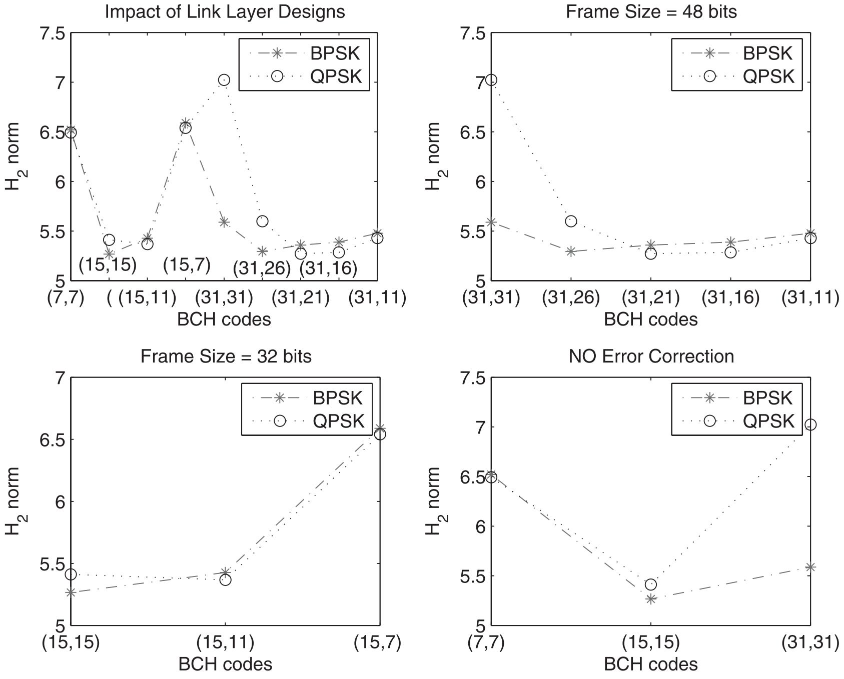


Fig. 8.8. TDMA with different link layer designs.

In Fig. 8.8, we plot the generalized H_2 norm against different BCH codes for both BPSK and QPSK. The first graph plots the performance of 18 different link designs. This is a little hard to read so we illustrate the results with 3 sub-plots. The second and third graph plot the performance under a fixed frame size of 48 bits and 32 bits respectively. For a given frame size, QPSK allows twice as many time slots as BPSK and the transmission time of each frame is only half of the BPSK transmission time due to the doubled data rate. However, QPSK incurs a higher probability of bit error, which in turn leads to higher probability of frame error. The probability of frame error can be reduced if a strong error correction code is used. As we see from the second plot, QPSK performs better when the code rate is low since more transmission errors can be corrected. Both the second and the third plot show that BPSK performs better than QPSK when the code rate is high. The last plot in the figure shows the performance comparison for different frame sizes when no error correction is used. Smaller frame sizes lead to more time slots and smaller probability of frame error when no error correction is used. Therefore, less delay and packet losses can be expected. However, the impact of data resolution kicks in. For small frame sizes, we have few bits to represent the signal. Thus the quantization noise is big. This is why (15, 15) outperforms (7,7) for BPSK and QPSK. It is surprising that the overall best performer is BPSK with 32-bit frames and no error correction coding, even though several other link designs perform only slightly worse. The reason is that this link design achieves the best tradeoff among the data resolution, time delay, and probability of packet loss in our control performance metric.

8.5.2 Cross layer design

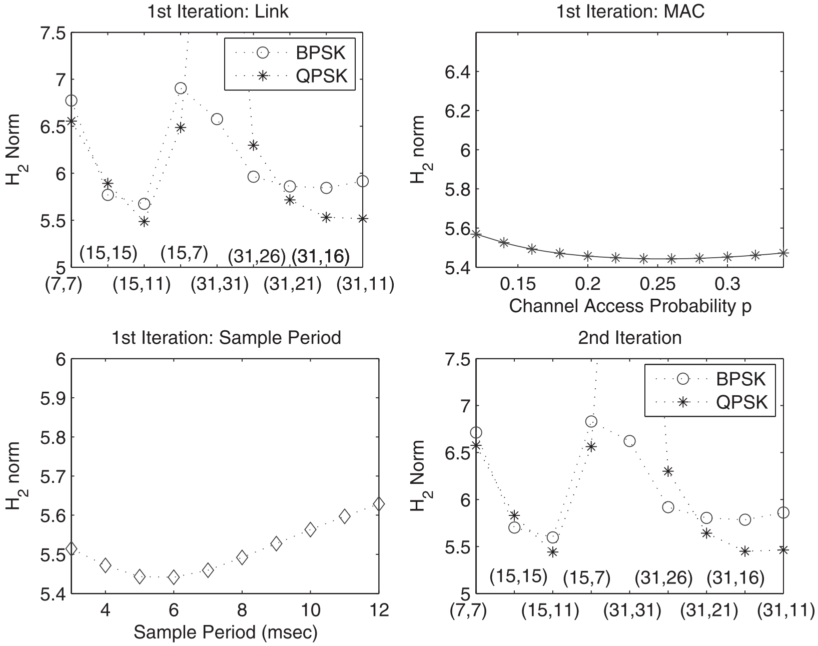


Fig. 8.9. Cross layer design: the link, MAC and application layer.

With Fig. 8.9, we illustrate the procedure of an iterative cross-layer design for the link layer, the MAC protocol, and the sample period. Our link design parameters are the modulation scheme, the frame size, and the error correction coding. We use RA with ACK as the MAC protocol and we vary the channel access probability p in order to optimize the performance. We assume all senders have an equal channel access probability and use the same link design for simplicity. The application layer parameter is the sample period. We start with an initial sample period of 5 msec and the channel access probability $p = \frac{1}{6}$. The first plot in Fig. 8.9 shows the control performance as a function of different link designs. The optimal link design is QPSK and $(15, 11)$ BCH codes in 32-bit frames. We keep this link design and optimize the MAC protocol within the class of RA with ACK for the initial sample period of 5 msec. The second plot shows the H_2 norm as the function of the channel access probability in RA with ACK. The control performance first improves (H_2 norm decreases) and then degrades as the channel access probability increases. When the senders access the channel with small probabilities, the channel is mostly idle but senders are not transmitting at rates they need to clear their buffers. Thus the packet delay is long. On the other hand, large access probabilities lead to collisions, which also cause packet delay and losses. The optimal choice is $p = 0.26$. The third step is to choose the optimal sample period now that we have updated our choice of both the MAC protocol and the link design. The third plot compares the system performance as the sample period varies. The optimal sample period is $T = 6$ msec. Then we go back to the first step. The fourth plot shows the control performance versus different link designs for $T = 6$ and $p = 0.26$. Again, QPSK with (15,11) is the best link design. The algorithm converges in the next step when $p = 0.26$ is again optimal within the class of RA with ACK protocols. Even though the H_2 norm of the control system would have been as large as 11.3 if we had chosen QPSK with 48-bit frame and no error correction coding, the performance gain in this case is not significant. Any sensible choice of the link design gives a reasonable performance. This is quite different from what we have studied in our previous work [6], where a different controller is used. The cross-layer design result in [6] is shown in Fig. 8.10.

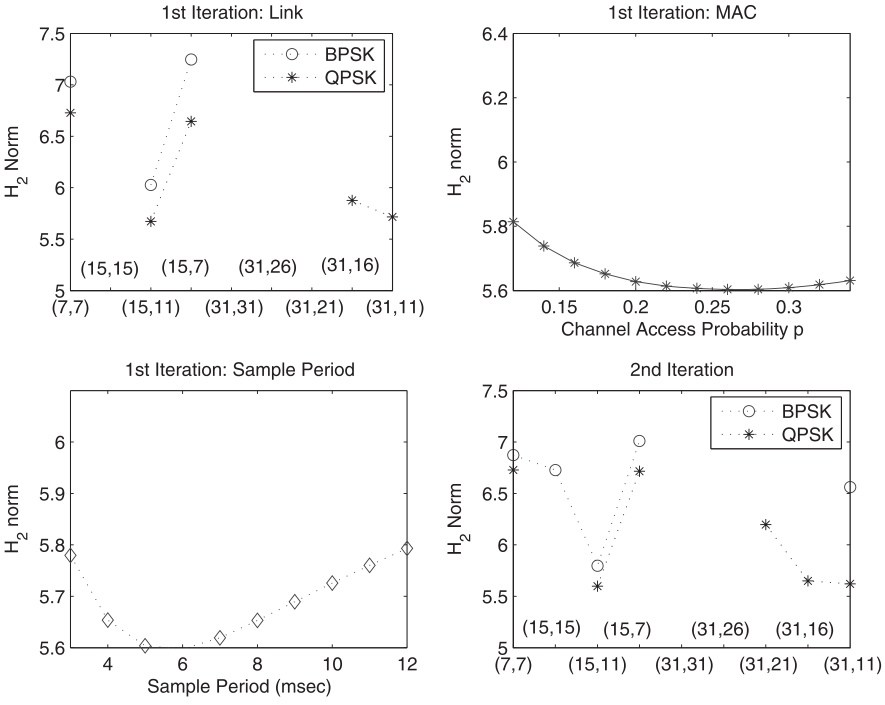


Fig. 8.10. Cross layer design: the link, MAC and application layer.

There are 18 data points on the first graph in Fig. 8.10 but only 8 are visible. That is because the other 10 link designs lead to system instability and thus the H_2 norm is infinite. In this example, all links with a 48-bit frame size when BPSK is used make the system unstable. This is because collisions lead to a large probability of packet loss if the number of retransmissions is small, even with very reliable links. The performance gain with this controller design is more dramatic. If we had designed these layers separately, we could have chosen a reliable link design with BPSK, 48-bit frames and strong error correction coding, which leads to system instability.

The controller discussed in this chapter gives a strict performance improvement for all the network design choices. For example, we compare the performance of the two control algorithms with a few link designs in the table below. The sample period is fixed at $T = 5$ msec. We see that the near optimal controller design discussed in this chapter performs strictly better (smaller H_2 norm) than the heuristic controller design in [6].

Codebook	Modulation	Heuristic	Near Optimal
(7, 7)	BPSK	7.0302	6.7747
(15, 15)	BPSK	∞	5.7703
(31, 16)	BPSK	∞	5.8433
(31, 11)	BPSK	∞	5.9155
(31, 16)	QPSK	5.8761	5.5315
(31, 11)	QPSK	5.7166	5.5169

The difference in the performance gain of the cross-layer design may be explained by the different controller choices. In Fig. 8.9, we use a near-optimal controller design, which adapts to the random delays and packet losses. The Kalman filter gives a near optimal state estimate which adapts to the sensor measurement losses. The state feedback controller adapts to random delays and packet losses in the control command optimally. In Fig. 8.10, the controller design is a heuristic and is not adaptive to partial measurement losses at the state estimator. The improvement in the control design gives a much broader stability region in terms of the network throughput. Yet the optimal controller design seems to take away some of the potential gains of the cross-layer network optimization. Of course, our cross-layer design is only optimizing a small set of network variables and we restrict all the links to have the same design. A full-scale cross-layer optimization is expected to give much more significant performance gains for the optimal controller design discussed in this chapter.

8.6 Conclusions and Discussion

We propose a cross-layer framework for joint design of distributed control and wireless networks. The network design goal is to optimize the control performance, which is an implicit function of the network performance. Similarly, control design choices impact network performance, which in turn impacts controller performance.

Thus a joint design of control and communication is necessary. Cross-layer design provides a broad framework where each layer of the network protocol stack, including the controller design, can be optimized relative to the end-to-end performance.

The goal of cross-layer design is to provide the best end-to-end performance of the application. In a distributed control system, the control system is the application of the network. Thus cross-layer design also includes designing control algorithms that are adaptive and robust to the network performance. We extend the separation principle for controllers with packet losses and find the optimal LQG controller to be composed of a Kalman filter and a delay dependent state feedback controller. The Kalman filter is an extended version of the classical Kalman filter since the observations sent to the Kalman filter can be lost during transmission.

We then consider an iterative cross-layer design over a subset of the network layers with an optimal controller as the application layer. Such an iterative design can give substantial performance gains for certain controller designs. We also uncover some surprising insights. In particular, we show that an uncoded link design, which is often undesirable due to its unreliability, can be optimal under certain circumstances since it achieves the optimal tradeoff among data resolution, time delay and packet loss probability. Note that an iterative design is only suboptimal. A true joint design over all the network layers should give more significant performance gains.

An intriguing question to ask is if there is separation between the control design and the communication design. The control performance index is a complicated function of both the control design and the communication design. Determining the optimality in the separate designs is therefore a challenging problem.

The problem becomes even more difficult when we have a fast fading channel, where the performance provided by the wireless links is time varying. In a fast fading environment, the probabilistic performance provided by the network is no longer stationary. There is a lack of theory in evaluating and designing such systems. A network control system on the move, such as an Automated Highway System, needs to take the time-varying channel into account. Adaptive link layer techniques are commonly used to compensate for the links. In a cross-layer design, the MAC protocol, the routing algorithm, and the controller designs must adapt to the channel states as well. This area is just beginning to be explored. An important question to ask in this adaptive cross-layer design is what parameters shall be shared among different layers of the network and how each layer can be made robust to changing network conditions.

References

[1] P. Varaiya, Smart cars on smart roads: Problems of control, *IEEE Trans. on Autom. Control,* **38**:195–207, Feb. 1993.

[2] L. Xiao, M. Johansson, H. Hindi, S. Boyd and A. Goldsmith, Joint optimization of communication rates and linear systems, *IEEE Trans. on Autom. Control,* **48**:148–153, Jan. 2003.

[3] X. Liu and A. Goldsmith, Kalman filtering with partial observation losses, Manuscript, Stanford University, 2004.

[4] X. Liu and A. Goldsmith, Wireless communication tradeoffs in distributed control, *Proc. IEEE Conf. on Decision and Control,* 688–694, Dec. 2003.

[5] X. Liu and A. Goldsmith, Wireless medium access control in networked control systems, *Proc. IEEE American Control Conference,* June 2004.

[6] X. Liu and A. Goldsmith, Wireless network design for distributed control, *Proc. IEEE Conf. on Decision and Control,* Dec. 2004.

[7] M.S. Branicky, S.M. Phillips and W. Zhang, Scheduling and feedback co-design for networked control systems, *Proc. IEEE Conf. on Decision and Control,* 1211–1217, Dec. 2002.

[8] F.-L. Lian, J. Moyne and D. Tilbury, Network design consideration for distributed control systems, *IEEE Trans. on Control Systems Technology,* **10**:297–307, Mar. 2002.

[9] F.-L. Lian, J. Yook, P. Otanez, D. Tilbury and J. Moyne, Design of sampling and transmission rate for achieving control and communication performance in networked agent systems, *Proc. American Control Conference,* 3329–3334, June 2003.

[10] G.C. Walsh, O. Beldiman and L. Bushnell, Error encoding algorithms for networked control systems, *Proc. IEEE Conf. on Decision and Control,* 4933–4938, Dec. 1999.

[11] T. Yoo, E. Setton, X. Zhu, A. Goldsmith and B. Girod, Cross-layer design for video streaming over wireless ad hoc networks, *Proc. IEEE International Workshop on Multimedia Signal Processing,* Siena, Italy, 179–182, Sept. 2004.

[12] S. Cui, R. Madan, S. Lall and A.J. Goldsmith, Cross-layer optimization in TDMA-based sensor networks with energy constraints, Manuscript, Stanford University, 2004.

[13] S. Shakkottai, T.S. Rappaport and P.C. Karlsson, Cross-layer design for wireless networks, *IEEE Commun. Mag.,* **41**:74–80, Oct. 2003.

[14] T. ElBatt, A. Ephremides, Joint scheduling and power control for wireless ad hoc networks, *IEEE Trans. Wireless Communications,* **3**:74–85, Jan. 2004.

[15] T. Lang, Q. Zhao and G. Mergen, Multipacket reception in random access wireless networks: From signal processing to optimal medium access control, *IEEE Commun. Mag.,* **39**:108–112, Nov. 2001.

[16] T. Chen and B. Francis, *Optimal Sampled-Data Control Systems*, Springer, 1995.

[17] H. Witsenhausen, A counterexample in stochastic optimum control, *SIAM Journal on Control,* **6**:131–147, 1968.

[18] B. Azimi-Sadjadi, Stability of networked control systems in the presence of packet losses, *Proc. IEEE Conf. on Decision and Control,* 676–681, Dec. 2003.

[19] V. Gupta, D. Spanos, B. Hassibi and R.M. Murray, On LQG control across a stochastic packet-dropping link, Manuscript, California Institute of Technology, 2004.

[20] J. Nilsson, Real-Time Control Systems with Delays, Ph.D. Thesis, Lund Institute of Technology, 1998.

[21] Y. Ji, H.J. Chizeck, X. Feng and K.A. Loparo. Stability and control of discrete-time jump linear systems, *Control Theory and Advanced Technology,* **7**:247–270, 1991.

[22] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M.I. Jordan and S.S. Sastry, Kalman filtering with intermittent observations, *IEEE Transactions on Automatic Control*, **49**:1453–1464, Sept. 2004.

[23] R. Horn and C. Johnson, *Matrix Analysis*, Cambridge University Press, 1990.

[24] J. Proakis, *Digital Communications*, McGraw-Hill, 1995.

[25] T. Rappaport, *Wireless Communications: Principles and Practice*, Prentice Hall, 1996.

9

Network Pricing for QoS: A ‘Regulation’ Approach*

Dinesh Garg[1], Vivek S. Borkar[2], and D. Manjunath[3]

[1] Department of Computer Science and Automation
Indian Institute of Science, Bangalore, 560 012, India
dgarg@csa.iisc.ernet.in

[2] School of Technology and Computer Science
Tata Institute of Fundamental Research, Mumbai, 400 005, India
borkar@tifr.res.in

[3] Department of Electrical Engineering
Indian Institute of Technology, Mumbai, 400 076, India
dmanju@ee.iitb.ac.in

9.1 Background

Increasing stream traffic on the Internet and higher performance expectations from the users of services built on the elastic traffic model have meant that the network should provide more predictable performance in terms of delays and loss of packets. Further, with increasing variety in applications and in users, the network should also provide different grades of service. Since the supply of network resources, e.g., buffer and bandwidth, is fixed over small timescales, predictability of performance requires that the network be capable of controlling the behavior of the demand for these network resources, especially during times of congestion when the instantaneous demand exceeds supply. Allocating network resources among competing demands during periods of congestion is essentially a conflict resolution problem. In this chapter we consider pricing of network resource usage to resolve this. Appropriate usage and congestion based pricing will provide incentives that influence the users’ behavior and their price sensitivity will elicit their true requirement.

9.2 Approaches to Network Pricing

Pricing in multi-service packet networks attracted very little research till the 1990s, possibly because of a lack of economic background in the networking community [54]. An excellent introduction to Network Economics is available in [54],

* V. S. Borkar’s work was supported in part by grant number III.5(157)/99-ET from the Department of Science and Technology, Government of India. D. Manjunath’s work was supported in part by grant number 1(1)/2004-E-Infra from the Ministry of Information Technology, Government of India.

possibly the only networking textbook to devote a chapter for this purpose. The INDEX (INternet DEmand EXperiment) [2, 13] experiment was a pioneering attempt to study the controlling of Internet demand through congestion pricing. This is also possibly the only experimental study of pricing Internet service.

A simple approach to network pricing for QoS would be to provide different types of service by having a separate queue for each service type and service them according to a discipline that provides the requisite QoS. The Paris Metro pricing (PMP) [41] is just such a scheme in which service in the different queues are priced differently. [41] suggests that each of the queues be serviced at the same rate much like in the Paris Metro of yore and the 'locals' of Mumbai today. Although no guarantees are offered, relative QoS is provided in the high-priced part of the network because high prices tend to keep the average arrival rate lower and hence the traffic that chooses a high-price service will, on the average, experience less congestion than those that choose the low-priced service. [17] and [23] assess the viability of the PMP pricing scheme for the service provider. [11] describes an adaptive pricing scheme to maximize the network revenue and also analyzes the social optimality of the greedy policy in a PMP-like system that is called the 'Tirupati-queue'. [52, 53] obtain the stability criteria, the revenue rate and mean delays in the PMP and the Tirupati systems when the customers adopt a 'join minimum cost queue' policy and the prices are static. [43] considers the problem of pricing service classes to force users to choose the nominal grade of service that they are assigned.

Different grades of QoS can also be provided through a priority mechanism. [9, 18] describe a priority-based pricing mechanism. Here the users indicate the value of their traffic by selecting a priority level and during congestion periods, low priority traffic is delayed or even dropped. The impact of priority pricing on QoS for a typical user and congestion is studied in [18, 19].

See [14] and [10] for a nice survey on some early pricing schemes for QoS-enabled networks.

Much of the rest of the network pricing literature is aimed at developing a theory for network pricing which has been addressed from many perspectives. In the rest of this section we first provide an overview of two of these approaches—(1) welfare-maximization based pricing and (2) game-theoretic schemes. The network and the users define utility functions on the price of service and the congestion experienced by the service. In the welfare-maximization schemes the users adjust their resource usages and the network adjusts the prices to maximize their respective utility functions. In the latter approach, the objective is to converge towards an equilibrium that is a good operating point for the network. We also provide an overview of the regulation network pricing that we propose in this chapter.

9.2.1 Welfare maximization methods

The network and the users are assumed to have well defined utility functions that characterize their resource requirements and their willingness to pay per unit resource. Typically, the network is assumed to not know the individual utility functions of the users. The QoS received by the user from the network is assumed to be built into the resource usage and hence the utility function, like for example, by translating the desired QoS into an effective bandwidth requirement. The goal of the network is to allocate resources to maximize a network objective function that depends on the users’ utilities, e.g., the total utility over all the users which is also called ‘social welfare’. To achieve this goal, the network uses pricing as a means to obtain implicit information about the users’ utility functions and to allocate resources accordingly. A typical scheme of this kind would be of *tâtonnement* type where the optimization problem of the user is the dual of that of the network.

A considerable amount of literature based on this approach is devoted to rate control where the network adjusts the price per unit bandwidth along the routes according to the demand and the users respond by adjusting their transmission rates to optimize their utility function. One of the earliest works of this type is [38]. Users are differentiated based on their traffic characteristics and the maximum end-to-end delay they are willing to experience. Users purchase bandwidth and buffers from the network to satisfy their QoS requirements. The network and the users iteratively adjust respectively their prices and demand to converge to an optimum allocation among the users. A ‘proportionally fair’ bandwidth allocation in which the network resource allocation is proportional to the users willingness to pay is proposed in [25]. [26] proposes a simple rate control mechanism where the network adjusts the price per unit bandwidth on the routes and the users choose a transmission rate to maximize their utility function. It is shown that in equilibrium, the resulting scheme maximizes the social welfare and achieves a proportionally fair allocation of the network resources. In [25, 26], the overall objective is decomposed into separate optimization subproblems for the network and for the users, where each user chooses a willingness to pay and the network allocates rates to these sources in a way that is proportionally fair. Thus in this approach, the users decide their payment and receive what the network allocates. In [36] the users decide their resource usage and pay what the network charges.

Welfare maximization mechanisms for fair allocation of resources have also been applied to window based congestion control algorithms. [31, 33] describes a family of window-based allocation schemes that can be used to achieve a proportionally fair allocation or approximate a max-min fair allocation arbitrarily closely. This scheme was motivated by TCP, the Internet’s congestion control protocol. [16] describes another marking scheme for TCP packets to achieve proportional fairness. See [37] for a recent survey on the design of fair congestion control protocols for the Internet.

In an interesting study, [45, 46] propose a dynamic programming formulation of the revenue and welfare maximization problem and show that the performance of an optimal dynamic pricing strategy is closely matched by a suitably chosen static price, which does not depend on instantaneous congestion. Other works in this genre include a study about the dynamics of congestion pricing in [15]. Some of the theory developed for wireline networks is now being applied to transmission scheduling in wireless networks, e.g., [40, 47].

9.2.2 Game theoretic approach

There are many game theoretic formulations for rate allocation and pricing of bandwidth usage. Much of the work in this direction considers routing of the traffic and derives the conditions for the existence and uniqueness of an equilibrium [1, 24, 27, 30, 32, 42]. The existence of equilibria allows the design of network management policies that induce efficient equilibria by penalizing the users who deviate from the equilibrium. In a recent work [28], QoS routing is considered where an arriving connection finds the route that minimizes the total cost while satisfying its QoS constraints, e.g., maximum end-to-end delay. The cost of reserving a unit rate over a link is a function of the aggregate rate reserved on the link. The interaction among the various connections is modeled as a game and the goal of the network provider is to decide link prices in such a way that the operating point coincides with the Nash equilibrium of the underlying game. [55] considers game theoretic pricing for rate control and shows that the proportionally fair allocation is a Nash bargaining solution. In [3], the interaction between the service provider and the users is modeled as a Stackelberg game where users are the followers and the service provider is the leader whose goal is to maximize revenue.

[39] models a priority based network as a non-cooperative game between users and the network and shows that a unique equilibrium exists for this game and that the bandwidth allocation in equilibrium is weighted max-min fair. [20] considers pricing by multiple service providers as a non-cooperative Nash game and shows that the equilibrium pricing may be unfair and inefficient.

9.2.3 Other work

Providing QoS guarantees to competing traffic classes using a methodology based on economic models is considered in [48]. There is also a significant amount of work on auction based methods. Auction mechanisms have been suggested in [34]. More recently, [49] describes a smart market based mechanism for pricing in a Diff-Serv based network in which the users bid for the service and the network assigns the quality to the users based on their bids.

9.2.4 ‘Regulation’ based pricing: a preview

In this chapter we consider a different approach to pricing of network services to provide QoS to users. We propose a simple dynamic pricing scheme for differentiated service that is based on a ‘regulation’ viewpoint. In this scheme we assume that a nominal (‘ideal’) profile for the resource utilization in each of the different grades of service is specified *a priori*. We dynamically adjust the prices for the different grades of service so as to modulate user behavior in a manner that drives resource utilization towards this nominal profile. Service grade guarantees are provided through the nominal profile prescribed for each grade of service. This scheme was originally proposed for a single service station in [7]. We now extend it to the network case.

Each link in the network offers multiple grades of service, much like the per hop behavior of DiffServ [5]. A set of routes are defined on the network. A route r is defined as a sequence of n_r 2-tuples of the form (i, j) where i is a link on the route and j is the service grade for that route on link i. As mentioned above, the nominal congestion levels in our scheme are prescribed *a priori* for each of the routes. These can be arrived at as an offline static optimization problem (two possibilities are mentioned later) or otherwise. Arriving packets are allowed to choose any of the routes that are available. The choice of a route implicitly chooses the grade of service on each link.

We can see that our pricing scheme is inherently a *secondary* pricing scheme, i.e., it is a scheme for stabilizing the desired resource utilization variables around values separately arrived at through another *primary* scheme. This is in the spirit of regulation problems in optimal control where one tries to control a system so as to make it track a predetermined trajectory.

The scheme supports multiclass traffic with each class having its own utility function for choosing the route, and hence the service type, based on the grade of service, current congestion levels and the prices and/or any other criteria. This function could take any form (but must satisfy some properties to be enumerated later) and the network does not have to know this utility function. As mentioned above, the nominal congestion levels in our scheme are prescribed *a priori*.

We will argue later that a link state protocol can be used to exchange the congestion and pricing information required by the pricing scheme. We will also see that the 'regulation' based pricing scheme has the following desirable characteristics.

- It provides well defined guarantees, albeit statistical, on the grades of service.
- It supports congestion control and traffic management.
- It is economically efficient and elicits the true behavior of users.
- It is simple to implement, requires minimal measurement, and is compatible with current technologies and proposed standards.

This work is admittedly at a 'proof of concept' level and will need some refinement before it can be converted into a realistic scheme for large networks. We will discuss some of these issues.

The rest of the chapter is organized as follows. Section 9.3 describes the network and the user model. In Section 9.4 we describe the pricing scheme and present its analysis. We also describe some variations on the basic scheme that are expected to improve the system performance. Section 9.5 describes some of the simulation results. Section 9.6 discusses issues in the implementation of the scheme where we describe methods to calculate the operating point and also to communicate the congestion information. We conclude in Section 9.7.

9.3 Network and User Model

We consider a packet communication network of N links with link i having a transmission capacity of μ_i and providing J_i grades of service to the packets that are to be transmitted on the link. A separate logical queue may be maintained for packets corresponding to each grade of service. Define a route r as a sequence of n_r links and the grade of service on each of these links, i.e., route $r := [(i_1, j_1), (i_2, j_2), \ldots, (i_{n_r}, j_{n_r})]$ where $i_1, \ldots, i_{n_r}$ are the links used by route r and $j_1, \ldots, j_{n_r}$ are the service grades used on the respective links. Let $\mathcal{R}$ denote the set of routes with $|\mathcal{R}| = K$.

The J_i queues of link i can be serviced according to an arbitrary policy that is designed to provide the required QoS, e.g., strict round robin, weighted fair queueing (WFQ) or any of their variants.

Let $y_i(t) := [y_{i,1}(t), \ldots, y_{i,J_i}(t)]$, $y_{i,j}(t) \in \Re$ denote the state of link i at time t with increasing $y_{i,j}(t)$ denoting increasing demand (congestion or utilization) for service grade j of link i. Let $y(t) := [y_1(t), y_2(t), \ldots, y_n(t)]$ be the state of the network at time t. Although we allow $y(t)$ to be any real vector that will be defined by the policy in which the multi-class queues are served, it is logical to let $y_{i,j}(t)$ be either buffer occupancy (expressed as the number of bits or number of packets) or the total arrival rate into the class. We assume the existence of a 'natural' upper bound $b_{i,j}$ on $y_{i,j}(t)$, $j = 1, 2, \ldots, J_i$, $i = 1, 2, \ldots, N$ (e.g., the buffer size if $y_{i,j}(t)$ is the queue length, or the capacity allocated to grade j on link i if $y_{i,j}(t)$ is utilization). In the sequel, we assume $y_{i,j}(t)$ to be the buffer occupancy and describe the model around this assumption. The description can be easily modified to suit other measures of congestion.

We assume that the congestion contribution of the routes using service grade j on link i combine in some way to yield $y_{i,j}(t)$. Let $z^r_{i,j}(t)$ be the contribution of route r towards the congestion in service grade j of link i at time t. Then, for $t > 0$, $j = 1, \ldots, J_i$ and $i = 1, \ldots, N$,

$$b_{i,j} \geq y_{i,j}(t) = \phi_{i,j}\left(z^1_{i,j}(t), \ldots, z^K_{i,j}(t)\right). \tag{9.1}$$

$y_{i,j}(t)$ will be independent of $z^r_{i,j}$ if service grade j is not used by route r on link i. Of course, $\phi(\cdot)$ has to be increasing in $z^r_{i,j}$. In the simplest model we could assume that contributions are additive and have $y_{i,j}(t) = \sum_{r \in \mathcal{R}_{i,j}} z^r_{i,j}(t)$ where $\mathcal{R}_{i,j}$ is the set of routes using service grade j on link i.

Let $z^r(t) \in \Re$ denote the congestion on route r at time t with increasing $z^r(t)$ denoting increasing congestion, and hence demand. We define $z^r(t)$ as a function of its contribution on the links of the route, i.e.,

$$z^r(t) = \psi_r\left(z^r_{i_1,j_1}(t), \ldots, z^r_{i_{n_r},j_{n_r}}(t)\right). \tag{9.2}$$

$\psi_r(\cdot)$ could be defined so as to prescribe the average of the total number of packets in the network ($z^r(t) = z^r_{i_1,j_1}(t) + \cdots + z^r_{i_{n_r},j_{n_r}}(t)$), keep the maximum of the congestion on the links due to this route at a prescribed level ($z^r(t) = \max\{z^r_{i_1,j_1}(t), \ldots, z^r_{i_{n_r},j_{n_r}}(t)\}$) or maintain a minimum average congestion on the links on the route ($z^r(t) = \min\{z^r_{i_1,j_1}(t), \ldots, z^r_{i_{n_r},j_{n_r}}(t)\}$).

Let $z(t) = [z^1(t), z^2(t), \ldots, z^K(t)]$ be a vector representing the state of all the routes of the network at time t.

To be able to provide a required level of QoS, the network service provider fixes an operating point y^* for the service classes and an operating point z^* for the routes, where

$$\begin{aligned} y_i^* &= [y_{i,1}^*, \ldots, y_{i,J_i}^*], \\ y^* &= [y_1^*, \ldots, y_n^*], \\ z^* &= [z^{1*}, z^{2*}, \ldots, z^{K*}]. \end{aligned} \tag{9.3}$$

The price per unit traffic on route r at time t is denoted by $p^r(t)$. Define $p(t) := [p^1(t), \ldots, p^K(t)]$ to be the network price vector at time t. We assume that $p(t)$ is posted by the service provider and is available at the users. $p(t)$ depends on $z(t)$ and it is adapted so as to make the congestion level on the routes around the desired operating point z^*. y^* does not directly enter our adaptation scheme but enters some of the mechanisms that we suggest for choosing z^*. It can also be used by the traffic in defining their utility functions.

We next describe the model for the user process. As we have mentioned earlier we allow multiclass traffic. Users of class s are assumed to incur a cost of $C_s^r(x, z^r(t), p^r(t))$ when it injects x units of traffic into route r at time t and this cost is increasing in $z^r(t)$ and $p^r(t)$. An example cost function can be of the form $C_s^r(x, z^r(t), p^r(t)) = xp^r(t) - U_s^r(x, z^r(t))$ where $U_s(x, z^r(t))$ is the utility of sending x units of traffic on route r when the congestion level is $z^r(t)$.

Following [7] two user models can be defined—a *small user* model and a *large user* model. Consider a user of class s that needs to send $x_s(t)$ units of traffic at time t. Let $\mathcal{R}_s \subset \mathcal{R}$ be the set of routes that class s can use. In the *small user model* the user will assign all the x_s units to the single route from among those that it can use that minimizes its cost. For the example cost function defined above, the user will allocate its traffic to the route r obtained as

$$\arg\min_{r \in \mathcal{R}_s} \left\{ p^r(t) x_s(t) - U_s^r(x_s(t), z^r(t)) \right\}.$$

In the *large user model* the source will partition $x_s(t)$ among usable routes so as to minimize its cost. For the cost function that we give above, a large user with $x_s(t)$ units of traffic will send $x_s^r(t)$ to route r, so as to maximize

$$\sum_{r \in \mathcal{R}_s} \left\{ U_s^r(x_s^r(t), z^r(t)) - p^r(t) x_s^r(t) \right\}$$

subject to $\sum_{r \in \mathcal{R}_s} x_s^r(t) = x_s(t)$. Here, we have assumed that utility functions are additive over the routes.

9.4 The Price Adaptation

The aim of the price adaptation scheme is to adapt the route price so as to keep the traffic on the route close to the prescribed nominal profile z^*. As can be seen from our description of the users, we assume that the users behave in an individually optimal manner with respect to their respective utility functions and choose the route that minimizes their cost (or conversely maximizes their utility). The 'price setter' for route r uses z^{r*} and $z^r(t)$ and adjusts the price to achieve the regulation objective. The price per unit volume of the traffic on route r and the congestion on it, $p^r(t)$ and $z^r(t)$ respectively, are communicated to the user population with possibly non-zero delays.

A simplistic way of regulating the congestion on route r to z^{r*} would be to use zero price when $z^r(t) < z^{r*}$ and an extremely high price when $z^r(t) > z^{r*}$. This clearly leads to high fluctuations in the arrival process to the routes and can cause severe packet losses. To ensure a graceful adaptation of the traffic process into the routes, we use the following price adaptation equation. Let $t_1, t_2, t_3, \ldots$ be the epochs at which the price is adapted.

$$p^r(t_{i+1}) = \mathit{\Gamma}\left(p^r(t_i) + ap^r(t_i)(z^r(t_{i+1}) - z^{r^*})\right) \;\forall\, r \in \mathcal{R}. \tag{9.4}$$

Here $a > 0$ is a small scalar called the 'learning parameter' and $\mathit{\Gamma}(\cdot)$ is the projection onto the interval $[\eta, M]$, $\eta, M, > 0$, with η a small number and M a very large number. The projection $\mathit{\Gamma}(\cdot)$ puts an upper bound on the price for a route and also prevents the scheme from getting stuck at zero, i.e., $\mathit{\Gamma}(x) = \min(\max(x, \eta), M)$. The intuition behind the adaptation equation in (9.4) is simple—decrease the price if the congestion on a route falls below the nominal and vice versa.

We call Equation 9.4 the linear deviation scheme because $(z^r(t_{i+1}) - z^{r*})$ is the deviation of the current congestion from the nominal value and the adaptation equation is linear in the deviation.

9.4.1 Analysis

We sketch briefly the theoretical underpinnings of our scheme, along the lines of [7]. We assume the following: If the price vector was frozen at $p = [p_1, \ldots, p_K]$, then the queue length process in the routes on the queues, and hence the congestion on the routes, is asymptotically stationary ergodic. Let $E_p[\,\cdot\,]$ denote the corresponding stationary average. Define $h(\cdot) = [h_1(\cdot), \ldots, h_K(\cdot)]^T : (\Re^+)^K \to (\Re)^K$ by $h_r(p) := p_r(E_p[z_r(t)] - z_r^*)\ \forall r \in \mathcal{R}$. Consider the 'limiting o.d.e.' for our algorithm:

$$\dot{q}(t) = h(q(t)). \tag{9.5}$$

We shall need the following assumptions.

(A1) $h(\cdot)$ is continuously differentiable.
(A2) $h_r(p) > 0$ when $p_r = \eta$ and < 0 when $p_r = M$ for $1 \leq r \leq K$.
(A3) $\frac{\partial h_i}{\partial p_j} \geq 0$ for $i \neq j$ and $i, j = 1, \ldots, K$.

(A1) implies that (9.5) is locally well-posed. It will hold, e.g., when the stationary law of $\{z(t)\}$ depends smoothly on p when the latter is kept fixed, which is a mild requirement. The first half of (A2) means that when the price for a route is very low (to be precise, $= \eta$), it will attract traffic, in turn pushing the prices up. The second half means that high prices will correspondingly lead to low traffic into that route, hence lower congestion, low enough to warrant a drop in the prices. (A3) implies that an increase in the price of one route tends to drive arrivals to the others and increase the congestion levels there. Both make eminent economic sense and will hold, for example, when the users have a cost function that is increasing in prices and join the route that minimizes the cost, and either a cheap alternative is always available or balking occurs. The presence of the projection operator $\Gamma(\cdot)$ ensures that the iterates remain bounded.

Standard results from the ‘o.d.e.’ approach to stochastic approximation algorithms [29] imply that the algorithm in (9.4) will asymptotically track (9.5) in the following sense: Define $\tilde{p}(t), t \geq 0$, by

$$\tilde{p}(na) = p(n),\ n \geq 0,$$

with linear interpolation on each interval $[na, (n+1)a]$. Then for any $T > 0$,

$$\lim_{t_0 \to \infty} E[\sup_{t \in [t_0, t_0+T]} ||\tilde{p}(t) - \hat{p}_0(t)||^2] = O(a), \tag{9.6}$$

where $\hat{p}_0(\cdot)$ is a solution of (9.5) on $[t_0, \infty)$ with $\hat{p}_0(t_0) = \tilde{p}(t_0)$. See, e.g., [8]. Thus (9.4) is a ‘noisy discretization’ of (9.5) with $O(a)$ error. The projection $\Gamma(\cdot)$ would lead to a boundary correction in the o.d.e. limit if the driving vector field were directed outwards at any point in the boundary. This follows from the explicit expression for this correction term in, e.g., [29], Section 5.1. In our case such a correction term is missing in the o.d.e. limit, because the driving vector field of (9.5) will be directed inwards at the boundary of the region $[\eta, M]^K$, thanks to (A2). This last observation also implies in particular that the associated flow maps this region into itself and hence by the Brouwer fixed point theorem [44], has at least one fixed point, i.e., (9.5) has at least one equilibrium in this region. Absence of (A2) will allow boundary equilibria which is not a problem, except that they warrant a messier analysis involving boundary correction. These, however, may not be the *desired* equilibria, in the sense that they may not correspond to the target z^*. The idea here is that η, M, z^* should be arrived at from known traffic characteristics, so that (A2) is reasonable. If the input traffic rate is either too high or too low, it is no longer so. Mathematically, the assumptions (A1)–(A3) qualify (9.5) as a cooperative o.d.e. in the sense of [21, 50]. Thus we have:

Lemma 1. *(a) For generic initial conditions (i.e., for initial conditions belonging to an open dense set), $q(\cdot)$ converges to the set of equilibria H of (9.5) (though not necessarily to a single point thereof).*
(b) If the inequalities in (A3) above are strict, then for generic initial conditions, $q(\cdot)$ converges to one of the equilibria in H, depending on the initial condition.

These are minor variations of the results in the references above as observed in [7]. The boundedness of trajectories required in these results of [50] is ensured by (A2). For later reference, we denote by G the open dense set of initial conditions for which the conclusions of the lemma hold. The condition in *(b)* above can be relaxed to the requirement that the Jacobian matrix $Dh(x) \overset{\Delta}{=} [[\frac{\partial h_i}{\partial p_j}(p)]]_{1 \leq i,j \leq K}$ be irreducible at every point. Also, both (a) and (b) will mean point convergence if H is discrete. Note also that $H \neq \phi$ is a part of the conclusion of Lemma 1. We now seek conditions for H to be discrete. Consider the condition:

(B1) The Jacobian matrix $Dh(x)$ is non-singular at every point in H.

This condition is not too restrictive: Define $\tilde{h}(\cdot) := E_p(z(t))$. By Sard's theorem ([44], p. 130), $D\tilde{h}(p^*)$, and therefore $Dh(p^*)$, is non-singular for almost all choices of z^*, i.e., for all z^* outside a set of zero Lebesgue measure. (Some care has to be taken in applying this argument when natural constraints force the range of h to be strictly a subset of $\Re^K$, see [7].) By the inverse function theorem [51], the zeros of h, i.e., the equilibria of (9.5) are isolated under (B1). Structural stability considerations [22] suggest that it is reasonable to suppose that these equilibria are hyperbolic. Let S denote the (discrete) set of stable equilibria. Then the considerations above make the following a reasonable assumption:

(B2) There exists an open dense set U such that all trajectories of (9.5) initiated in U converge to some point in S.

Intuitively, we want U^c to include the initial conditions excluded by Lemma 1 *and* the equilibria not in S and their stable manifolds, i.e., U will be the union of the domains of attraction of the stable equilibria of the o.d.e. We also assume:

(B3) $\{p(t)\}$ is asymptotically stationary and the stationary law ν_a of $p(t)$ (where 'a' is the stepsize in (9.4)) has a density w.r.t. the Lebesgue measure on $\Re^K$.

The stationarity would be true, e.g., if the process $z(\cdot)$ were a function of a positive recurrent Markov chain, a common situation. The Lebesgue continuity of the stationary distribution is an assumption. If it were not true, one could enforce it by adding additional Lebesgue-continuous i.i.d. noise to the r.h.s. of our iteration. In our simulations, the intrinsic randomness of the system seemed to suffice. Let $B(\epsilon) := \{p \in \Re^K : \inf_{p^* \in S} ||p - p^*|| < \epsilon\}$ (i.e., the $\epsilon-$neighborhood of S) and set

$$\begin{aligned} T(x) &= \sup\{t \geq 0 : q(t) \notin B(\epsilon)\} \\ &= \inf\{t \geq 0 : q(s) \in B(\epsilon)\ \forall\ s > t\}, \end{aligned}$$

where $q(\cdot)$ satisfies (9.5) with initial condition x. $T(x)$ is the least time such that the trajectory starting at x lies in $B(\epsilon)$ thereafter. (Thus $T(x) < \infty$ for $x \in U$.) Conditions (B2), (B3) imply in particular that $\nu_a(\bigcup_n \{x : T(x) \leq n\}) = 1$. Thus given any $\delta > 0$, we can pick an $N(\delta) \geq 1$ such that $\nu_a(\{x : T(x) \leq N(\delta)\}) > 1 - \delta$. Consider a stationary process $\{p(t)\}$ governed by (9.4) and the corresponding equation (9.5) with $t_0 = 0$ and $T = N(\delta)a$. Then it is clear that

$$\nu_a(B(\epsilon)) = P(\tilde{p}(T) \in B(\epsilon + O(a))) > 1 - \delta. \tag{9.7}$$

We have proved (see also [4]):

Theorem 1. *The stationary distribution ν_a of (9.4) concentrates on the stable equilibria of (9.5) in the sense of (9.7).*

Note that *any* equilibrium point $\hat{p}$ of (9.5) corresponds to the desired equality $z^* = E_{\hat{p}}[z(n)]$, so they are all equivalent for our purposes.

Remark: As in [7], one has the following sufficient conditions for (9.5) to have a unique equilibrium which is asymptotically stable,:

$$\left|\frac{\partial g_i}{\partial p_i}(p)\right| > \sum_{j\neq i}\left|\frac{\partial g_i}{\partial p_j}(p)\right|, \quad \text{for route indices } i, j = 1, \ldots, K, \tag{9.8}$$

where $g(p) := p + h(p) \ \ \forall\, p \in \Re^K$. This means that the congestion in the ith route will be much more sensitive to its own price changes than to the price changes in other competing routes.

We shall now describe qualitatively what to expect from the price adaptation scheme for the various, progressively more restrictive scenarios described above. In absence of (B1)–(B3), since (9.5) still converges to the equilibrium set S ‘generically’, we can expect the prices to wander in a neighborhood of this set with high probability. Note, however, that any price vector in S will lead to average congestion being z^*. Thus, while the prices may wander, one can expect the average congestion profile to fluctuate around z^* as desired. The situation when (B1)–(B3) hold is better. Results of [12] suggest that typically the adaptation scheme will spend large times in the neighborhood of one of the isolated equilibria in S with rare transitions from one to the other. Under (9.8), of course, we expect the adaptation scheme to remain in a neighborhood of p^* most of the time.

We remark here that our control scheme allows for occasional resets. To motivate the need for such a feature, consider a network in which there has been a link failure or capacity augmentation causing z^* to be changed. In this case the system may be reset and the queue lengths will begin to adapt to the new operating point. Finally, there may be delays in obtaining and processing price information due to physical transmission delays and/or due to the need for reducing the overhead imposed by this feedback. However, arguing along the lines of [6], the effect of delay in the analysis of the stochastic approximation scheme above can be shown merely to add an additional $O(a)$ error (Recall that the ‘error’ $|p^r(t) - p^r(t-\tau)|$ for a bounded random delay τ is $O(aT)$, where T is any bound on τ). Hence we ignore it. Simulations validate this observation.

9.4.2 Alternative price adaptation equations

The following alternative schemes can also be used.

1. *Linear Relative Deviation Scheme*: This scheme is motivated by the argument that the marginal price per unit deviation in the route congestion should be higher for the route which is targeted to operate at lower congestion level. Thus we replace $(z^r(t_{i+1}) - z^{r^*})$ in the linear-absolute scheme by $\left(\frac{z^r(t_{i+1}) - z^{r^*}}{z^{r*}}\right)$. The adaptation equation under this scheme will be

$$p^r(t_{i+1}) = \mathit{\Gamma}\left(p^r(t_i) + ap^r(t_i)\left(\frac{z^r(t_{i+1}) - z^{r^*}}{z^{r*}}\right)\right) \ \forall\, r \in \mathcal{R}. \tag{9.9}$$

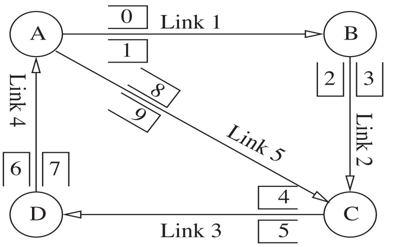


Fig. 9.1. A four-node, five-link network. Each queue is numbered to identify the routes. Even numbered queues are the QoS queues and the odd numbered queues are non-QoS queues.

2. *Quadratic Deviation Scheme:* The squares of the actual and nominal traffic are used to calculate the deviation in the adaptation equation. This will enhance the errors and hence provide higher marginal prices in the upward swings (which is of more concern because they may cause packet drops) than the downward swings. The price adaptation equation will be of the form

$$p^r(t_{i+1}) = \Gamma\left(p^r(t_i) + ap^r(t_i)\left((z^r(t_{i+1}))^2 - (z^{r^*})^2\right)\right) \;\forall\, r \in \mathcal{R}. \quad (9.10)$$

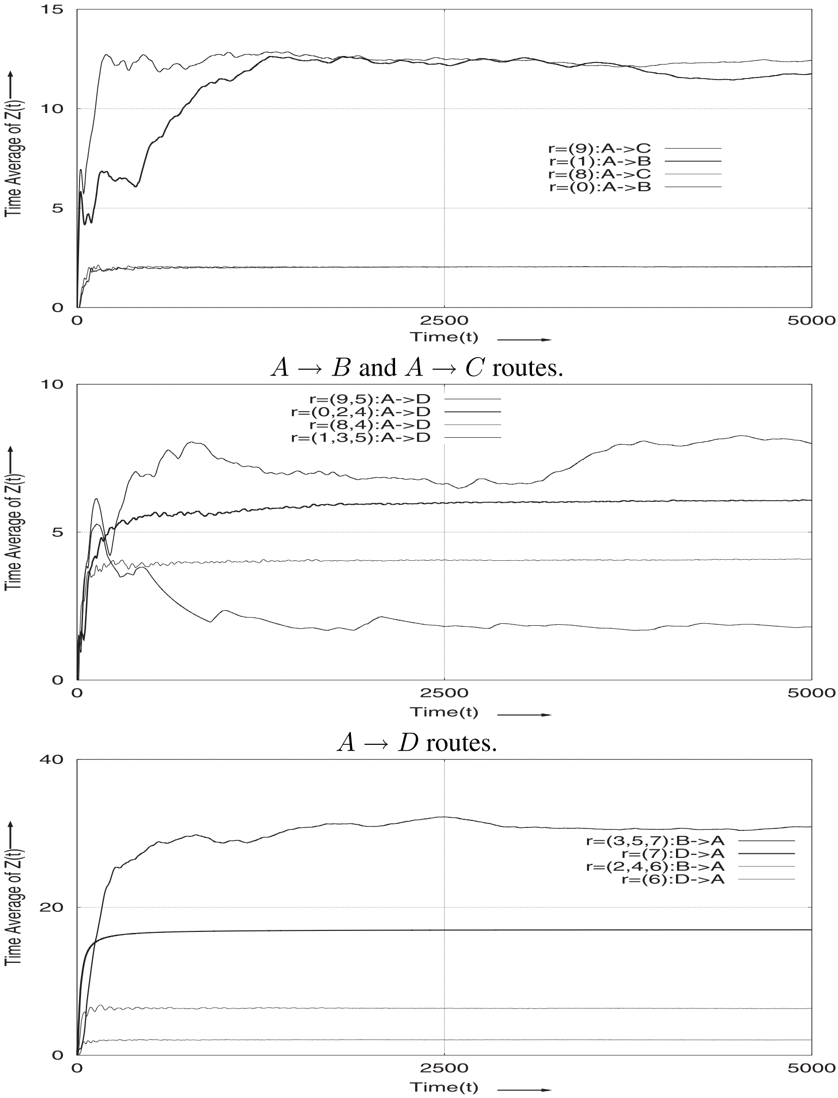
3. *Quadratic Relative Deviation Scheme:* This is the quadratic version of the linear-relative scheme.

$$p^r(t_{i+1}) = \Gamma\left(p^r(t_i) + ap^r(t_i)\left(\frac{(z^r(t_{i+1}))^2 - (z^{r^*})^2}{(z^{r^*})^2}\right)\right) \;\forall\, r \in \mathcal{R}. \quad (9.11)$$

Other variations are also possible. We could calculate the average deviations and the relative deviations over small moving time-windows or shifting time-windows. We could also mandate that a price change be committed only if the price change (relative or absolute) exceeds a specified threshold. We could also discretize the prices.

9.5 Simulation Results

We describe the results from our simulations of a four-node, five-link network shown in Fig. 9.1. Note that the links are directional. Each link maintains two classes of queues—class 0 for QoS traffic where the queue length will be maintained at 2 packets for each route and class 1 is a non-QoS class with no price for service. Buffer capacity of class 0 queues is 8 while that of class 1 queues is 20. The routes are shown in Table 9.1. Propagation delays and the communication delays in conveying the prices are assumed to be zero. Interarrival times are i.i.d. hyperexponential (an equal mixture of exponentials of rate 1.5 and 0.5) and service times are constant. The small user model is considered in which an arrival of class α at time t chooses the route for which $\alpha p^r(t) + (1-\alpha)\sum_{i,j\in r} y_{i,j}(t)$ is minimum from among those that it can take to reach its destination, α is uniformly distributed in $(0,1)$. For example, a packet from A to D has a choice of four routes while that from A to B has only two



$A \to B$ and $A \to C$ routes.

$A \to D$ routes.

$B \to A$ and $D \to A$ routes.

Fig. 9.2. Long term congestion.

Src-Dstn	Routes
$A \to B$	$[(1,0)]$ (queue 0); $[(1,1)]$ (queue 1)
$A \to D$	$[(1,0),(2,0),(3,0)]$ (queues 0, 2 & 4), $[(1,1),(2,1),(3,1)]$ (queues 1, 3 & 5)
	$[(5,0),(3,0)]$ (queues 8 & 4); $[(5,1),(3,1)]$ (queues 9 & 5)
$B \to A$	$[(2,0),(3,0),(4,0)]$ (queues 2, 4 & 6); $[(2,1),(3,1),(4,1)]$ (queues 3, 5 & 7)
$D \to A$	$[(4,0)]$ (queue 6); $[(4,1)]$ (queue 7)
$A \to C$	$[(5,0)]$ (queue 8); $[(5,1)]$ (queue 9)

Table 9.1. The route set for the simulation experiments. The sequence of queues for the routes are also given.

choices. The simulation is run for one million time units. As we say above, $z^{r*}_{i,j} = 2$ for all i, j, r and the $z^{r*} = z^{r*}_{i_1,j_1} + \cdots + z^{r*}_{i_{n_r},j_{n_r}}$.

The congestion data is collected at the arrival instants. The long term average of the congestion at time t is calculated by taking the sample average of the congestion seen by arrivals up to time t. Fig. 9.2 shows the long term congestion in the 12 routes in the network. Observe that in the routes with controlled queues, the average congestion approaches the prescribed value fairly quickly and maintains the prescribed average.

The short-term average of the congestion seen by the arrivals is calculated by using a shifting-window and obtaining the sample average over the arrivals in the window. Fig. 9.3 shows these short term averages for the last 100,000 time units of the simulation. Note that it is reasonably close to the prescribed value. Fig. 9.3 also shows the instantaneous congestion for the last 500 time units of the simulation. Observe that there could be some fluctuations in the instantaneous values while the averages are maintained at the prescribed value.

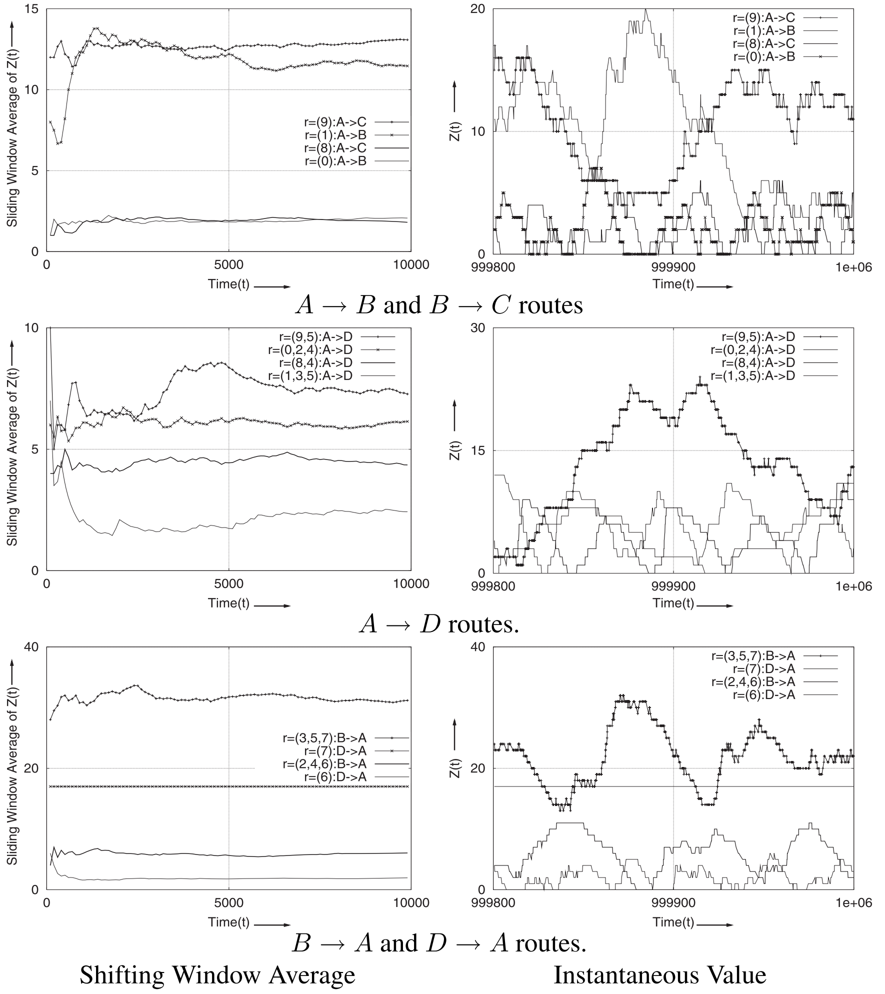
We also show the instantaneous queue lengths of the ten link-queues in the network for the last 1000 time units of the simulation. Observe that the queues do saturate and packets get dropped in both the controlled and the uncontrolled queues.

Recall from our discussion earlier that communication delays cause an $O(aT)$ error where T is an upper bound on the delay. To verify this, in [7] extensive simulations were carried out for the single link case with communication delays in conveying both the price and the congestion information and it was found that the adaptation scheme works well. We don't expect this would change in the network case.

9.6 Implementation Issues

9.6.1 Choosing the operating point

We suggest three possible ways to choose the operating point. A simple scheme is devised to guarantee a certain average delay on each route. Let d^r be the desired mean delay on route r. Partition d^r into $d^r_{i_1,j_1}, \ldots, d^r_{i_{n_r},j_{n_r}}$ such that $d^r_{i_1,j_1} + \cdots + d^r_{i_{n_r},j_{n_r}} = d^r$ and choose $z^{r*} = d^r$ and $z^{r*}_{i_k,j_k} = d^r_{i_k,j_k}$, $k = 1, \ldots, n_r$.



$A \rightarrow B$ and $B \rightarrow C$ routes

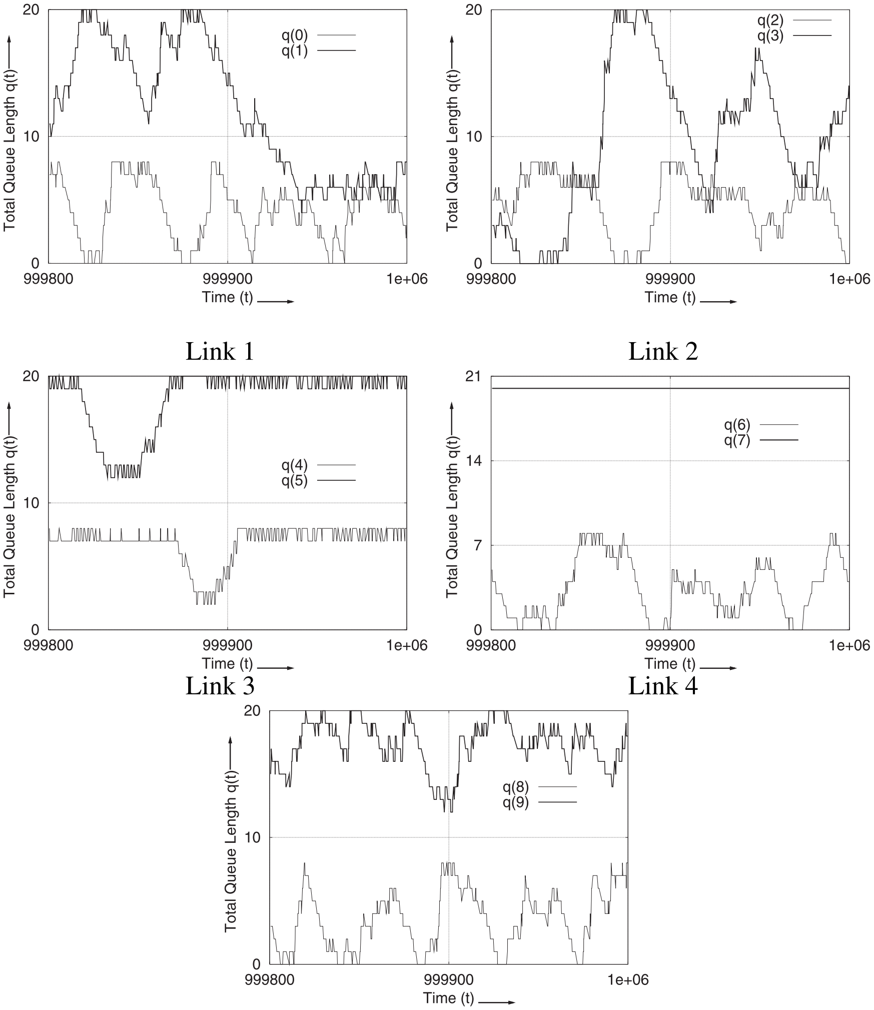
$A \rightarrow D$ routes.

$B \rightarrow A$ and $D \rightarrow A$ routes.

Shifting Window Average Instantaneous Value

Fig. 9.3. The left graphs show the average congestion on the route measured using a shifting window of 100 units. The right graphs show the instantaneous congestion.

A second scheme would be to choose z^{r*} to satisfy an upper bound on the packet drop probability. Define service grade j on link i by $\alpha_{i,j} > 0$ which is an upper bound on the packet drop probability. This means $\nu\left(y_{i,j}(t) > b_{i,j}\right) \leq \alpha_{i,j}$, where $b_{i,j}$ is the upper bound on $y_{i,j}(t)$ and $\nu(\cdot)$ is the probability measure. Using Chebyshev inequality, $\nu\left(y_{i,j}(t) > b_{i,j}\right)$ may be bounded from above by $y^*_{i,j}/b_{i,j}$. Therefore, $y^*_{i,j} \leq b_{i,j}\alpha_{i,j}$ would ensure the desired level of QoS. Further, the operating points z^*_r for the routes can be decided by solving the following optimization problem:



Link 1

Link 2

Link 3

Link 4

Link 5

Fig. 9.4. Instantaneous total queue lengths in the ten link-queues in the network.

$$\max\left(\min_{r:r\in\mathcal{R}} \; z^r\right)$$

subject to

$$(b_{i,j}\alpha_{i,j} \geq)\;\, y^*_{i,j} \geq \sum_{r\in\mathcal{R}_{i,j}} z^r \;;\; j = 1,2,\ldots,J_i \;;\; i = 1,2,\ldots,N$$

Here $\mathcal{R}_{i,j}$ is the set of routes using service grade j on link i. The idea behind the scheme above is to accommodate as much traffic on the routes as we can, while still maintaining the prescribed level of QoS at each of the individual queues.

A third method to choose the network operating point z^{r*} is as follows. Let $c_{i,j}$ be the penalty per packet loss from the queue of service grade j on link i. Then the average penalty for this queue would be $c_{i,j}\nu\left(y_{i,j}(t) > b_{i,j}\right)$ which is bounded from above by $c_{i,j}y^*_{i,j}/b_{i,j}$, and which is further bounded from above by $c_{i,j}\alpha_{i,j}$. Since the users are charged for their traffic on each route, the service provider would ideally like the routes to operate at point $z^* = [z^{1^*}, z^{2^*}, \ldots, z^{K^*}]$ that would maximize the profit. Thus, the operating point z^* can be decided by solving the following problem that maximizes the minimum profit of the service provider:

$$\begin{aligned}
&\max\left\{\sum_r \eta z^r - \sum_{i,j}\left(c_{i,j}\alpha_{i,j}\sum_r z^r_{i,j}\right)\right\}\\
&\text{subject to}\\
&(b_{i,j}\alpha_{i,j} \geq)\ y^*_{i,j} \geq \sum_r z^r_{i,j} \text{ for } j = 1, \ldots, J_i \text{ and } i = 1, \ldots, N
\end{aligned}$$

where η is the minimum possible price that the service provider can charge.

9.6.2 Communicating prices and congestion information

The instantaneous prices and the congestion parameters of the routes on each of the links may be exchanged using a link-state protocol like in OSPF. Simple modifications to the standard OSPF protocol along the lines of the QOSPF protocol suffice. With the congestion information on all the links available at the ingress link, the ingress node can calculate the route congestion and hence choose the route on which the packet is to be forwarded. Note that our analysis allows for non-zero delays in communicating the congestion and pricing information and the effect is shown to be negligible.

9.7 Discussion

This chapter leaves plenty of room for further work in several directions. One can try out many other variations of the price adaptation scheme and develop heuristics that are in the same spirit as (9.4). See [7] for more suggestions. The operating point could also be selected using other methods, e.g., those based on the arrival rate of packets into the route. Of course, we could experiment with different arrival processes and service time distributions and user models. Some of these have been investigated but not presented here and we have found that the control objective is achieved.

An important variation would be to develop a *combined link-route pricing*. Intuitively, one would like to assign prices to the 2-tuple of service grade and link rather than to routes. The route price should be simply the sum (or a simple function) of the prices of the service grades on the links over which the route is defined. A naive possibility is to obtain route prices using the adaptation equation of (9.4) for *some* routes chosen so that if we assume the route prices are sums of link prices, one can solve for the latter explicitly in terms of the former. In turn, the prices for the remaining routes can be calculated by simply summing the prices of links that occur in the route. Our preliminary numerical experiments with such a scheme *did not* show encouraging results. Nevertheless it is worth exploring further.

If the number of source-destination pairs is large, the schemes discussed in this chapter may not be practical because of the number of routes one needs to define. However, note that much of networking literature essentially considers routes, especially while considering stream or circuit multiplexed traffic. Also, with the emergence of MPLS technology and the use of label switched paths which are essentially routes, the idea that we present in this chapter does not look so impractical and can be implemented at the ingress router of an MPLS domain. Further, we believe that the scheme that we present in this chapter can have applications to DiffServ-aware MPLS networks as described in [35].

Acknowledgments

Dinesh Garg acknowledges Prof. Y. Narahari for providing him with the initial exposure to this strand of research and for facilitating his visit to TIFR, Mumbai, during which much of this work carried out.

References

[1] E. Altman, T. Başar and R. Srikant, Nash equilibria for combined flow control and routing in networks: Asymptotic behavior for a large number of users, In *Proceedings of the 38 IEEE Conference on Decision and Control*, Phoenix, AZ, December 1999.

[2] J. Altmann, B. Rupp and P. Varaiya, Internet demand under different pricing schemes, In *Proceedings of ACM E-Commerce*, Cambridge, MA, December 1999.

[3] T. Başar and R. Srikant, Revenue-maximizing pricing and capacity expansion in a many-users regime, In *Proceedings of IEEE INFOCOM*, New York, June 2002.

[4] M. Benaim and M.W. Hirsch, Stochastic approximation algorithms with constant stepsize whose average is cooperative, *The Annals of Applied Probability*, **9**, 1999.

[5] S. Blake, M. Black, D. Carlson, E. Davies, Z. Wang and W. Weiss, An architecture for differential services, *IETF RFC*, **2475**, December 1998.

[6] V.S. Borkar, Asynchronous stochastic approximation, *SIAM Journal of Control and Optimization*, **36**:840–851, 1998. (A correction note in ibid. **38** (2):662–663, 2000).

[7] V.S. Borkar and D. Manjunath, Charge based control of diffserv-like queues, *Automatica*, **40** (12), December 2004.

[8] V.S. Borkar and S.P. Meyn, The O.D.E. method for convergence of stochastic approximation and reinforcement learning, *SIAM Journal of Control and Optimization*, **38**:447–469, 2000.

[9] R. Cocchi, S. Shenker, D. Estrin and L. Zhang, Pricing in computer networks: Motivation, formulation, and example, *IEEE/ACM Transactions on Networking*, **1** (6):614–27, December 1993.

[10] L.A. DaSilva, Pricing for QoS-enabled networks: A survey, *IEEE Communication Surveys*, 2000.
http://www.comsoc.org/livepubs/surveys/public/2q00issue/dasilva.html.

[11] P. Dube, V.S. Borkar and D. Manjunath, Differential join prices for parallel queues: Social optimality, dynamic pricing algorithms and application to Internet pricing, In *Proceedings of IEEE INFOCOM*, New York, June 2002.

[12] P. Dupuis, Large deviations analysis of some recursive algorithms with state-dependent noise, *The Annals of Probability*, **16**:1509–1536, 1988.

[13] R. Edell and P. Varaiya, Providing internet access: What we learn from INDEX, *IEEE Network*, **13** (5):18–25, 1999.

[14] M. Falkner, M. Devetsikiotis and I. Lambadaris, An overview of pricing concepts for broadband IP networks, *IEEE Communications Surveys*, **3** (2), 2000.
http://www.comsoc.org/pubs/surveys/2q00issue/2q00-lead.html.

[15] A. Ganesh and K. Laevens, Dynamics of congestion pricing, Technical Report MSR-TR-2000-07, Microsoft Research, Microsoft Corporation, Redmond, WA, June 2000.

[16] R.J. Gibbens and F.P. Kelly, Resource pricing and the evolution of congestion control, *Automatica*, **35**:1969–1985, 1999.

[17] R.J. Gibbens, R. Mason and R. Steinberg, Internet service classes under competition, *IEEE Journal on Selected Areas in Communication*, **18** (12):2490–2498, 2000.

[18] A. Gupta, D. Stahl and B. Whinston, Priority pricing in integrated service networks, In L.W. McKnight and J.P. Bailey, Eds., *Internet Economics*, 323–352, MIT Press, Cambridge, MA, 1997.

[19] A. Gupta, D.O. Stahl and A.B. Whinston, Pricing of services on the Internet, In F. Phillips and W.W. Cooper, Eds., *IMPACT: How ICC Research Affects Public Policy and Business Markets, A Volume in Honor of G. Kozmetsky*, Quorum Book, CT, 1995.

[20] L. He and J. Walrand, Internet pricing with multiple providers, In *Proceedings of the Allerton Conference*, October 2003.

[21] M.W. Hirsch, Systems of differential equations that are competitive or cooperative, II: Convergence almost everywhere, *SIAM Journal of Mathematical Analysis*, **16**:423–439, 1985.

[22] M.W. Hirsch and S. Smale, *Differential Equations, Dynamical Systems, and Linear Algebra*, Academic Press, New York, NY, 1997.

[23] R. Jain, T. Mullen and R. Hausman, Analysis of Paris Metro pricing for QoS with a single service provider, In *Proceedings of International Workshop on Quality of Service (IWQoS)*, Karlsruhe, Germany, June 2001.

[24] T. Jiminez, E. Altman, T. Başar and N. Shimkin , Competitive routing in networks with polynomial costs, *IEEE Transactions on Automatic Control*, **47** (1):92–96, January 2002.

[25] F.P. Kelly, Charging and rate control for elastic traffic, *European Transactions on Telecommunications*, **8**:33–37, 1997.

[26] F.P. Kelly, A. Maulloo and D. Tan, Rate control in communication networks: Shadow prices, proportional fairness, and stability, *Journal of the Operational Research Society*, **49**:237–252, 1998.

[27] Y.A. Korilis, A. Lazar and A. Orda, Architecting non-cooperative networks, *IEEE Journal on Selected Areas in Communications*, **13** (7):1241–1251, September 1995.

[28] Y.A. Korilis and A. Orda, Incentive-compatible pricing strategies for QoS routing, *Networks and Spatial Economics*, Special Issue on Crossovers between Transportation Planning and Telecommunications, **4** (1):39–53, 2004.

[29] H.J. Kushner and G. Yin, *Stochastic Approximation Algorithms and Applications*, Springer-Verlag, New York, 1997.

[30] R. La and V. Anantharam, Optimal routing control: Game theoretic approach, In *Proceedings of the 36 IEEE Conference on Decision and Control*, San Diego, CA, December 1997.

[31] R. La and V. Anantharam, Charge-sensitive TCP and rate control in the Internet, In *Proceedings of the IEEE INFOCOM*, 1166–1175, Tel Aviv, Israel, March 2000.

[32] R. La and V. Anantharam, Optimal routing control: Repeated game approach, *IEEE Transactions on Automatic Control*, **47**(3):437–450, March 2002.

[33] R. La and V. Anantharam, Utility based rate control in the Internet for elastic traffic, *IEEE/ACM Transactions on Networking*, **10** (2):272–286, April 2002.

[34] A.A. Lazar and N. Semret, Auctions for network resource sharing, Technical Report CU/CTR/TR 468-97-02, Center for Telecommunications Research, Columbia University, 1997.

[35] F. Le Faucheur, L. Wu, B. Davie, S. Davari, P. Vaananen, R. Krishnan, P. Cheval and J. Heinanen, Multi-protocol label switching (MPLS) support of differentiated services, *IETF RFC*, **3270**, May 2002.

[36] S.H. Low and D.E. Lapsley, Optimization flow control-I: Basic algorithm and convergence, *IEEE/ACM Transactions on Networking*, **7** (6):861–74, December 1999.

[37] S.H. Low and R. Srikant, A mathematical framework for designing a low-loss, low-delay internet, *Network and Spatial Economics*, **4** (1):75–102, March 2004.

[38] S.H. Low and P. Varaiya, A new approach to service provisioning in ATM networks, *IEEE/ACM Transactions on Networking*, **1** (5):547–53, October 1993.

[39] P. Marbach, Priority service and max-min fairness, In *Proceedings of IEEE INFOCOM*, New York, June 2002.

[40] P. Marbach and R. Berry, Downlink resource allocation and pricing for wireless networks, In *Proceedings of IEEE INFOCOM*, New York, June 2002.

[41] A. Odlyzko, Paris Metro pricing for the Internet, In *Proceedings of ACM Conference on Electronic Commerce (EC’99)*, 140–147, Denver, CO, November 1999.

[42] A. Orda, R. Rom and N. Shimkin, Competitive routing in multiuser communication networks, *IEEE/ACM Transactions on Networking*, **1** (5):510–521, 1993.

[43] A. Orda and N. Shimkin, Incentive pricing in multiclass systems, *Telecommunication Systems*, **13**:241–267, 2000.

[44] J.M. Ortega and W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

[45] I. Ch. Paschalidis and Y. Liu, Static pricing in multiservice communication networks, In *Proceedings of IEEE INFOCOM*, New York, June 2002.

[46] I. Ch. Paschalidis and J.N. Tsitsiklis, Congestion-dependent pricing of network services, *IEEE/ACM Transactions on Networking*, **8** (2):171–184, April 2000.

[47] Y. Qiu and P. Marbach, Bandwidth allocation in ad hoc networks: A price-based approach, In *Proceedings of IEEE INFOCOM*, San Francisco, CA, April 2003.

[48] J. Sairamesh, D.F. Ferguson and Y. Yemini, An approach to pricing, optimal allocation and Quality of Service provisioning in high-speed packet networks, In *Proceedings of IEEE INFOCOM*, Boston, MA, April 1995.

[49] J. Shu and P. Varaiya, Pricing network services, In *Proceedings of IEEE INFOCOM*, San Francisco, CA, April 2003.

[50] H.L. Smith, *Monotone Dynamical Systems*, American Mathematical Society, Providence, RI, 1995.

[51] M. Spivak, *Calculus on Manifolds*, W.A. Benjamin Inc, New York, 1965.

[52] R. Tandra, N. Hemachandra and D. Manjunath, Diffserv node with join minimum cost queue policy and multiclass traffic, *Performance Evaluation*, **55**:69–91, 2004.

[53] R. Tandra, N. Hemachandra and D. Manjunath, Join minimum cost queue for multiclass customers: Stability and performance bounds, *Probability in the Engineering and Informational Sciences*, **18**:445–472, 2004.

[54] J. Walrand and P. Varaiya, *High-Performance Communication Networks*, Morgan Kaufmann, San Francisco, CA, 2nd Edition, 2000.

[55] H. Yaiche, R.R. Mazumdar and C. Rosenberg, A game-theoretic framework for bandwidth allocation and pricing in broadband networks, *IEEE/ACM Transactions on Networking*, **8** (5):667–678, October 2000.

10

Achieving Fairness in a Distributed Ad-Hoc MAC

Rajarshi Gupta and Jean Walrand

Department of Electrical Engineering and Computer Sciences
University of California, Berkeley, USA
{guptar,wlr}@eecs.berkeley.edu [*]

Summary. Many distributed multiple access (MAC) protocols use an exponential backoff mechanism. In that mechanism, a node picks a random backoff time uniformly in an intervals that doubles in size after a collision. When used in an Ad-Hoc network, this backoff mechanism is unfair towards nodes in the middle of the network. Indeed, such nodes tend to experience more collisions than nodes with fewer neighbors; consequently, they often choose larger delays than those other nodes. We propose a different backoff mechanism that achieves a fairer allocation of the available bandwidth by *decreasing* the backoff delay upon collision or failure to send a packet. That is, a node becomes more aggressive after each failure. Accordingly, we call the mechanism the Impatient Backoff Algorithm (IBA). The nodes maintain the stability of the algorithm by resetting, in a distributed way, the average backoff delays when they become too small. We perform a Markov analysis of the system to prove stability and fairness in simple topologies. We also use simulations to study the performance of IBA in random Ad-Hoc networks and compare with an exponential backoff scheme. Results show that IBA achieves comparable mean throughput, while delivering significantly better fairness.

Key words: Fairness, MAC, Ad-Hoc Networks.

10.1 Introduction

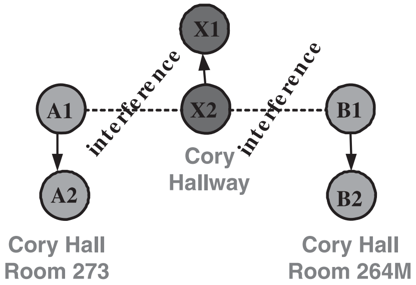
It has been observed that the widely used exponential backoff mechanism (e.g., IEEE 802.11) is unfair towards nodes in the middle of an Ad-Hoc network with multiple interference domains (see [1] and [2]). This unfairness results from the higher degree of contention that these nodes face compared to nodes at the outer edges. We illustrate that unfairness and we propose a new backoff scheme to reduce it.

10.1.1 Unfairness of exponential backoff

To demonstrate the unfairness of the exponential backoff, we consider the network with three links as shown in Fig. 10.1. Transmissions A1-A2 and B1-B2 both

[*] This work was supported by the National Science Foundation under a NeTS grant.

interfere with X1-X2, while A1-A2 and B1-B2 do not interfere with each other. Link X therefore faces more contentions and is more likely to experience collisions than its neighbors.



Active Link	Received on A	Received on B	Received on X
A	6		
A,B	6	6	
A,X	3		3
A,B,X	4	4	2

Fig. 10.1. Simple topology demonstrating the unfairness of 802.11.

We perform a simple experiment with three pairs of laptops located in three different rooms, as shown in Fig. 10.1, to implement this topology. The laptops use 802.11b with link rates of 11 Mbps. The table summarizes the achieved rates on each link. Note that MAC overhead limits the maximum possible capacity of a link to $6Mbps$. We observe that links A and B can simultaneously achieve the maximum $6Mbps$ rate. When links A and X are on at once, they share the channel equally, each receiving approximately $3Mbps$. On the other hand, if A, B and X are all on simultaneously, link X's throughput drops significantly. This simple experiment indicates the unfairness of 802.11b towards nodes in the middle of the network.

It is easy to see the cause of this unfairness. 802.11b follows a backoff mechanism whereby nodes try to capture the channel after waiting for a random backoff selected uniformly in an interval. Upon a collision, nodes are required to double the interval and try again (i.e., try less aggressively). Consider node X above. It faces more contention than nodes A and B, and thereby has a lesser chance of success. Consequently, it collides more often than nodes A and B, and backs off more. As a result, node X succeeds much less often — as demonstrated by the results in Fig. 10.1. More generally, this phenomenon biases the network against nodes in the middle of an Ad-Hoc network. This effect is particularly undesirable because multi-hop routes tend to use the middle of the network more often than the outer edges.

The motivation for this approach is stability: by backing off exponentially fast, the rate of transmission attempts decreases quickly even if more nodes become active. Consequently, the likelihood that one node succeeds quickly approaches 1. Of course, the backoff delay increases somewhat but remains generally small compared to the packet transmission times. When all the nodes share a collision domain, the fairness issue does not arise. Thus, such a backoff scheme is suitable for the shared collision domain of the Aloha network, the original Ethernet, and typical 802.11 configurations. As we just discussed, the situation is quite different in Ad-Hoc networks.

10.1.2 Impatient backoff algorithm

We propose a novel backoff mechanism, the Impatient Backoff Algorithm (IBA), that attempts to improve the fairness in a distributed MAC algorithm running across multiple interference domains. When using IBA, nodes *decrease* their average backoff delay upon collision — thereby becoming *more* aggressive in attempting to capture the next slot. Also, nodes increase their average backoff delay upon successful transmission. The danger of the scheme becoming unstable because of frequent collisions is handled by resetting the average backoff delays when they get too small.

We use a Markov chain model to show that IBA achieves fairness in simple topologies. We demonstrate the stability of the backoff scheme under reasonable assumptions by proving the positive recurrence of the Markov chain. We also evaluate the throughput tradeoff required in order to achieve fairness.

Our analysis does not take into account collisions that happen due to propagation delays and imperfect knowledge of interference. We have built a simulation model that captures these effects to study the performance of IBA in an arbitrary topology and compare it against traditional 802.11-like exponential backoff mechanisms. Simulation results demonstrate that IBA is able to maintain a level of mean throughput comparable to exponential backoff in a random network — but achieves significantly better fairness. We use the simulation model to study the effect of a realistic reset mechanism which propagates imperfectly the reset control message hop by hop. We also evaluate the variations caused by the values of certain design parameters and comment on their choices.

The chapter is organized as follows. We begin by describing related work in Section 10.2. In Section 10.3, we present the backoff model for IBA, which we evaluate analytically in Section 10.4. IBA design parameters and their effects are discussed in Section 10.5, while Section 10.6 presents simulation results before we end with sections on conclusions and future work.

10.2 Related Work

There is a wide body of literature dealing with 802.11 protocols (e.g. [3]), standards being available at [4]. Our focus is on the backoff mechanism utilized to handle congestion in these networks — and their resulting throughput and fairness.

Bianchi [5] presents a two-dimensional Markov model of the exponential backoff mechanism in 802.11. By assuming that the probability of collision of a node does not depend on its own state history, the author is able to derive expressions for the packet transmission probability and saturation throughput. Ergen and Varaiya extend that work in [6]. Their model incorporates carrier sense, non-saturated traffic and SNR, for both basic and RTS/CTS access mechanisms. Analysis of the model shows that the throughput first increases, and then decreases with the number of active stations. A valuable aspect of this work is the modeling of variable packet lengths and the un-slotted nature of the protocol (i.e., no synchronization between nodes). Varaiya and co-workers have also looked at related problems in [7], [8] and [9].

Slotted media access protocols have been studied for several decades (see [10]for an overview). In particular, a detailed analysis of Slotted Aloha MAC has evaluated its throughput and fairness. More recently, Yuan and Marbach [11] have proposed a rate control for random access networks. By controlling the *rate* at which the nodes attempt to transmit, by increasing it when idle and decreasing it after collision, the system is shown to be stable, i.e., positive recurrent.

Our model for IBA differs from [5], [6] and also [11] in a crucial aspect — we consider Ad-Hoc networks spanning multiple interference domains. Consequently, the nodes do *not* all share the same medium, and so have *different* degrees of contention, and therefore different collision probabilities. It is this fact that biases such protocols against middle nodes in the network. Addressing this fairness issue is the primary goal of this chapter, and leads us to propose the strategy of becoming *more* aggressive upon collision.

10.3 Backoff Model

10.3.1 Assumptions

We make simplifying assumptions on the MAC model. The main assumption is that packet transmissions occur in a slotted and synchronized fashion. Each packet time slot is divided into two phases:

1. Backoff Contention Phase. In this phase, each node that has a packet to send generates a random backoff value. It waits for these many backoff mini-slots. The backoff mini-slots are much much smaller than the packet transmission slot. If it has not heard a transmission from a neighboring node while waiting, the node sends out a short 'Slot Capture Message'. All neighbors which hear this slot capture message (i.e., within transmission range), or carrier senses noise in the channel (i.e., within interference range) will keep quiet for this slot.
2. Packet Transmission Phase. At the end of the Backoff Contention Phase, all nodes which successfully sent out the slot capture messages will transmit a *constant sized* packet. Thus, only the nodes that have generated a backoff delay smaller than or equal to those of their neighbors get to transmit a packet. A successful transmission is confirmed by an acknowledgement (ack).

An example is shown in Fig. 10.2, where five nodes in a line contend for the channel. Each node is assumed to interfere with neighbors two hops away. The lower part of the picture shows one slot. During the Backoff Contention Phase, node A chooses a smaller backoff than B and C – consequently it is able to send a Slot Capture Message that is heard by B and C, which keep quiet. Nodes D and E are not affected by A's slot capture, and E wins that contention. As a result, nodes A and E utilize the Packet Transmission phase in parallel. Although C had a smaller backoff than E, C is quiet in this slot, allowing E to transmit. Note that nodes wait till the end of the entire Backoff Contention phase, before beginning the Packet Transmission phase.

Collisions can occur in this scheme if two neighbors choose the same backoff. In this case, neither will hear the other's slot capture message. Hence both nodes will try to send packets, and collide, resulting in a wasted packet transmission slot. A collision will also occur if a node within interference range is unable to discern the slot capture. Note that this scheme does not employ an RTS/CTS mechanism, hence hidden terminals [12] will not be accounted for. For instance, the transmissions of B and E do collide at C.

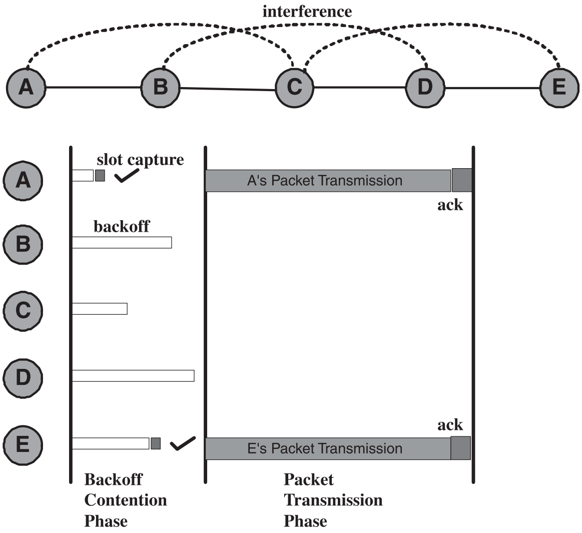


Fig. 10.2. Two phases of Impatient Backoff.

Finally, slotted transmission implicitly assumes synchronization between the nodes. We do not address the details in this chapter, other than to indicate that the level of synchronization required is no more than any standard slotted MAC protocol.

10.3.2 Exponential random backoff rate

Traditional backoff mechanisms choose a random backoff uniformly in $[0, B_L]$, where B_L is the backoff limit. The mean backoff in this case is $B_L/2$. Instead of using an uniform random variable, IBA chooses the backoff using an exponential random variable with mean $B_L/2$. The exact number of backoff mini-slots is then determined by rounding the random variable and capping it at a maximum value, since the exponential random variable is unbounded.

We utilize the exponential random variable since it offers some useful characteristics that we use in the analysis of the scheme in Section 10.4. In particular, when n nodes with mean backoffs $b_1, b_2, \ldots, b_n$ contend, node i wins the contention when backoff $B_i < B_j \forall j \neq i$. The probability of this occurrence is calculated as follows:

$$P(\text{Node } i \text{ wins contention}) = \frac{1/b_i}{\sum_{j=1}^{n}(1/b_j)}. \tag{10.1}$$

In this calculation, we ignore the probability that the smallest backoff delay falls in the same mini-slot as another one.

10.3.3 Updating average backoff delays

The key principle of IBA is that nodes that face more contention become more aggressive so that they can get their fair share of the channel. This is achieved by updating the backoff based on feedback received in the last slot.

Assume that a node has a mean backoff delay b and has a packet to transmit. If the node fails to send in the current slot, either because of a collision or because it loses during the contention phase, it *decreases* its mean backoff delay by a multiplicative factor $m > 1$. On the other hand, if the node transmits successfully, it *increases* its mean backoff delay by the same factor. Note that decreasing the mean backoff delay makes a node more likely to win during the next contention phase.

$$\boxed{\begin{array}{l}\text{Upon failure, } b := b/m \\ \text{Upon success, } b := b \times m\end{array}}$$

10.3.4 Reset

The obvious problem of an aggressive backoff mechanism is one of collision meltdown. When there are a lot of contending nodes, all but one decrease their mean backoff delay. This increases the chance that several of them pick a small backoff delay in the same mini-slot and hence collide. In order to avoid this situation, we propose the IBA Reset mechanism. The idea is to increase the mean backoff delay of *every* node by a constant factor R_F whenever any node's mean backoff delay falls below a reset limit R_L. A reset does not alter the result of the contention phase, since equation 10.1 is unchanged by multiplying every term by R_F. In Section 10.5, we discuss the choice of values for R_L and R_F.

In reality, it is impossible to change the mean backoff delay of all nodes simultaneously; any reset message needs to propagate through the network. In Section 10.6.3, we simulate a reset propagation scheme, as also the loss of some reset messages, and see that their effect on overall throughput and fairness is minimal.

10.3.5 Fairness

Qualitatively, we understand fairness to imply that two nodes in similar situations ought to get the same share of the bandwidth. For simple networks, it is easy to see the "fair" allocation, e.g. in Fig. 10.1, we would like each of the three transmitting nodes to get $1/3$ of the channel bandwidth. In more complicated and/or random topologies, evaluating fairness is a trickier proposition.

There are several metrics of fairness in use e.g. Max-Min Fair [10], Proportional Fair [13], Jain's Fairness Index [14], etc. In this chapter, we utilize Jain's index to measure scheduling fairness, and to compare against other MAC schemes. We prefer Jain's index [14] since it measures the fairness in terms of an optimal desired throughput. Let the measured throughput be $Z_1, \ldots, Z_n$ and the desired throughput be $D_1, \ldots, D_n$. Define $x_i = \frac{Z_i}{D_i}, \forall i = 1 \ldots n$. Then the fairness F is given by

$$F = \frac{\left(\sum x_i\right)^2}{n \sum x_i^2}. \tag{10.2}$$

10.4 Analysis

We first evaluate IBA in some simple topologies that are amenable to analysis. We especially consider the star topology shown in Fig. 10.3(a). We also analyze the triangle clique topology in which three nodes all interfere with each other.

10.4.1 Star topology

The star topology is characterized by a middle node X that interferes with every other node, while the outer nodes do not interfere with each other. The figure shows a star with 4 arms, although we analyze a star with n outer nodes. Recall that we are modeling a non-RTS/CTS scheme. So we are concerned only with the interference between nodes as they transmit and need not consider the location of the receivers.

The star topology is of special interest in terms of fairness, since node X contends with every other node in the network and traditional backoff mechanisms are unfair towards it. We want to ensure that our scheme is fair for this simple topology.

When node X wins the backoff contention, it is the only one that transmits. However, when any other node wins the contention, node X has to keep quiet and so the remaining outer nodes believe they won the round and they all transmit. Thus, assuming that the nodes always have a packet to send, the mean backoff delays of all the outer nodes remain equal.

Let the mean backoff of the middle node be b_1 and that of each of the n outer nodes be b_2. Since we use exponential random variables to generate the backoffs, the probability that the middle node wins the contention, as in equation (10.1), is given by

$$t_X = \frac{\frac{1}{b_1}}{\frac{1}{b_1} + \sum_{j=1}^{n} \frac{1}{b_2}} = \frac{\frac{1}{b_1}}{\frac{1}{b_1} + \frac{n}{b_2}}. \tag{10.3}$$

To simplify notation, we define $r_i = 1/b_i$ to be the rate of the exponential random variable generating the backoff for node i. With this notation, equation (10.3) becomes

$$t_X = \frac{r_1}{r_1 + nr_2}. \tag{10.4}$$

Note that for the purpose of analysis, we assume the backoff to be a real number, and not rounded off to integers, as described in Section 10.3.2. This would prevent collisions from ever occurring since two neighbors have 0 probability of choosing the exact same backoff. In Section 10.4.3, we analyze the star topology with collisions.

It is also worthwhile to consider the throughput-delay tradeoff in the star topology. The maximum throughput is achieved when the middle node is quiet, and the outer nodes are always on, this leads to a total throughput of n. On the other hand, a fair share allows the middle node to be active $1/2$ the time, while the other nodes are active during the remainder. Consequently, the throughput is only $\frac{1}{2} + \frac{1}{2}n = \frac{n+1}{2}$.

10.4.2 Fairness in star topology

Assume that all the outer nodes start with the same initial mean backoff delay. Then, the star topology can be evaluated by an appropriate Markov Chain whose states capture the ratio between the mean backoff delays (i.e., between the backoff rates). Let state S_k designate the state when $\frac{b_2}{b_1} = \frac{r_1}{r_2} = n.(m^2)^k = n.m^{2k}$, as shown in Fig. 10.3(b). Note that the ratio of backoff rates is a Markov chain since the probability of success or failure, and therefore of moving to another state, is completely determined by the ratio of the rates r_1 and r_2 independently of the past.

Following equations (10.3) and (10.4), the probability that the middle node X in state S_k succeeds in the next slot is given by $\frac{n.m^{2k}}{n.m^{2k}+n} = \frac{m^{2k}}{m^{2k}+1}$. If X wins, its backoff rate is updated to $r_1 := r_1/m$, while all the other nodes update their backoff rates to $r_2 := r_2 \times m$. As a result, the ratio between the rates becomes $n.m^{2k-2}$, i.e., the chain moves to state S_{k-1}. Similarly, from state S_k, the probability that X loses in the next lot is $\frac{n}{n.m^{2k}+n} = \frac{1}{m^{2k}+1}$, and this moves the state to S_{k+1}.

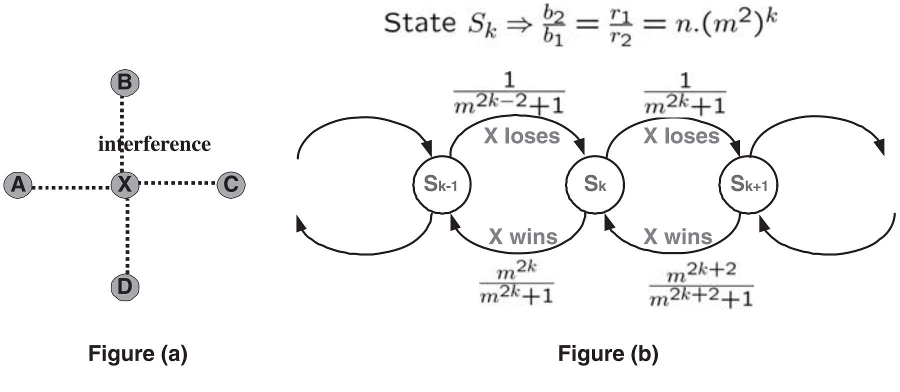


Fig. 10.3. (a) Star topology (b) k^{th} state in star topology M.C.

The Markov chain has the structure of a birth-death chain, as shown in Fig. 10.4, that is symmetric about state S_0. This chain is seen to be positive recurrent for all $m > 1$ because the average drift of the absolute value of the state is negative whenever the state is not 0. We evaluate the steady state probabilities $\pi_i, i \in \mathbb{Z}$ of the various states. We begin by calculating π_1 in terms of π_0.

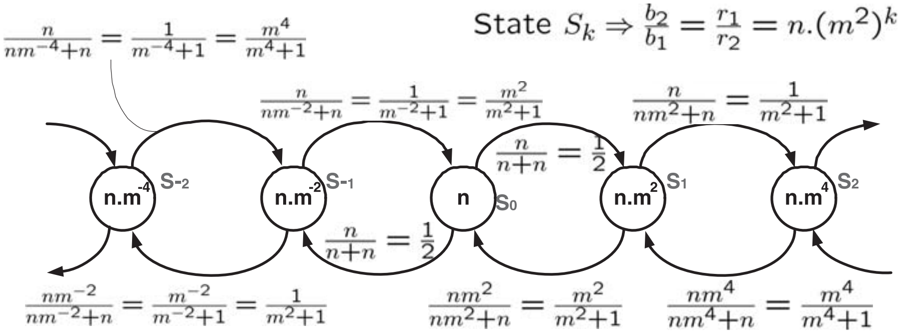

$\frac{n}{nm^{-4}+n} = \frac{1}{m^{-4}+1} = \frac{m^4}{m^4+1}$
State $S_k \Rightarrow \frac{b_2}{b_1} = \frac{r_1}{r_2} = n.(m^2)^k$
$\frac{n}{nm^{-2}+n} = \frac{1}{m^{-2}+1} = \frac{m^2}{m^2+1}$
$\frac{n}{nm^2+n} = \frac{1}{m^2+1}$
$\frac{n}{n+n} = \frac{1}{2}$
n.m^{-4}
S-2
n.m^{-2}
S-1
n
S0
n.m^2
S1
n.m^4
S2
$\frac{n}{n+n} = \frac{1}{2}$
$\frac{nm^{-2}}{nm^{-2}+n} = \frac{m^{-2}}{m^{-2}+1} = \frac{1}{m^2+1}$
$\frac{nm^2}{nm^2+n} = \frac{m^2}{m^2+1}$
$\frac{nm^4}{nm^4+n} = \frac{m^4}{m^4+1}$

Fig. 10.4. Fairness of star topology.

$$\begin{aligned}
\pi_0\left(\frac{1}{2}+\frac{1}{2}\right) &= \pi_{-1}\frac{1}{m^{-2}+1}+\pi_1\frac{m^2}{m^2+1}\\
\Rightarrow \pi_0 &= 2\pi_1\frac{m^2}{m^2+1},\ \text{as } \pi_1=\pi_{-1} \text{ by symmetry}\\
\Rightarrow \pi_1 &= \frac{m^2+1}{2m^2}\pi_0
\end{aligned}$$

We can then use this value of π_1 to determine π_2.

$$\begin{aligned}
\pi_1\left(\frac{1}{m^2+1}+\frac{m^2}{m^2+1}\right) &= \pi_0\frac{1}{m^{2\times 0}+1}+\pi_2\frac{m^{2\times 2}}{m^{2\times 2}+1}\\
\Rightarrow \pi_1.1 &= \pi_0\frac{1}{2}+\pi_2\frac{m^4}{m^4+1}\\
\Rightarrow \pi_0\left(\frac{m^2+1}{2m^2}-\frac{1}{2}\right) &= \pi_2\frac{m^4}{m^4+1}\\
\Rightarrow \pi_0\frac{1}{2m^2} &= \pi_2\frac{1}{1+\frac{1}{m^4}}\\
\Rightarrow \pi_2 &= \frac{1}{2}\times\left(1+\frac{1}{m^{2\times 2}}\right)\times\frac{1}{m^{2\times 1}}\times\pi_0.
\end{aligned}$$

By repeating the process, we evaluate the steady state probability of state S_k as,

$$\pi_k = \frac{1}{2}\times\left(1+\frac{1}{m^{2k}}\right)\times\frac{1}{m^{k(k-1)}}\times\pi_0. \tag{10.5}$$

Finally, we can evaluate the actual probabilities:

$$\sum_{k=-\infty}^{\infty} \pi_k = \pi_0 + 2\sum_{k=1}^{\infty} \pi_k = 1 \tag{10.6}$$

$$\Rightarrow \pi_0 + 2\sum_{k=1}^{\infty} \frac{1}{2}\left(1 + \frac{1}{m^{2k}}\right)\frac{1}{m^{k(k-1)}}\pi_0 = 1 \tag{10.7}$$

$$\Rightarrow \pi_0 \left[1 + \sum_{k=1}^{\infty}\left(1 + \frac{1}{m^{2k}}\right)\frac{1}{m^{k(k-1)}}\right] = 1 \tag{10.8}$$

We can evaluate π_0 for various values of m.

m	1.05	1.2	1.5	2	4	10
π_0	0.123	0.230	0.325	0.395	0.471	0.495

We can also prove the following theorem about the expected transmission probability of nodes in the star topology.

Theorem 1. *Each node in the star topology has an expected transmission probability of* $\frac{1}{2}$.

Proof. The transmission probability of X at state S_k is given by $t_k^X = \frac{m^{2k}}{m^{2k}+1}$ (Fig. 10.3(b)). Then, the expected transmission probability of node X, $E[t^X]$ may be evaluated as

$$E[t^X] = \sum_{k=-\infty}^{\infty} \pi_k t_k^X = \pi_0 \frac{1}{2} + \sum_{k=1}^{\infty} \pi_k (t_k^X + t_{-k}^X). \tag{10.9}$$

However,

$$t_k^X + t_{-k}^X = \frac{m^{2k}}{m^{2k}+1} + \frac{m^{-2k}}{m^{-2k}+1} = 1. \tag{10.10}$$

Hence,

$$E[t^X] = \frac{\pi_0}{2} + \sum_{k=1}^{\infty} \pi_k = \frac{1}{2} \tag{10.11}$$

(from equation 10.6). Finally, the expected probability of successful transmission for the outer nodes is $1 - \frac{1}{2} = \frac{1}{2}$. □

The Markov chain model analyzed above assumes that the initial ratio between r_1 and nr_2 is a power of m^2. If the initial ratio r' lies between two powers of m^2, then the resultant states will be $r' \times m^{2k}, k = -\infty, \ldots, \infty$. This positive recurrent chain will also drift towards states where $r' \times m^{2k} \approx 1$.

In addition to each node having an average transmission rate of 1/2, this Markov chain has a strong drift toward S_0, which suggests that the short-term behavior is fair.

10.4.3 Probabilistic model of star topology with collisions

The model above is not affected by collisions in the system. A collision in the star topology causes *all* the mean backoff delays to be divided by m. As a result, the state does not change and the probability of success for any node, as given in equations (10.3) and (10.4), is unchanged.

Calculating the probability of collision: We take into consideration the probability of collision in the star topology and analyze the resulting chain of events. We model here the probability of collision due to rounding off the chosen backoff delay. Consider some outer node Z. A collision occurs with node X if the two backoff delays B_X and B_Z are within a fixed time $\varDelta$ of each other, corresponding to the duration of a backoff mini-slot.

If $B_Z > B_X + \frac{\varDelta}{2}$, then Z will hear X's slot capture message and keep quiet. The reverse occurs if $B_Z < B_X - \frac{\varDelta}{2}$. So we only need to consider the probability that $B_Z \in [B_X - \frac{\varDelta}{2}, B_X + \frac{\varDelta}{2}]$ given that $B_Z \geq B_X - \frac{\varDelta}{2}$.

$$\begin{aligned}
&P(\text{X collides with A at time } t)\\
&= P\left(B_X \in [t, t+dt]\right) \times P\left(B_Z \in (t - \frac{\varDelta}{2}, t + \frac{\varDelta}{2}) | B_Z \geq t - \frac{\varDelta}{2}\right)\\
&= P\left(B_X \in [t, t+dt]\right) \times P\left(B_Z \in [0, \varDelta]\right), \text{since } B_Z \text{ is memoryless.}
\end{aligned}$$

We know that the pdf of the exponential random variable B_X, with backoff rate $r_1 = 1/b_1$, is given by $b_X(t) = r_1 e^{-r_1 t}$. Similarly $b_Z(t) = r_2 e^{-r_2 t}$. Therefore,

$$p_{XZ} = P(\text{X collides with Z}) \tag{10.12}$$

$$= \int_0^\infty r_1 e^{-r_1 t} \left(\int_0^\varDelta r_2 e^{-r_2 t} du \right) dt \tag{10.13}$$

$$= \left(1 - e^{-r_2 \varDelta}\right) \int_0^\infty r_1 e^{-r_1 t} dt = 1 - e^{-r_2 \varDelta}. \tag{10.14}$$

For a particular outer node Z, the collision probability $p_Z = p_{XZ}$ since X is the only node that it may collide with.

The probability that X collides at all can now be evaluated in terms of the probability of each of the n neighbors. This is the probability that any one of the neighbors picks a backoff delay in $[B_X - \frac{\varDelta}{2}, B_X + \frac{\varDelta}{2}]$, given that its backoff delay is $\geq B_X - \frac{\varDelta}{2}$.

$$p_X = P(\text{X collides}) = 1 - P(\text{X does not collide with anyone}) \tag{10.15}$$

$$= 1 - [(1 - p_{XZ})^n] = 1 - \left[\left(1 - (1 - e^{-r_2 \varDelta})\right)^n\right] \tag{10.16}$$

$$= 1 - e^{-n r_2 \varDelta}. \tag{10.17}$$

Modeling event probabilities in star topology: We can use the values of p_X and p_Z to model the system. Recall that the backoff rates of all the outer nodes move together, so the transmission and collision probabilities of all the outer nodes remain equal, assuming they are the same initially.

Again, let the backoff rates of the middle node be r_1 and the backoff rates of the n outer nodes be r_2. Then, X attempts to transmit with a probability $t_X = \frac{r_1}{r_1+nr_2}$, from equation (10.4). And an outer node transmits with probability $t_Z = \frac{nr_2}{r_1+nr_2}$. Given t_X, and p_X, we can calculate X's probability of successful transmission $s_X = t_X(1 - p_X)$. Similarly, we can also calculate the probability of success s_Z for the outer nodes.

Numerically evaluating the steady state of the system: We evaluate the steady state of the system for values of $n = 2, 4, 10, 20, 50$. In each case, the star topology achieves a fair sharing of the bandwidth. The middle node gets the same throughput as the outer nodes.

Even if we bias the initial rates (e.g. by starting the middle node with an initial mean backoff of 1024 instead of 16), the fairness of the Impatient Backoff Algorithm is maintained. This situation with 10 outer nodes is plotted in Fig. 10.5. While $s_X << s_Z$ initially, it converges quickly (in less than 40 time slots) with s_Z. The two success probabilities then track each other. For the 500 time slots shown in the figure, the average success probabilities $\overline{s_X} = 0.4748$ and $\overline{s_Z} = 0.4750$.

If there is no initial bias, the two success probabilities converge even sooner. The periodic jumps in the values of s_X and s_Z (seen in Fig. 10.5) are caused by the reset phenomenon — when the backoff delays reset, the collision probability falls abruptly, thereby enhancing the success probabilities for the next slot. Without a reset, the two values s_X and s_Z converge asymptotically and stay together.

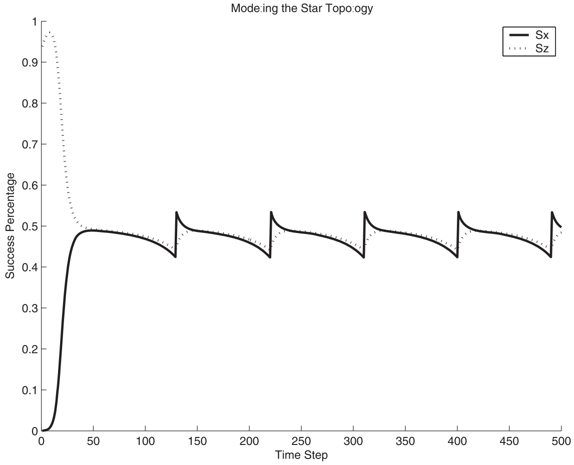


Fig. 10.5. Modeling the star topology (with initial bias against middle node).

The collision probabilities depends on the choice of several parameters. We assume an initial mean backoff of 16. We set $\Delta = 1$, to model the case where two nodes collide when their backoff delays are rounded to the same integer. We also incorporate the reset mechanism outlined in Section 10.3.4. We set the reset limit $R_L = 16/5$ and the reset factor $R_F = 10$.

Fairness in star topology with changing number of neighbors: So far we have assumed that the nodes always have packets to transmit. Now we study the more realistic situation when the nodes only want to transmit some of the time. Every 1000 time slots we randomly pick a subset of the 10 outer nodes to go to sleep, while the middle node always wants to transmit. Thus the relative backoff delays go awry every 1000^{th} time slot and relies on the IBA mechanism to get back in sync.

The results are plotted in Fig. 10.6. The blue line plots the success probability s_X for the middle node while the red line plots the success probability s_Z for the outer nodes (the two lines are too close to identify separately). The number of active nodes n during each block of slots is also plotted using the dotted line.

As seen in the figure, s_X and s_Z track each other closely and neither is able to overwhelm the other, even with a changing number of outer nodes. The average success probabilities over the entire time duration is $\overline{s_X} = 0.4722$ and $\overline{s_Z} = 0.4719$, thus yielding a fairness index of 1.

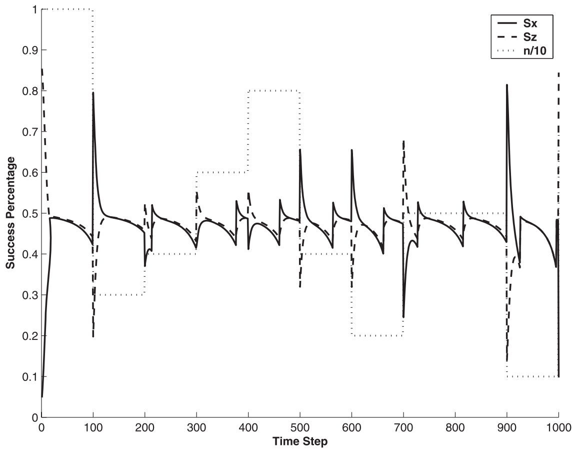


Fig. 10.6. Modeling the star topology with a changing number of neighbors.

10.4.4 Triangle topology model

The other topology that we analyze is the triangle topology, where each of three nodes interferes with the others. Clearly, the situation is symmetric, but we need to ensure that starting at any set of backoff rates, the system drifts back towards a state where all nodes have an equal probability of successful transmission.

The evolution of the system is modeled by a Markov chain that represents the backoff rates of the three nodes. A state is denoted by the triple $(1, b, c)$ that specifies the backoff rates of the three nodes in increasing order. We express each of the rates as a multiple of the smallest rate, so the first term is always 1. Thus the probability of the first node winning the contention is $\frac{1}{1+b+c}$, while the other two nodes win with probability $\frac{b}{1+b+c}$ and $\frac{c}{1+b+c}$, respectively. Every transmission results in one backoff rate being divided by m, and the other two rates being multiplied by m. As a result, the ratio of rates between any two nodes changes by a factor of m^2; consequently values b and c in the Markov chain are powers of m^2. In Fig. 10.7, we represent the Markov chain resulting from a triangle topology. In the figure, we use a value of $m = 2$ as an illustration.

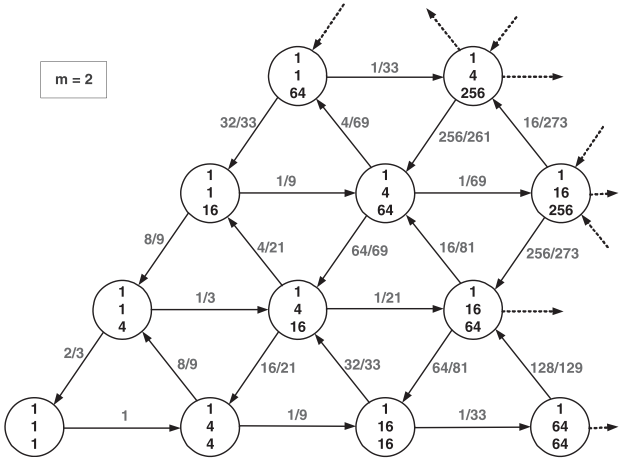


Fig. 10.7. Markov chain of triangle topology.

The Markov chain is clearly irreducible, but it is periodic with a period of 3. In order to conclude that the system is fair, we need to show the Markov chain is positive recurrent.

Theorem 2. *The Markov chain that models the backoff rates in the triangle topology is positive recurrent.*

Proof. Let $\mathcal{A}$ be a finite set of the states (to be defined below), including $(1, 1, 1)$. Define the function f as

$$f(1, b, c) = \log_{m^2} b + \log_{m^2} c. \tag{10.18}$$

Given state S, we further define the Lyapunov drift function

$$\gamma(S) = -f(S) + \left(\sum_{T \in N(S)} P_{ST} . f(T) \right) . \tag{10.19}$$

Here $N(S)$ are the states neighboring to state S, and P_{ST} is the transition probability from state S to state T. By Pakes' Lemma [15], if there is some ϵ such that $\gamma(S) \leq -\epsilon < 0, \forall\ S \notin \mathcal{A}$, then the chain is positive recurrent. Let $0 < \epsilon \ll 1$. By observing the structure of the Markov chain, we notice that there are three varieties of states as illustrated in Fig. 10.8.

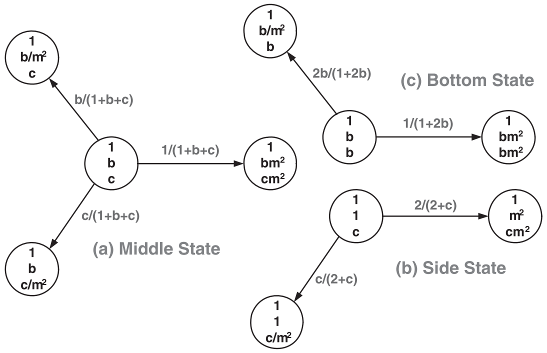


Fig. 10.8. Evaluating Lyapunov function on the triangle topology M.C.

Case 1 (Middle State): In this case, there are three possible neighboring states: $(1, bm^2, cm^2)$, $(1, \frac{b}{m^2}, c)$ and $(1, b, \frac{c}{m^2})$, with transition probabilities as shown in Fig. 10.8(a). Then,

$$\begin{aligned}
&\gamma(S) < -\epsilon \\
&\Rightarrow -\left(\log_{m^2} b + \log_{m^2} c\right) + \frac{1}{1+b+c}\left(\log_{m^2}(bm^2) + \log_{m^2}(cm^2)\right) \\
&\quad + \frac{b}{1+b+c}\left(\log_{m^2}\frac{b}{m^2} + \log_{m^2} c\right) + \frac{c}{1+b+c}\left(\log_{m^2} b + \log_{m^2}\frac{c}{m^2}\right) < -\epsilon \\
&\Rightarrow -\left(\log_{m^2} b + \log_{m^2} c\right) + \frac{1}{1+b+c}\left(1 + \log_{m^2} b + 1 + \log_{m^2} c\right) \\
&\quad + \frac{b}{1+b+c}\left(\log_{m^2} b - 1 + \log_{m^2} c\right) + \frac{c}{1+b+c}\left(\log_{m^2} b + \log_{m^2} c - 1\right) < -\epsilon \\
&\Rightarrow \frac{1}{1+b+c}(2-b-c) < -\epsilon \\
&\Rightarrow \frac{2+\epsilon}{1-\epsilon} < b+c.
\end{aligned}$$

We can verify that this inequality holds when

$$b + c > 3. \tag{10.20}$$

Case 2 (Side State): The side state is characterized by $(1, 1, c)$, and has two possible neighbors: $(1, m^2, cm^2)$ and $(1, 1, \frac{c}{m^2})$. From Fig. 10.8(b),

$$\begin{aligned}
&\gamma(S) = -\left(\log_{m^2} c\right) + \frac{2}{2+c}\left(1 + 1 + \log_{m^2} c\right) + \frac{c}{2+c}\left(\log_{m^2} c - 1\right) < -\epsilon \\
&\Rightarrow \frac{1}{2+c}(4 - c) < -\epsilon \\
&\Rightarrow \frac{4 + 2\epsilon}{1 - \epsilon} < c.
\end{aligned}$$

This is true when

$$c > 5. \tag{10.21}$$

Case 3 (Bottom State): The bottom state is characterized by $(1, b, b)$, and has two possible neighbors: $(1, bm^2, bm^2)$ and $(1, \frac{b}{m^2}, b)$. From Fig. 10.8(c),

$$\begin{aligned}
&\gamma(S) = -\left(\log_{m^2} b + \log_{m^2} b\right) + \frac{1}{2b+1}\left(1 + \log_{m^2} b + 1 + \log_{m^2} b\right) \\
&+ \frac{2b}{2b+1}\left(\log_{m^2} b - 1 + \log_{m^2} b\right) < -\epsilon \\
&\Rightarrow \frac{1}{2b+1}(2 - 2b) < -\epsilon \\
&\Rightarrow \frac{2 + \epsilon}{2(1 - \epsilon)} < b.
\end{aligned}$$

This is true when

$$b > 2. \tag{10.22}$$

The state variables b and c may be expressed as $m^{2k}, k \in \mathbb{Z}^+$. Depending on the value of m, we can choose $\mathcal{A}$ s.t. $(1, b, c) \in \mathcal{A} \iff b = m^{2k_1} \leq 2$ and $c = m^{2k_2} \leq 5$. Then, $\gamma(S) < -\epsilon < 0 \; \forall \; S \notin \mathcal{A}$, since we satisfy equations (10.20)–(10.22). By Pakes' Lemma, the chain is positive recurrent. □

This analysis allows us to conclude that no matter what the initial backoff delays are, the three nodes will drift towards states where their backoff delays are equal. This observation is also confirmed by simulation results in Section 10.6.4.

10.5 Choosing Algorithm Parameters

IBA is characterized by three design parameters m, R_L and R_F whose choices affect the performance of the system. In this section, we present qualitative arguments for our choices.

10.5.1 Reset limit R_L

The reset limit R_L is the smallest value to which we allow the mean backoff to fall until it is reset. Having a very small value for R_L allows nodes to maintain low mean backoff delays. Since the actual number of backoff slots is rounded to the nearest integer, multiple neighbors with low mean backoff delays are more likely to choose the same backoff, leading to frequent collisions.

Choosing a large value for R_L will alleviate this problem but leads to large backoff values. In our current model this has no ill-effect since all nodes always wait for the completion of the backoff contention phase before attempting to transmit. For simulation purposes, we choose $R_L = \frac{16}{5} = 3.2$, which is $1/5^{th}$ of the initial mean backoff value.

10.5.2 Reset factor R_F

Consider a dense subgraph in the Ad-Hoc network, e.g. a clique topology with q nodes. In such a situation, at each slot, at most one node succeeds while all the others fail. Consequently, $q - 1$ of the nodes divide their mean backoff delay by m. For a large enough q, at least one of the nodes repeatedly decreases its rate and cross R_L, causing a reset. Quantitatively, resets occur approximately every $\log_m R_F$ slots, provided $\log_m R_F < q$.

Choosing a large R_F indeed decreases the reset frequency, but this decrease happens only on a slower logarithmic scale. In our simulations we choose $R_F = 10$.

10.5.3 Multiplicative factor m

In Section 10.4.3 and 10.4.4, we showed that IBA is stable and fair in the star and triangle topologies for $m > 1$. In Section 10.6.2, simulations show this to be true even for random topologies. However, we need to select an optimal value for m.

A small value of m allows nodes to maintain a lower mean backoff delay on average, since the backoff rate may be closer to R_L without hitting it. This leads to more collisions, following the same argument as Section 10.5.1. On the other hand, a large value of m causes frequent resets. In a dense network, the frequency of resets is approximately $\log_m R_F$.

The compromise is to choose as large a value of m as possible (to minimize collisions), yet choose it small enough to keep resets under control. For a practical system, this is entirely a design decision based on the efficiency of the reset mechanism. We chose $m = 1.2$ for our simulations. In this case, the reset frequency is bounded above by the rate of one reset every $\log_{1.2} 10 = 12.6$ slots.

10.6 Simulation Results

We have built a simulation model using Matlab [16] that allows us to compare the performance of the Impatient Backoff Algorithm (Section 10.3) with a slotted exponential backoff algorithm (EBA). EBA chooses a backoff uniformly in a given range, starting with $[0, 32]$. The node with the smallest backoff in its neighborhood is able to transmit in that slot. A collision doubles the range, while a successful transmission brings it back to the initial range. No changes are made to the backoff when the node is quiet. We evaluate over a wide range of topologies, as we explain below.

10.6.1 Simple topologies

We first evaluate our scheme by simulating simple topologies. The results are summarized in Table 10.6.1. For each topology, we list the mean throughput and the fairness, for both EBA and IBA. As explained earlier, we use Jain's Fairness Index (Section 10.3.5) to compare the two schemes, assuming an optimally fair allocation lets each node get an equal share of the throughput.

	Exponential Backoff (EBA)		Impatient Backoff (IBA)	
Topology	Mean Throughput	Fairness	Mean Throughput	Fairness
Star with 2 arms	0.5498	0.91	0.4878	1
Star with 4 arms	0.5537	0.89	0.4866	1
Star with 10 arms	0.5519	0.90	0.4880	1
3 node clique	0.3201	1	0.3094	1
5 node clique	0.1871	1	0.1745	1
10 node clique	0.0890	1	0.0777	1
Square	0.4792	1	0.4773	1
Pentagon	0.3863	1	0.3832	1
Hexagon	0.4267	1	0.4671	1

Table 10.1. Simulation results for simple topologies

In the star topologies, EBA is unfair, while IBA has a fairness of 1. However, in order to achieve fairness, IBA has to pay a throughput tradeoff. The rest of the topologies are symmetric, so both schemes are fair in such topologies. We observe that the mean throughput of IBA is comparable to that of EBA in all these cases.

10.6.2 Random topologies

A more interesting comparison is visible in random topologies. We first consider a random field of size $4km \times 4km$ and place 100 Ad-Hoc nodes at random on it. The nodes have a transmission range of $500m$ and an interference range of $1km$. Every node is assumed to have packets to send at all times. Note that at any slot, multiple transmission can take place at different parts of the field.

Figs. 10.9 and 10.10 express the simulation results for EBA and IBA, on the same topology. We denote a node by a circle with its center at the node's location. The area

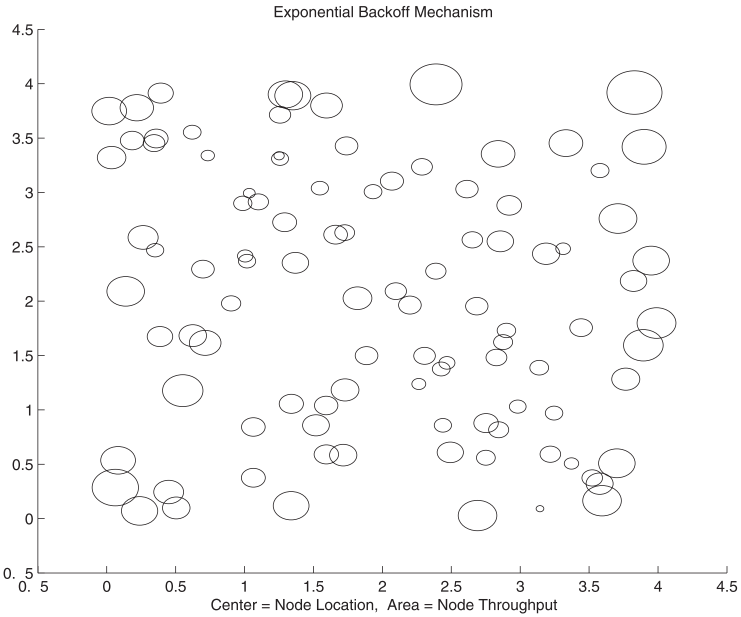


Fig. 10.9. Node throughput in a random topology: Exponential Backoff.

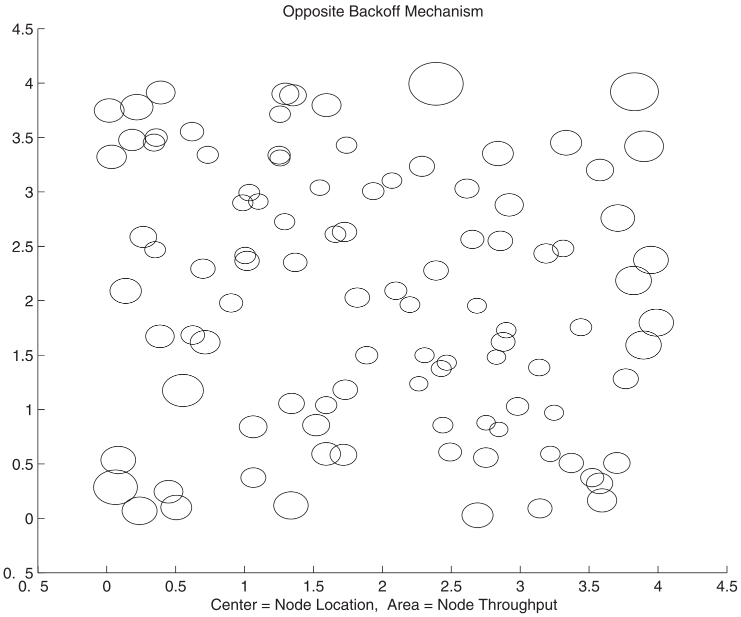


Fig. 10.10. Node throughput in a random topology: Impatient Backoff.

of the circle is proportional to the throughput achieved by the node (i.e., number of successful transmissions) scaled by the total number of slots.

By comparing the two graphs visually, we can qualitatively see the fairer nature of IBA. EBA in Fig. 10.9 includes many nodes with very small throughput – all of whom are able to increase their throughput in IBA. Jain's Fairness Index in the EBA simulation is only 0.66 and it increases to 0.75 for IBA. The lowest throughput achieved by a node in EBA is merely 0.0090, while the lowest throughput in IBA is 0.0490. The mean throughput for the two schemes is comparable, at 0.1066 and 0.1046.

Table 10.6.2 summarizes the results from simulations across several random topologies. In each case, we generated a different random topology and ran the simulations on both backoff schemes. We notice that the mean throughput is comparable, while IBA achieves a significantly higher fairness. Also, the minimum throughput received by a IBA node is 3 to 5 times higher than in EBA.

	Exponential Backoff			Impatient Backoff		
Topology	Mean Thrpt	Min Thrpt	Fairness	Mean Thrpt	Min Thrpt	Fairness
100 nodes; $4 \times 4km^2$	0.1066	0.0090	0.66	0.1046	0.0490	0.75
100 nodes; $4 \times 4km^2$	0.1029	0.0140	0.71	0.1001	0.0440	0.82
100 nodes; $4 \times 4km^2$	0.1032	0.0120	0.71	0.1031	0.0380	0.81
50 nodes; $2.5 \times 2.5km^2$	0.0842	0.0130	0.71	0.0839	0.0490	0.83
50 nodes; $2.5 \times 2.5km^2$	0.0963	0.0220	0.68	0.0903	0.0430	0.74
25 nodes; $2 \times 2km^2$	0.1188	0.0040	0.69	0.1004	0.0590	0.87

Table 10.2. Simulation results for random topologies

10.6.3 Reset propagation and lost resets

As explained in Section 10.3.4, IBA requires all nodes to reset their mean backoff delays when *any* mean backoff goes below a reset limit. In a practical situation, this requires the propagation of the reset message through the network. We model this in our simulations by propagating the reset message hop by hop from the originating node. A node resets its backoff only when it receives a reset message. The reset messages have a time-to-live field to ensure expiration after a single reset. Also, a node does not reset more than once in a fixed number of slots, thus multiple reset messages starting from different parts of the network at around the same time cause the backoff delays to be reset only once.

Furthermore, reset messages may get lost in the way. Our simulations account for situations where up to 10% of reset messages get lost.

It turns out that reset propagation and loss has marginal effect on the throughput and fairness of IBA. The simulation results given above in Section 10.6.2 in fact take into account both these effects. Qualitatively, we can see why the reset propagation and loss does not affect the long term fairness of the system. A delayed or lost reset message implies that the intended recipient node does not reset its backoff in a timely manner and hence continues to have a low backoff. However, the result of this is that it wins a few contentions unfairly — which in turn causes it to increase its backoff. The unfairness only allows a few extra packet transmissions and does not persist. Thus the scheme is fair in the long run.

10.6.4 Variations in simulation scenarios

We also attempt a few variations in the simulation scenarios. In each of the cases, the throughput and fairness numbers are comparable to the previous cases, hence we do not repeat them explicitly.

Movement: We want to investigate the effect of random walk movement on IBA. Every node moves randomly in discrete steps every 10 time slots.

Changing Number of Nodes: We also consider the case when nodes do not always have packets to send. This is simulated by considering blocks of slots when a node is either awake or asleep. For simple topologies, we evaluate the fairness of IBA against the fair share

of bandwidth based on the active nodes during each block of slots. The results show IBA to maintain its fairness even with a changing number of neighbors.

An example is shown in Table 10.6.4 for a clique of five nodes. Different nodes have a different fair share since they are active for varying amounts of time, but the IBA scheme is able to provide them with that share. These experiments yield a fairness of 1.

Node No.	Throughput	Fair Share
1	0.2040	0.2040
2	0.2220	0.2217
3	0.1330	0.1340
4	0.1710	0.1717
5	0.2080	0.2073

Table 10.3. IBA maintains fairness even when not all nodes are transmitting

Biased Initial Backoff Rate: We also extend the idea of an initial bias (Section 10.4.3) to a general topology. We start with an initial bias against a few selected nodes by making their initial mean backoff 64 times other mean backoff delays, and observe whether these nodes indeed get a fair share of the throughput. Under IBA, these backoff delays soon catch up and the nodes subsequently get an equal share of the bandwidth. While their initial backoff delays are high, these nodes tend to lose the contention, causing their mean backoff delays to be divided by m. Thus they catch up in approximately $\log_m 64$ slots.

10.7 Conclusions

Traditional distributed Ad-Hoc MAC protocols (e.g., 802.11b) use random backoff delays to avoid collisions and share bandwidth amongst contending nodes. The fundamental backoff rule is to become less aggressive upon collision. We observe that in a network spanning multiple interference domains, this approach leads to unfairness towards nodes in the middle of the network.

We propose a novel backoff scheme that counters the conventional backoff wisdom. Nodes in our Impatient Backoff Algorithm *decrease* their backoff when they collide or are unable to send — thereby becoming *more* aggressive. This approach tends to help nodes with more neighbors and leads to a fairer allocation of bandwidth. The danger of the system becoming unstable due to frequent collisions is handled by resetting the mean backoff delays when they get too low.

We use Markov chains to analyze IBA in simple topologies. We look at two extreme topologies — the unfair star topology, and the symmetric clique topology. By proving positive recurrence of the system, we show that IBA is indeed stable. We are also able to demonstrate fairness of the system, even as the number of nodes in the topology is changing dynamically.

We compare the performance of IBA with an idealized slotted exponential backoff scheme. Results show that in simple topologies IBA is always able to achieve a fairness index of 1, albeit at the cost of some throughput tradeoff. In a random topology, IBA maintains the same mean throughput as EBA but has a significantly higher fairness index. Further extensions involving movements and nodes that switch between active and sleep phases also give similar results.

10.8 Extensions and Future Work

Clearly, the MAC model outlined in Section 10.3 is an idealized version that ignores many practical effects. The most important of these is the slotted nature of the protocol. It would be most interesting to develop an un-slotted version of the protocol, which would allow different packet lengths and also remove the need for the nodes to be synchronized. An RTS/CTS mechanism for IBA will also be useful to solve the hidden terminal problem.

IBA is by no means a complete MAC protocol. It is however a radically different approach to backoff mechanisms in distributed Ad-Hoc networks. This chapter shows the benefits of the novel scheme, and also suggests mechanisms to make IBA practicable. The hope is for follow up work in this space to make this idea a reality.

Acknowledgements. This research was stimulated by related work of Pravin Varaiya ([6], [7], [8], [9]). The second author thanks Pravin for thirty years of friendship, inspiration, and shared joy of research. The authors are grateful to Bill Hodge for his help with experiments and to Antonis Dimakis, Mustafa Ergen, John Musacchio, Wilson So, and Teresa Tung for their constructive comments on this work.

References

[1] J. Li, C. Blake, D. S. J. De Couto, H. I. Lee, and R. Morris, Capacity of Ad Hoc Wireless Networks, *Proceedings ACM Mobicom*, Rome, Italy, July 2001.

[2] R. Gupta, J. Musacchio, and J. Walrand, Sufficient rate constraints for QoS flows in ad-hoc networks, *UCB/ERL Technical Memorandum M04/42*, Fall 2004.

[3] IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, P802.11, Nov 1997.

[4] IEEE P802.11: The Working Group for Wireless LANs. http://grouper.ieee.org/groups/802/11/.

[5] G. Bianchi, Performance analysis of the IEEE 802.11 distributed coordination function, *IEEE Journal on Selected Areas in Communications*, **18**(3):535–547, March 2000.

[6] M. Ergen and P. Varaiya, Throughput analysis and admission control for IEEE 802.11a, *ACM-Kluwer MONET Special Issue on WLAN Optimization at the MAC and Network Levels*, to appear.

[7] M. Ergen and P. Varaiya, Throughput formulation and wlan optimization in mixed data rates for IEEE 802.11 DCF mode, *GLOBECOM-CAMAD*, 2004.

[8] M. Ergen, B. Dundar, and P. Varaiya, Throughput analysis of an extended service set in 802.11, *GLOBECOM*, 2004.

[9] S. Coleri, PEDAMACS: Power Efficient and Delay Aware Medium Access Protocol for Sensor Networks, Masters Thesis, Electrical Engineering and Computer Sciences Department, University of California, Berkeley, December 2002.

[10] D. Bertsekas and R. Gallagher, *Data Networks* (2nd Edition), Prentice Hall, 1991.

[11] C. Yuan and P. Marbach, Rate Control in Random Access Networks, preprint.

[12] L. Kleinrock and F. Tobagi, Packet switching in radio channels, Part II: The hidden terminal problem in carrier sense multiple access and the busy tone solution, *IEEE Transactions in Communications*, **COM-23**(12):1417-1433, Dec 1975.

[13] F.P. Kelly, A. Maullo, and D. Tan, Rate control in communication networks: Shadow prices, proportional fairness and stability, *Journal of the Operational Research Society*, **49**:237-252, March 1998.

[14] D. Chiu and R. Jain, Analysis of the increase and decrease algorithms for congestion avoidance in computer networks, *Computer Networks and ISDN Systems*, **17**:1–14, 1989.

[15] A.G. Pakes, Some conditions of ergodicity and reurrence of markov chains, *Operations Research*, **17**:1058–1061, 1969.

[16] Matlab Simulation Environment, The Mathworks Inc., http://www.mathworks.com.

11

Cooperation, Trust and Games in Wireless Networks

John S. Baras and Tao Jiang

Institute for Systems Research and
Electrical and Computer Engineering Department
University of Maryland
College Park, MD 20742
baras@isr.umd.edu, tjiang@glue.umd.edu

11.1 Introduction

As an important concept in network security, trust is interpreted as a set of relations among agents participating in the network activities. Trust relations are based on the previous behavior of an agent within a protocol. Trust establishment in distributed and resource-constraint networks, such as mobile ad hoc networks (MANETs), sensor networks and ubiquitous computing systems, is much more difficult but more crucial than in traditional hierarchical architectures, such as the Internet and base station- or access point-centered wireless LANs. Generally, this type of distributed networks have neither pre-established infrastructures, nor centralized control servers or trusted third parties (TTP). The trust information or evidence used to evaluate trustworthiness is provided by peers, i.e. the agents that form the network. Furthermore, resources (power, bandwidth, computation etc.) are normally limited because of the wireless and ad hoc environment, so the trust evaluation procedure should only rely on local information. Schemes that depend only on local interaction also have the desired emergent property that enables fast reaction to network member changes, topology changes and security changes that frequently happen in mobile networks. Therefore, the essential and unique properties of trust management in this new paradigm of wireless networking, as opposed to traditional centralized approaches are: **uncertainty and incompleteness** of trust evidence, trust value is between -1 and 1; **locality** in trust information exchange; **distributed computation**.

Trust establishment is a process starting from a small set of agents who are known to be trustworthy. For example, the first few peers to join a network are often known to be trustworthy, while the majority are neutral, i.e. with trust value 0. They are subsequently evaluated by agents who have direct interaction with them. Those evaluating agents are either the physical or logical neighbors of target agents. Based on their observations and evidence, they are able to provide opinions on the target agent, to build the trust value (also called reputation) of the target agent. The whole network therefore evolves as the local interactions iterate from “isolated trust islands” to “a connected trust graph." Our interest is to discover rules and policies that establish trust-connected networks using only local interactions, to understand the impact of local interactions on the whole network and also to find the conditions under which trust spreads to a maximum set, as well as the parameters that speed up or slow down this transition.

There have been several works on trust computation based on interactions with one-hop physical neighbors. In [2], for instance, first-hand observations are exchanged between neighboring nodes, where node A adjusts his opinion for B, based on how close B's evidence is to A's previous opinion about another node C. It provides an innovative model to link nodes' trustworthiness with the quality of the evidence they provide. Our work emphasizes the inference of trust value instead of generating the direct trust, which is similar to [7] and [8], where weighted averages were used to aggregate multiple votes for trust evaluation and provided promising results on using this simple local interaction rule to correctly evaluate trust in distributed networks. Particularly in [7], different kinds of malicious behaviors have been simulated and their results showed that by ranking nodes according to the trust value, the network application (in their case, file downloading in p2p networks) doesn't get affected by malicious nodes. However, the results in both [7] and [8] are based on simulation. In this chapter, we analyze a local interaction rule using graph theory and provide a theoretical justification for network management that facilitates trust propagation.

In wireless networks such as mobile ad hoc networks and sensor networks, most of the functions (routing, mobility management, and security) must rely on cooperation between nodes. In addition, such cooperation utilizes local information and local (between neighbors) interactions. This is probably the most important difference between this type of networks and traditional networks, such as the Internet and cellular networks.

In the wireless networks of interest in this chapter, nodes are not under the control of any central authority. In other words, each node is its own authority. The network is generated in a more distributed and asynchronous manner. In this situation, the most reasonable assumption is that each node will try to maximize its benefit by exploiting the network, even if this means adopting a selfish behavior. This selfishness means that nodes are not willing to participate, without additional incentives, in the common networking functions, such as route discovery, packet forwarding and security management, which are always resources consuming, including power batteries and bandwidth consumption.

Over the last few years, there has been an increasing amount of research on designing mechanisms to encourage nodes to collaborate. Basically, the approaches taken can be divided into two categories: one is based on incentive techniques, which normally rely on various kinds of trust or reputation systems to promote cooperation and circumvent misbehaving nodes [2, 3, 9]; the other is inspired from game theory, where payoffs are assigned to different strategies of nodes, and Nash equilibria in non-cooperative games are considered to be the optimal and stable solutions [5, 13].

In our chapter, the interactions among nodes are also modeled as games, which are cooperative games rather than non-cooperative games, where players always conflict. In cooperative games, players form coalitions to obtain the optimum payoffs. The key assumption that distinguishes cooperative game theory from noncooperative game theory is that players can negotiate effectively [10]. We will discuss how negotiation can help to form the *grand coalition* that includes all players together. Another way to form a grand coalition is through a trust establishment mechanism: nodes which do not cooperate will be penalized by the trust establishment mechanism. How trust establishment mechanisms can help in cooperative games is also analyzed. Furthermore, we show that trust establishment and evolution of cooperation go hand in hand by viewing the whole network as a distributed dynamical system.

As discussed, trust computation is distributed and restricted to only local interactions in a MANET. Each node, as an autonomous agent, makes the decision on trust evaluation individually. The decision is based on information it has obtained by itself or from its neighbors. Those aspects are analogous to situations in statistical mechanics of complex systems with game theoretic interactions. Game theory, and more specifically the theory of evolutionary games, provide the framework for modeling individual interactions. This circle of ideas has a lot in common with randomized optimization methods from statistical physics.

One of the simplest local interaction models is the Ising model [11], which describes the interaction of magnetic moments or spins, where some spins seek to align with one another (ferromagnetism), while others try to anti-align (antiferromagnetism). The Ising spin model consists of n spins. Each spin is either in position "up" or "down." Any configuration of spins is denoted as $\mathrm{s} = \{s_1, s_2, \ldots, s_n\}$, where $s_i = 1$ or -1 indicating spin i is up or down respectively. A Hamiltonian, or energy, for a configuration s is given by

$$H(\mathrm{s}) = -\frac{1}{T} \sum_{\forall i \in V, j \in N_i} J_{ij} s_i s_j - \frac{mH}{T} \sum_i s_i \tag{11.1}$$

where T is the temperature. The first term represents the interaction between spins. The second term represents the effect of the external (applied) magnetic field. In the Ising model the local interaction "strengths" are all equal to a parameter J. In the more complex case of spin glass the J_{ij} are different and may even come from random processes [11].

The problem of computing the ground state (global minimum of energy) for the Ising model (and even more so for spin glasses) is an NP-hard problem. There are 2^n possible configurations for the model, the computation becomes infeasible when n gets large. So we must use heuristic methods to find low energy configurations. As proposed in [1], we could imagine that the spins try to reduce their own *frustration* (or energy) individually, and come up with an interesting cooperative game. In game theoretic terms, the *payoff* for node i, when the graph has a configuration $\mathrm{s} = \{s_1, s_2, \ldots, s_n\}$, is

$$\pi_i = \sum_{j \in N_i} J_{ij} s_i s_j . \tag{11.2}$$

When $J_{ij} = 1$, the agents are rewarded for aligning their spin states; when $J_{ij} = -1$ they want to take on opposite states (anti-align their spins) in order to maximize their payoffs. Agents interact in order to maximize their own payoffs.

This model provides the inspiration for our approach, as it can be directly used for distributed trust computation. Let s_i be the trust value assigned to node i, where $s_i \in \{-1, 1\}$. Node i will be assigned a trust value according to the opinion of the majority of its neighbors. We set $J_{ij} = 1, \forall j \in N_i$. Then the payoff of agent i is $\pi_i = s_i \sum_{j \in N_i} s_j$. In order to maximize π_i, i will set s_i with the same sign as $\sum_{j \in N_i} s_j$, which is actually the same value as neighbor majority vote. Simulations using Simulated Annealing (SA) show that the average payoff of the whole network is a function of the temperature T in the Ising model. High temperatures, in the trust computation context, mean that the agents are very conservative and not willing to change their trust values, the payoffs are near 0, which is the expected payoff for a random set of s_i from $\{-1, 1\}$. While, as the temperature decreases (aggressive agents), the algorithm becomes greedier and payoffs increase, most of the nodes will reach agreement. Recently there has been very strong interest in the application and extension of such optimization schemes from the statistical mechanics of spin glasses and associated games to optimization and other problems in information technology [11].

In the Ising model, and the more complex models of spin glasses, an important characteristic is *phase transition phenomena*. It is observed that when the temperature is high, all the spins behave nearly independently (no long-range correlation), whereas when temperature is below a *critical temperature* c_0, all the spins tend to stay the same (i.e., cooperative behavior). Phase transitions are also studied in evolutionary prisoner's dilemma games [14]. Phase transition is a common phenomenon that takes place in any combinatorial structure, where a large combinatorial structure can be modeled as a system consisting of many locally interacting components. A phase transition corresponds to a change in some global (macroscopic) parameter of the system as the local parameters are varied. Distributed trust computation is essentially a cooperative game where nodes interact with their neighbors locally.

The structure of the chapter is as follows. In Section 11.2 we develop the network model and the framework of cooperative games for analyzing cooperation among the agents. In Section 11.3 we analyze the cooperative game framework and show that agent cooperation can be achieved employing negotiations between the agents. We also develop a dynamic distributed trust mechanism framework and demonstrate that it can also induce cooperation among agents, albeit without negotiations. In Section 11.4 we investigate the dynamic evolution of both cooperative games and trust mechanisms and establish certain quantitative measures and characteristics of the "spread" of cooperative behavior among agents. Finally, Section 11.5 contains our conclusions and a brief description of future research directions.

11.2 Problem formulation

11.2.1 System model

The network is modeled as an undirected graph $G(V, E)$. Throughout this chapter, we use the terms *node*, *player* and *agent* interchangeably, where a node i is an element in the set V. Nodes are players that play games among themselves. Since we only consider direct interaction among nodes, nodes only play games with their neighbors, which are denoted as:

$$\mathcal{N}_i \triangleq \{j|(i,j) \in E\} \subseteq \{1,\ldots,N\} \setminus \{i\}.$$

The neighbor set of agent i, $\mathcal{N}_i$, can represent the set of agents with which i is allowed to communicate (giving rise to a logical interconnection network), or the set of agents which i can sense, transit or receive information from (physical wireless communication links).

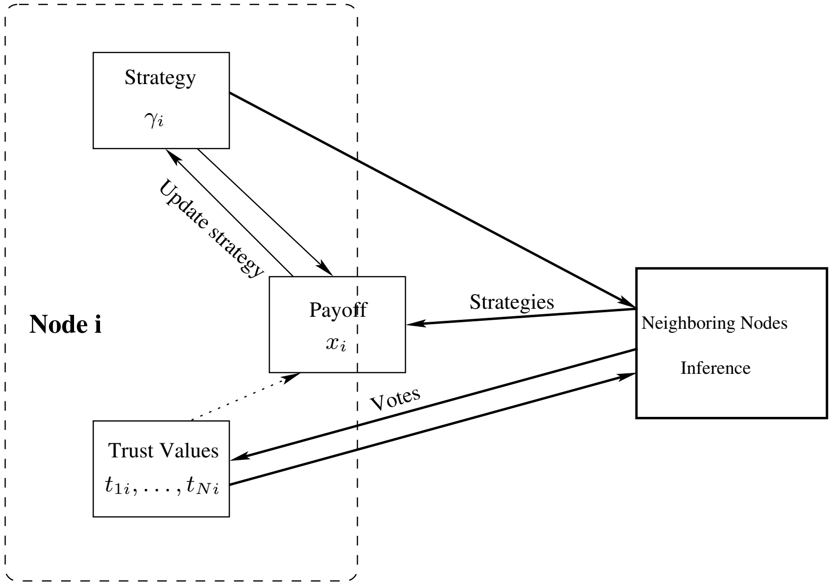


Fig. 11.1. System operation block-graph for a typical node.

In our model, each node has a self-defined playing strategy, which is denoted by γ_i for node i. Another characteristic of each node is its trust values, which are dependent on the opinions of other nodes. Trust values of a node can be different for different players. For instance, t_{ji} and t_{ki} are the trust values of i provided by distinct player j and k, and possibly $t_{ji} \neq t_{ki}$. Fig. 11.1 is a block graph demonstrating how nodes interact among their neighbors, where the payoff of node i after playing games is represented as x_i. The procedure is summarized as the following three rules:

- Strategy updating rule: as shown in Fig. 11.1, nodes update strategies based on their own payoffs. They tend to choose rules that obtain the maximum payoffs.
- Payoff function: the payoffs are functions of the strategies of all participants. For a specific node, the payoff only depends on strategies of its neighbors and itself.
- Trust computation rule: trust values are computed based on votes, which are provided by neighbors and are related to the history (strategies and trust values) of the target node. Since trust values eventually have impact on the payoff of the node, there is a dotted line in Fig. 11.1 from trust values to payoff to represent their implicit relation.

For simplicity, we assume the system is memoryless. All values are dependent only on parameter values at most one time step in the past. Therefore, the system can be modeled as a discrete-time system:

$$\gamma_i(t+1) = f^i(x_i(t), \gamma_i(t), \gamma_j(t), t_{ij}(t)) \tag{11.3}$$

$$t_{ik}(t) = g^i(t_{ij}(t), v_{jk}(t)) \quad \forall k \in N \tag{11.4}$$

$$x_i(t) = h^i(\gamma_i(t), \gamma_j(t)) \tag{11.5}$$

$$v_{ij}(t) = p^i(\gamma_j(t), t_{ji}(t)) \tag{11.6}$$

where j stands for all neighbors of i, and v_{ij} is the value node i votes for j. In Section 11.4, we will analyze the dynamics of the system, especially the effect of trust propagation on the formation of cooperation. We first introduce the basic element of this system: the cooperative games among neighboring nodes.

11.2.2 Games

In this part, we give the formal definitions of the interaction games. In our work, we consider two-person games with perfect information, say, player (or node) P_1 interacts with player (or node) P_2.

Definition 1 (Strategy). *A strategy γ_i for P_i is the alternative P_i chooses based on the information it currently holds. The set of all strategies of P_i is called his strategy set (space), and it is denoted by Γ_i.*

Definition 2 (Payoff). *The payoff of player P_i is the function of the strategies of both players, which is denoted by $x_i = f_i(\gamma_1, \gamma_2)$.*

In a game, two rational players choose their strategies based on the information they have, and aim to achieve the optimum payoff. Games are generally divided into two categories: non-cooperative games and cooperative games. The essential difference of these two types of games is that in cooperative games players are allowed to negotiate while in non-cooperative games players play the game for their own sake. Therefore, in cooperative games, correlated mixed strategies are allowed, and the payoff can be transferred from one player to the other (though not always linearly). In what follows we will compare two different games by providing simple example games; our game model is based on a simple cooperative game and the interactions among neighbors.

Non-cooperative vs. cooperative games

One of the most well-known models in two-player non-cooperative games is the prisoner's dilemma. In the prisoner's dilemma, the strategy sets of both players are $\Gamma_i = \{\text{cooperate, defect}\}$. Then there are four combinations for (γ_1, γ_2) and the payoffs of two players are assigned in a matrix form as shown in Table 11.1, where

		P_1	
		C	D
P_2	C	(r, r)	(s, t)
	D	(t, s)	(p, p)

Table 11.1. Payoff matrix of prisoner's dilemma.

"C" stands for cooperate and "D" for defect. The payoffs are related to whether players cooperate or not and to what extent. For each possible pair of strategies, r is the "reward" payoff that each player receives if both cooperate, p is the "punishment" that each receives if both defect, t is the "temptation" that each receives if he alone defects and s is the "sucker's" payoff that he receives if he alone cooperates. The payoffs satisfy the following chain of inequalities:

$$t > r > p > s.$$

Players try to maximize their payoffs. For player P_1, strategy D is *strictly dominant* to the strategy C: whatever his opponent does, he is better off choosing D than C. By symmetry, D also strictly dominates C for player P_2. Thus two "rational" players will defect and receive a payoff of p, while two "irrational" players can cooperate and receive greater payoff r.

In cooperative games, players are allowed to negotiate and use the strategies according to their committed agreement. Under such an assumption, rational players either cooperate at the same time or defect simultaneously. If two players do not cooperate, the payoff they get is called the *disagreement vector* $\mathbf{f}^* \in \mathbf{R}^2$. If they cooperate, the players negotiate about which point in the set of feasible payoffs $L \in \mathbf{R}^2$ they will agree upon. So in cooperative games we need to investigate: 1) whether players are willing to reach a consensus on which feasible payoff to realize; 2) how to allocate the payoffs among the players. We can analyze a simple cooperative game that is a modification of the prisoner's dilemma: the disagreement vector $\mathbf{f}^* = (p, p)$, for simplicity let $p = 0$ and let the payoffs be defined as

$$\begin{aligned} x_1 &= f(a_2) - ca_1 \\ x_2 &= f(a_1) - ca_2 \\ a_1 + a_2 &\le E \end{aligned}$$

where a_1 and a_2 are some limited resources (with limit E) shared by two players, such as money or bandwidth in the network context, and f be a concave function. Fig. 11.2 depicts an example of the players' payoffs.

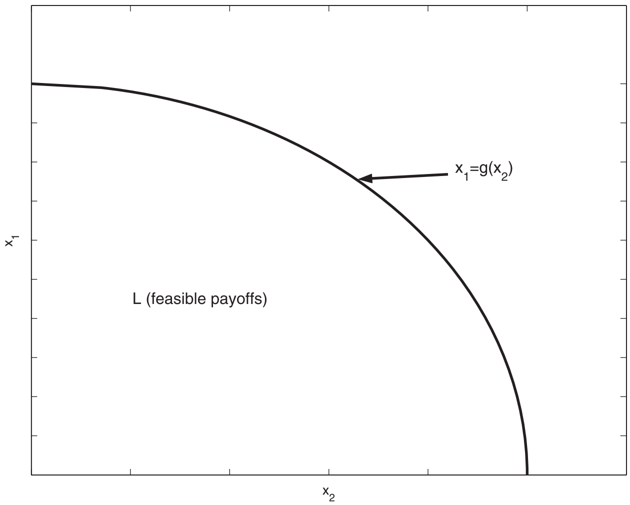


Fig. 11.2. Illustration of a two-player cooperative game.

The negotiation result $\mathbf{x} = (x_1, x_2)$ satisfies the following conditions

1. $\mathbf{x} \in L$ (feasibility);
2. $\mathbf{x} \geq \mathbf{f}^*$ (rationality);
3. $\mathbf{x'} \in L, \mathbf{x'} \geq \mathbf{x}$ imply $\mathbf{x'} = \mathbf{x}$ (Pareto-optimality).

Then the boundary of the compact, convex feasible set $D = L \cap \{\mathbf{x} : \mathbf{x} \geq \mathbf{f}^*\}$, i.e. the curve $x_1 = g(x_2)$ in Fig. 11.2, is the set of candidates for negotiation. Then the question is: on which point the agents would agree on if they cooperate? This will be discussed in Sect. 11.3.

Games on networks

In this chapter, we consider cooperative games on networks, where nodes play cooperative games with their neighbors iteratively. Assume that at each time step, two neighboring nodes only play the game once. Cooperative games are normally represented by the *characteristic function form* which is a finite set $N = \{1, \ldots, N\}$, the set of players and a function (characteristic function) $v : 2^N \rightarrow \mathbf{R}$ defined on all subsets (coalitions) of N with $v(\emptyset) = 0$. We denote such a game as $\Gamma = (N, v)$. Define S, a subset of N, as a coalition if all nodes in S cooperate. Then $v(S)$ is interpreted as the maximum utility (payoff) S can get without the cooperation of the rest of the players $N \setminus S$. In order to simplify our analysis, we assume the payoff only depends on the interacting two parties and the feasible payoff set of the two-player game is shown in Fig. 11.2. Suppose y_{ij} is the payoff of i from the game between i and j. Since games are played on networks, $y_{ij} \neq 0$ only if i and j are neighbors, and set $y_{ij} = 0$ if $i = j$ or i and j are not neighbors. For instance, consider two neighboring nodes i and j and let $S = \{i, j\}$, then

$$v(S) = \max\{y_{ij} + y_{ji}\}. \tag{11.7}$$

Apparently, the payoffs that maximize $v(s)$ are on the Pareto frontier of the convex set L. Substitute $y_{ij} = g(y_{ji})$ into (11.7), and we can derive the payoffs that maximize $v(s)$, denoted as (x_{ij}, x_{ji}). In a geometric interpretation, (x_{ij}, x_{ji}) is the point on the boundary of L from where a tangent to L can be drawn with slope -1. It is obvious that $\mathbf{x} = (x_{ij}, x_{ji})$ satisfies the negotiation conditions.

The following are assumptions made and used in this chapter:

- The games are with transferable utility, i.e., payoffs were given in linearly transferable utility.
- The cooperation is bilateral, i.e. for two neighboring nodes, either both cooperate or none cooperates. This is because there is no incentive for a node to altruistically contribute without receiving some payoff.
- Nodes cooperate with all the neighbors in the same coalition. If i is in coalition S, $j \in \mathcal{N}_i$ and $j \in S$, then i cooperates with j.

As we defined, a coalition is a subset of nodes that cooperate with all their neighbors in the coalition. Among all coalitions, there are so-called *maximum* coalitions which are not subsets of any other coalition, i.e., if S is a maximum coalition, then $\forall i \in S, j \notin S$, i and j do not cooperate with each other. In this chapter, all coalitions are maximum coalitions, so we omit *maximum* from now on. We could easily find the characteristic function of our cooperative game, which is the summation of the payoffs from all cooperative pairs in the coalition, as:

$$v(S) = \sum_{i,j \in S} x_{ij}. \tag{11.8}$$

Notice that $\forall i, v(\{i\}) = 0$. We denote the cooperative game defined from (11.8) as $\Gamma = (N, v)$.

In the next section, we will describe the details of the system model. Based on the model, we will investigate stable solutions for enforcing cooperation among nodes, and demonstrate two efficient methods for achieving such cooperation: negotiation and trust mechanism.

11.3 Cooperation in games

11.3.1 Cooperative games with negotiation

In Section 11.2.2, we reviewed and defined games, especially cooperative games that are used in our interaction model. In this section, we investigate the impact of the games on the collaboration in a network. First we start with a simple fact.

Lemma 1. *If* $\forall i, j,\ x_{ij} + x_{ji} \geq 0$*, then* $\Gamma = (N, v)$ *is a superadditive game.*

Proof. Suppose S and T are two disjoint sets ($S \cap T = \emptyset$), then

$$\begin{aligned} v(S \cup T) &= \sum_{i,j \in S \cup T} x_{ij} = \sum_{i,j \in S} x_{ij} + \sum_{i,j \in T} x_{ij} + \sum_{i \in S, j \in T} (x_{ij} + x_{ji}) \\ &= v(S) + v(T) + \sum_{i \in S, j \in T} (x_{ij} + x_{ji}) \geq v(S) + v(T). \end{aligned}$$

The last inequality holds by our assumption that $x_{ij} + x_{ji} \geq 0$. □

The main concern in cooperative games is how the total payoff from a partial or complete cooperation of the players is divided among the players. A *payoff allocation* is a vector $\mathbf{x} = (x_i)_{i \in N}$ in $\mathbf{R}^N$, where each component x_i is interpreted as the payoff allocated to player i. We say that an allocation $\mathbf{x}$ is *feasible for a coalition S* iff $\sum_{i \in S} x_i \leq v(S)$.

When we think of a reasonable and stable payoff, the first thing that comes to mind is a payoff that would give each coalition at least as much as the coalition could enforce itself without the support of the rest of the players. In this case, players couldn't get better payoffs if they form separate coalitions different from the grand coalition N. The set of all these payoff allocations of the game $\Gamma = (N, v)$ is called the *core* and is formally defined as the set of all n-vectors $\mathbf{x}$ satisfying the linear inequality:

$$\mathbf{x}(S) \geq v(S) \quad \forall S \subset N, \tag{11.9}$$

$$\mathbf{x}(N) = v(N), \tag{11.10}$$

where $\mathbf{x}(S) = \sum_{i \in S} x_i$ for all $S \subset N$. If Γ is a game, we will denote its core by $C(\Gamma)$. It is known that the core is possibly empty. Therefore, it is necessary to discuss existence of the core for the game Γ. We first give the definition of a family of common games: convex games [6]. The convexity of a game can be defined in terms of the *marginal contribution* of each player, which plays the role of first difference of the characteristic function v. Convexity of v can be defined in terms of the monotonicity of its first differences. The first difference (or the marginal contribution of player i) $d_i : 2^N \to \mathbf{R}$ of v with respect to player i is

$$d_i(S) = \begin{cases} v(S \cup \{i\}) - v(S) & \text{if } i \notin S \\ v(S) - v(S \setminus \{i\}) & \text{if } i \in S. \end{cases}$$

A game is said to be *convex*, if for each $i \in N$, $d_i(S) \leq d_i(T)$ holds for any coalition $S \subset T$.

Lemma 2. *$\Gamma(N, v)$ is a convex game.*

Proof. For Γ, $d_i(S) = \sum_{j \in S, j \neq i} (x_{ji} + x_{ij})$. Taking two sets $S \subset T$,

$$d_i(T) - d_i(S) = \sum_{j \in T \cap S^c} (x_{ji} + x_{ij}) \geq 0.$$

□

The core of a convex game is nonempty ([6]), thus $C(\Gamma) \neq \emptyset$. By Lemma 2, we have the following theorem,

Theorem 1. $\Gamma = (N, v)$ *has a nonempty core.*

Now let's find one of the payoff allocations that are in the core. For any pair of players (i, j), suppose the payoff allocation of the game between i and j is $(\hat{x}_{ij}, \hat{x}_{ji})$. Then we have the following

Corollary 1. *If the payoff allocation satisfies* $\hat{x}_{ij} \geq 0$ *and* $\hat{x}_{ji} \geq 0$*, then the payoff allocation* $\hat{x}_i = \sum_{j \in \mathcal{N}_i} \hat{x}_{ij}$ *is in the core* $C(\Gamma)$*.*

Proof. Take an arbitrary subset $S \subset N$,

$$\hat{x}(S) = \sum_{i \in S} \hat{x}_i = \sum_{i,j \in S} \hat{x}_{ij} + \sum_{i \in S, j \notin S} \hat{x}_{ij} \geq \sum_{i,j \in S} \hat{x}_{ij} = v(S);$$

the inequality holds because $\hat{x}_{ij} \geq 0, \forall i, j \in N$. □

Because we only consider transferable utility games, $\hat{x}_{ij} + \hat{x}_{ji} = x_{ij} + x_{ji} \geq 0$. Therefore $(\hat{x}_{ij}, \hat{x}_{ji})$ could be constructed in the following way:

$$\hat{x}_{ij} = \begin{cases} x_{ij} & \text{if } x_{ij} \geq 0, x_{ji} \geq 0 \\ x_{ij} + \lambda_{ij} x_{ji} & \text{if } x_{ij} < 0, x_{ji} > 0 \\ (1 - \lambda_{ij}) x_{ji} & \text{if } x_{ij} > 0, x_{ji} < 0 \end{cases}$$

where $0 \leq \lambda_{ij} = \lambda_{ji} \leq 1$, and $\hat{x}_{ij} \geq 0$ is achieved by carefully choosing λ_{ij}.

Obviously, the payoff allocation we provided in Corollary 1 is a set of points in the core, while there generally exist more points in the core that are not covered in the Corollary. However, this solution indicates a way to encourage cooperation in the whole network. The players that have positive gain can negotiate with their neighbors by sacrificing certain gain (offering their partial gain λx_{ji}). Though they cannot achieve their best possible payoff, they can set up a cooperative relation with their neighbors. This is definitely beneficial for the players who negotiate and sacrifice, since without cooperation they cannot get anything. This solution is also efficient and scalable, because players only need to negotiate with their direct neighbors.

Thus we established cooperative games among nodes in the network, and described an efficient way to achieve cooperation throughout the network. In the next section, we are going to discuss solutions by employing trust mechanisms, which do not require negotiation and the assumption on $x_{ij} + x_{ji} \geq 0$ can also be relaxed.

11.3.2 Trust mechanism

Trust is a useful incentive for encouraging nodes to collaborate. Nodes who refrain from cooperation get lower trust value and will be eventually penalized because other nodes tend to only cooperate with highly trusted ones. From Fig. 11.1 and the corresponding system equations, the trust values of each node will eventually influence its payoff. Let's assume, for node i, that the loss of not cooperating with node j

is a nondecreasing function of x_{ji}, because the more j loses, the more effort j undertakes to reduce the trust value of i. Denote the loss for i being non-cooperative with j as $l_{ij} = f(x_{ji})$ and $f(0) = 0$. For simplicity, assume the characteristic function is a linear combination of the original payoff and the loss, which is shown as

$$v'(S) = \sum_{i,j \in S} x_{ij} - \sum_{i \in S, j \notin S} f(x_{ji}). \tag{11.11}$$

The game with characteristic function v' is denoted as $\Gamma'(N, v')$. We then have

Theorem 2. *If* $\forall i, j,\ x_{ij} + f(x_{ji}) \geq 0$*, then* $C(\Gamma') \neq \emptyset$ *and* $x_i = \sum_{j \in N} x_{ij}$ *is a point in* $C(\Gamma')$*.*

Proof. First we prove Γ' is a convex game, given $x_{ij} + f(x_{ji}) \geq 0$. We have that $\forall i \in N$ in Γ',

$$d_i(S) = \sum_{j \in S, j \neq i} (x_{ij} + x_{ji}) - \sum_{k \notin S} f(x_{ki}) + \sum_{j \in S, j \neq i} f(x_{ij}).$$

Letting $S \subset T$,

$$\begin{aligned} d_i(T) - d_i(S) &= \sum_{j \in T \cap S^c} (x_{ij} + x_{ji}) + \sum_{k \in T \cap S^c} f(x_{ki}) + \sum_{j \in T \cap S^c} f(x_{ij}) \\ &= \sum_{j \in T \cap S^c} \big((x_{ij} + f_(x_{ji})) + (x_{ji} + f(x_{ij}))\big) \geq 0. \end{aligned}$$

Therefore $C(\Gamma')$ is nonempty. Next, we verify that $x_i = \sum_{j \in \mathcal{N}_i} x_{ij}$ is in the core. For any $S \in N$,

$$\begin{aligned} \sum_{i \in S} x_i - v(S) &= \sum_{i \in S} \sum_{j \in N} x_{ij} - \left(\sum_{i,j \in S} x_{ij} - \sum_{i \in S, k \notin S} f(x_{ki}) \right) \\ &= \sum_{i \in S, j \notin S} (x_{ij} + f(x_{ji})) \geq 0. \end{aligned}$$

□

Apparently, the payoff $x_i = \sum_{j \in \mathcal{N}_i} x_{ij}$ does not need any payoff negotiation. Thus we showed that by introducing a trust mechanism, all nodes are induced to collaborate with their neighbors without any negotiation.

In this section, we introduced two approaches that encourage all nodes in the network to cooperate with each other: 1) negotiation among neighbors; 2) trust mechanism. We proved that both approaches lead to a nonempty core for the cooperative game played in the network. However, we have only considered these two approaches separately, and the results are based on static settings. The more interesting problems are how these two intertwine and how the dynamics between the two approaches converge to a cooperative network — these are discussed in the next section.

11.4 Dynamics of cooperation

We have analyzed the effect of a trust mechanism on the formation of cooperation. However, what we concentrated on in Section 11.3.2 is the final impact of trust on the payoffs at the steady state. In this section, we are going to discuss two dynamic behaviors in the system: trust propagation and game evolution.

11.4.1 Trust propagation

Trust propagation is concerned with how trust evidences (usually negative evidences) propagate from the victims (those who do not receive desired services from their neighbors) and how the trust evidences of a certain node reach its neighborhood and trigger off revocation. The consequence of revocation is that the neighbors refuse to cooperate with the poorly-trusted node and finally isolate it.

Our model is motivated by considering a group of agents each of whom must decide between two alternative actions (trust or distrust a certain node), and whose decisions depend explicitly on the actions of other members of the group. Apparently, the other members are those who are interacting with the agent. In economic terms, this entire class of problems is known generically as binary decisions with externalities. Though it appears as a very simple binary decision problem, it is relevant to surprisingly complex problems, such as statistical mechanics. The decision rule in our model is basically a threshold rule. Agents are usually reluctant to switch their decisions, because decisions usually require more resources and time. But once their individual threshold has been reached, even a single evidence can trigger them into switching from one state to another. Our decision rule, which is particularly simple, while capturing the essential features outlined above, is the following. Every node keeps a state that represents its opinion on a particular node, say node i: 0 stands for distrust and 1 stands for trust. Suppose initially all nodes have the state 1, i.e., nodes first trust all others, but the state immediately changes if the node observes non-cooperation of the particular node i. We model the system evolving in discrete time. At each time step, a node observes the current states (either 0 or 1) of other nodes it interacts with, which we call its neighbors. The node adopts state 0 if at least a threshold fraction ϕ of its k neighbors are in state 0, else it adopts state 1.

Because of the differences in knowledge, preferences and observational capabilities across the nodes, the threshold ϕ is allowed to be heterogeneous. ϕ is determined by the individual node, and can be modeled as drawn at random from a pre-defined distribution with pdf f(ϕ). As we have discussed, to model the dynamics of the revocation, the states of all nodes are initially set to 1. At a certain time, the non-cooperative behavior of node i is observed, then a fraction (usually very small, because the network is sparse) of the nodes are switched to state 0. The whole network evolves at successive time steps, with all nodes updating their states in asynchronous order according to the threshold decision rule above. Once a node has switched to state 0, it remains at 0 for the rest of the dynamics.

The main objective of trust propagation is to explore how the trust revocation depends on the network interactions. Because building relationships and exchanging information are both costly, especially for wireless ad hoc networks, the interactions tend to be very sparse, so we consider only the properties of networks with low (node) degree. Our approach concentrates on two quantities: (i) the probability that the revocation is accepted by a sufficiently large portion of the network (or a finite fraction for an infinite network) triggered from a single node (or small fraction of nodes) — we call these phenomena global revocation; and (ii) the expected size of the global revocation.

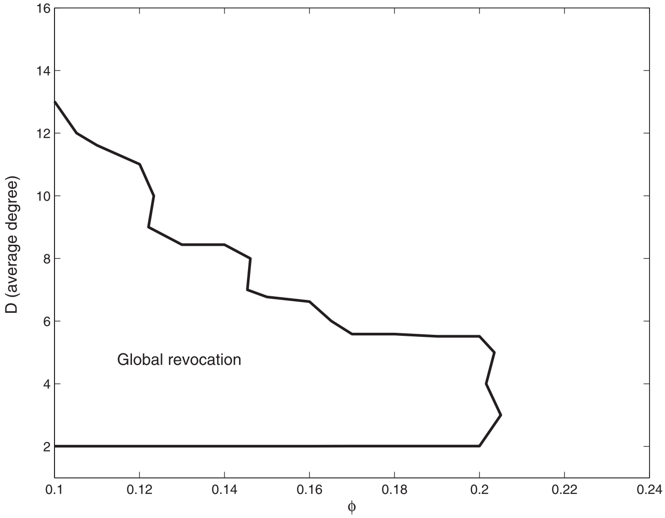


Fig. 11.3. Revocation windows for the threshold model. The network is a random graph based on Erdös and Rényi [4].

Fig. 11.3 graphically shows the condition for global revocation. For simplicity, we assume homogeneity, i.e., the threshold ϕ is the same for every node. The average (node) degree of the network [12] is given by d. The line encloses the region of the (ϕ, d) plane in which a large fraction (80%) of the network nodes accept the revocation. Fig. 11.4 illustrates that the fraction of nodes accepting revocation changes with the threshold ϕ, with fixed average (node) degree.

The phase transitions in Fig. 11.4 define the boundaries of the revocation windows. The exact solutions for the phase transitions are discussed in [15], which also provides the comparison of different network topologies. Therefore, the network topology and threshold value are crucial parameters for global revocation. This gives an important indication and reference for network management and decision control in sparse networks, where agents interact and make decisions based on information provided by their neighbors, and in collaboration with their neighbors.

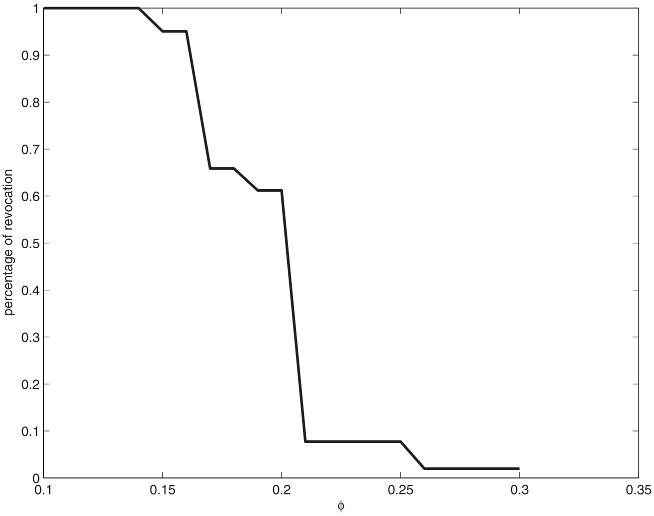


Fig. 11.4. Percentage of nodes accepting revocation vs. threshold ϕ, $d = 6$.

11.4.2 Game evolution

As shown in Sect. 11.3.2, the trust mechanism drives selfish nodes to sacrifice part of their benefits and thus promotes cooperation. In this section, the procedure and dynamics of such cooperation evolution are studied.

In this section, we assume that nodes either cooperate or do not cooperate with neighbors. $\gamma_{ij} = 1$ denotes that node i cooperates with its neighbor j, and $\gamma_{ij} = 0$ denotes that it does not cooperate with j. We assume that the payoff when one of them does not cooperate is fixed as $(0, 0)$, and as (x_{ij}, x_{ji}) when both cooperate. If $x_{ij} < 0$, we call the link (i, j) a *negative link* for node i, and when the opposite holds a *positive link*. Since all nodes are selfish, nodes tend to cooperate with neighbors that are on positive links, while they do not wish to cooperate with neighbors on negative links. Meanwhile, the trust mechanism is employed, which aims to function as the incentive for cooperation. In this part, we assume that revocation and nullification of revocation can propagate throughout the network as discussed in Section 11.4.1.

In our evolution algorithm, each node maintains a record of its past experience by using the variable $\Delta_i(t)$. First define $x_{a,i}(t)$ as the payoff i gains at time t and $x_{e,i}(t)$ as the expected payoff i can get at time t if i always chooses cooperation with all neighbors. Notice that the expected payoff can be different each time, since it depends on whether the neighbors cooperate or not at the specific time. Then compute the cumulative difference,

$$\Delta_i(t) = \Delta_i(t-1) + (x_{a,i}(t) - x_{e,i}(t)) , \qquad (11.12)$$

of the total payoff in the past minus the expected payoff if the node always cooperates. Each node chooses its strategy on the negative links by the following rule:

- if $\Delta_i(t) < 0$, node i chooses to cooperate, i.e., $\gamma_{ij} = 1, \forall j \in \mathcal{N}_i$.
- if $\Delta_i(t) \geq 0$, $\gamma_{ij} = 0$, if $j \in \mathcal{N}_i$ and $x_{ij} < 1$.

Notice that at time 0, $\Delta_i(0) = 0$. That is to say initially all nodes choose not to cooperate on the negative links, since they are inherently selfish. There are two other conditions that force non-cooperation strategies:

- nodes do not cooperate with neighbors that have been revoked.
- nodes do not cooperate with non-cooperative neighbors.

To summarize, as long as one of those aforementioned conditions is satisfied, nodes choose not to cooperate.

Since we allow and encourage nodes to rectify, i.e., to change their strategies from non-cooperation to cooperation, we define a temporal threshold τ in the trust propagation rule. Instead of always keeping 0 once the state is switched to 0, as in Section 11.4.1, we allow the nullification of revocation (switch back to state 1) under the condition that the revocation has been nullified for τ consecutive time steps. τ also represents the penalty for being non-cooperative. τ needs to be large so that the non-cooperative nodes would rather switch to cooperate than get penalized. However, large τ will also reduce the payoff.

The detailed algorithm is shown in Fig. 11.5.

Suppose the total payoff of node i, if every node cooperates, is $x_i = \sum_{j \in \mathcal{N}_i} x_{ij}$. We have the following

Theorem 3. *$\forall i \in N$ and $x_i > 0$, there exists τ_0, such that for a fixed $\tau > \tau_0$:*

1. *The iterated game converges to Nash equilibrium.*
2. *$\Delta_i(t)/t \to 0$ as $t \to \infty$.*
3. *i cooperates with all its neighbors for t large enough.*

Proof. Nodes without negative links, will always cooperate, thus $\Delta_i \equiv 0$. Therefore, we only consider nodes with negative links. First we prove that for t large enough $\Delta_i(t) < 0$. Define for node i, the absolute sum of positive payoffs and negative payoffs as $x_i^{(p)}$ and $x_i^{(n)}$ respectively. Then

$$x_i = x_i^{(p)} - x_i^{(n)}.$$

Therefore the first payoff for node i is $x_{a,i}(1) = x_i^{(p)} > 0$ and $\Delta_i = x_i^{(n)}$. Define T_{max} as the maximum propagation delay in the network. Then at $t = T_{max}$ all i's neighbors revoke i because at time $t = 1$, i didn't cooperate, and the payoff now is $x_{a,i}(T_{max}) = 0$ and $\Delta_i(T_{max}) = \Delta_i(T_{max} - 1) - x_i$. i continues to get 0 payoff till all neighboring nodes have used the penalty interval τ. It's easy to show that as τ is set large enough, i eventually gets negative Δ_i.

If i follows the strategy rules in Fig. 11.5, i starts to cooperate with all neighbors. The difference of the actual payoff and expected payoff is 0 from then on. Therefore $\Delta_i(t)/t \to 0$ as $t \to \infty$.

Assume node i deviates to non-cooperation, then it will get negative cumulative payoff difference as discussed above. So node i has no intention to deviate from

Consider node i, and the initial settings are as follows:

- all the trust states are set to $s_{ij} = 1, \forall j \in N$;
- the variable $\Delta_i(0) = 0$.

Node i chooses strategies and updates variables in each time step for $t = 1, 2, \ldots$:

1. The strategy on the game with neighbor j is set according to the following rule:
    - for negative links ($x_{ij} < 0$), choose non-cooperation strategy ($\gamma_{ij} = 0$) if $\Delta_i(t-1) \geq 0$;
    - if $s_{ij} = 0$, $\gamma_{ij} = 0$;
    - for all neighbors, $\gamma_{ij} = 0$ iff $\gamma_{ji} = 0$ (cooperation is bilateral);
    - otherwise $\gamma_{ij} = 1$.
2. For all $j \in \mathcal{N}_i$, update the trust state s_{ij} if one of the following three conditions is satisfied, otherwise keep the previous state
    - if i accepts a revocation on node j, $s_{ij} = 0$;
    - if the revocation has been nullified for more than τ consecutive steps, set $s_{ij} = 1$;
    - if $\gamma_{ji} = 0$, set $s_{ij} = 0$;
3. Compute the actual payoff $x_{a,i}(t)$ and expected payoff $x_{e,i}(t)$, then get the cumulative difference $\Delta_i(t)$ by Eqn.(11.12).

Fig. 11.5. Algorithm for game evolution modeling trust revocation.

cooperation. Therefore the game converges to its Nash equilibrium with all nodes cooperating. □

We have also performed simulation experiments with our evolution algorithm. In the simulations, we didn't assume the condition that $\forall i,\ x_i > 0$, instead the percentage of negative links is the simulation parameter. We can report that without this condition, our iterated game with the trust scheme can still achieve very good performance. Fig. 11.6 shows that cooperation is highly promoted under the trust mechanism. In Fig. 11.7, the average payoffs between the algorithm with strategy update and without strategy update are compared, which explains the reason why nodes converge to cooperation.

11.5 Conclusions and Future Directions

In this chapter we investigated fundamental methods by which collaboration in infrastructure-less wireless networks with mobile nodes can be induced, analyzed and evaluated. In this chapter we have also described a new framework within which the problem of distributed trust establishment and maintenance in a mobile ad hoc network (MANET) can be formulated and analyzed.

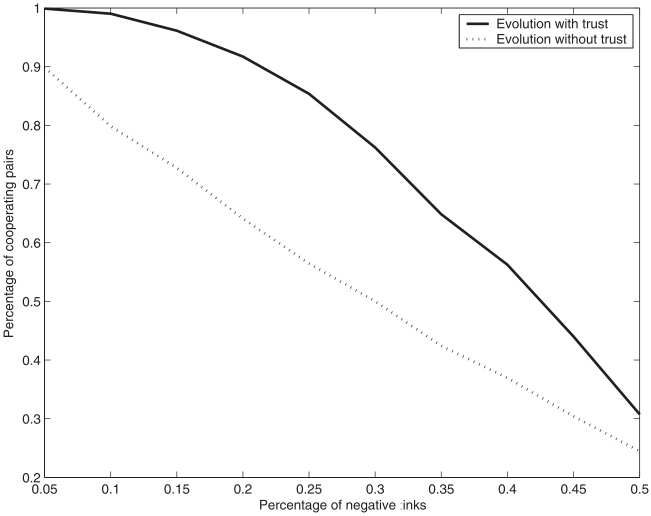


Fig. 11.6. Percentage of cooperating pairs vs. negative links.

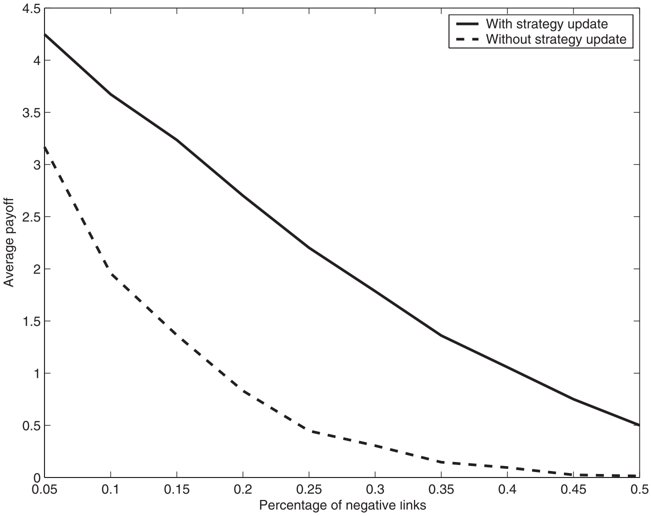


Fig. 11.7. Average payoffs vs. negative links.

We concentrated only on distributed methods that use local interactions. We developed and analyzed a cooperative game framework first and demonstrated how collaboration can be induced. We showed that negotiation between the mobile agents is an important component for achieving collaboration within this framework. We next developed a model for establishing, propagating and managing trust within a MANET. We showed that such trust mechanisms can also establish collaboration, even without negotiations between the mobile agents. Finally we investigated both the dynamics of games as well as of trust propagation as a means for quantifying the degree of collaboration achieved among the agents and of the speed by which this collaboration spreads in a large part of the network agents. In the context of our research reported here, we have drawn inspiration from analytical methods used in statistical mechanics investigations of the Ising model and spin glasses. these analogies include the existence and investigation of phenomena analogous to phase transitions.

Important current and future directions of our research program are the evaluation of the robustness of these mechanisms for collaboration in wireless networks, analysis of their reliability and identification of parameters (including topology types) that influence the dynamics and the qualities of the induced collaborative behavior.

Acknowledgment

Prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. Research also supported by a CIP URI grant from the U.S. Army Research Office under grant No DAAD19-01-1-0494. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

References

[1] J.N. Bearden, The spin glass bead game, Tech. Rept., L.L Thurstone Psychometric Laboratory, The University of North Carolina at Chapel Hill, 2000.

[2] S. Buchegger and J.Y.L. Boudec, The effect of rumor spreading in reputation systems for mobile ad-hoc networks, In: *Proceedings of Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt)*, Sophia-Antipolis, France, 2003.

[3] L. Buttyan and J.P. Hubaux, Stimulating cooperation in self-organizing mobile ad hoc networks, *ACM/Kluwer Mobile Networks and Applications*, **8**:5, 2003.

[4] P. Erdös and A. Rényi, On random graphs I. *Publ. Math.*, 290–297, 1959.

[5] M. Felegyhazi, L. Buttyan and J.P. Hubaux, Equilibrium analysis of packet forwarding strategies in wireless ad hoc networks – the static case. In: *Proceedings of Personal Wireless Communications (PWC '03)*, Venice, Italy, 2003.

[6] F. Forgo, J. Szep and F. Szidarovszky, *Introduction to the Theory of Games: Concepts, Methods, Applications*. Kluwer Academic Publishers, 1999.

[7] S.D. Kamvar, M.T. Schlosser and H. Garcia-Molina, The eigentrust algorithm for reputation management in p2p networks, In: *Proceedings of the Twelfth International World Wide Web Conference*, 640–651, Budapest, Hungary, 2003.

[8] S. Marti and H. Garcia-Molina, Limited reputation sharing in p2p systems, In: *Proceedings of the 5th ACM Conference on Electronic Commerce*, 91–101, ACM Press, New York, USA, 2004.

[9] S. Marti, T.J. Giuli, K. Lai and M. Baker, Mitigating routing misbehavior in mobile ad hoc networks, In: *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, 255–265, ACM Press, Boston, MA, USA, 2000.

[10] R.B. Myerson, *Game Theory: Analysis of Conflict*, Harvard University Press, 1991.

[11] H. Nishimori, *Statistical Physics of Spin Glasses and Information Processing: An Introduction*, Oxford University Press, 2001.

[12] J. Spencer, *The Strange Logic of Random Graphs*, Springer, 2001.

[13] V. Srinivasan, P. Nuggehalli, C.F. Chiasserini and R.R. Rao, Cooperation in wireless ad hoc networks, In: *Proceedings of IEEE INFOCOM*, San Francisco, CA, 2003.

[14] G. Szabo and C. Hauert, Evolutionary prisonner's dilemma games with voluntary participation, *Phys. Rev. E*, Stat. Nolin. Soft Matter Phys, **66** (6), 2002.

[15] D.J. Watts, A simple model of global cascades on random networks, In: *Proceedings of the National Academy of Sciences*, **99** (9), 2002.

[16] D.J. Watts, *Small Worlds: The Dynamics of Networks Between Order and Randomness*, Princeton University Press, 2004.

12

A Game Theoretic View of Efficiency Loss in Resource Allocation

Ramesh Johari[1] and John N. Tsitsiklis[2]

[1] Stanford University, Stanford, CA
ramesh.johari@stanford.edu
[2] MIT, Cambridge, MA
jnt@mit.edu

Summary. Motivated by resource allocation problems in communication networks as well as power systems, we consider the design of market mechanisms for such settings which are robust to gaming behavior by market participants. Recent results in this work are reviewed, including: (1) efficiency loss guarantees for a data rate allocation mechanism first proposed by Kelly, both when link capacities are fixed and when they are elastic; (2) characterization of mechanisms that minimize the efficiency loss, within a certain class of "simple" mechanisms; (3) extensions to general networks; and (4) mechanism design for supply function bidding in electric power systems.

12.1 Introduction

This chapter addresses a problem at the nexus of engineering, computer science, and economics: in large scale, decentralized systems, how can we efficiently allocate scarce resources among competing interests? On one hand, constraints are imposed on the system designer by the inherent architecture of any large scale system. These constraints are counterbalanced by the need to design mechanisms that efficiently allocate resources, even when the system is being used by participants who have only their own interests at stake.

We consider two main classes of resource allocation problems. First, we consider a setting where a resource in scarce supply must be allocated among multiple competing consumers. Second, we discuss a setting where multiple producers compete to satisfy a fixed demand. The former model is motivated by applications to communication networks, while the latter is motivated by electric power market design.

What goals might we have for markets in such settings? We would of course like the equilibria of mechanisms designed for such settings to be "desirable;" a common requirement is that equilibria should be Pareto efficient. In other contexts, we want the equilibria to satisfy a predetermined notion of fairness; or we may wish the resulting vector of monetary transfers to satisfy certain properties, such as profit maximization for the market operator. Beyond such constraints on the properties at equilibrium, however, we are also concerned with the complexity of such mechanisms. In particular, we may desire mechanisms which have relatively low information overhead: the strategy spaces of the players should be "simple," and the feedback from the market to the players should be "simple" as well. Often, such complexity issues arise in a discussion of the dynamic behavior of market mechanisms, in trying to determine whether equilibria are actually achieved over time by players.

In this chapter, we will focus on efficiency of mechanisms which maintain low complexity, appropriately defined. We focus on efficiency primarily as a first test of feasibility. Traditionally, economics has focused on selection of efficient mechanisms because mechanisms with inefficient equilibria are less likely to be useful in practice. Indeed, the classical theory of mechanism design is largely devoted to determining when fully efficient equilibria can be guaranteed (see, e.g., Chapter 23 of [23] for an overview).

The landmark contribution of mechanism design is the Vickrey–Clarke–Groves class of mechanisms, which guarantee efficient allocations at dominant strategy equilibria [4, 11, 32]; unfortunately, implementing VCG mechanisms is generally a very complex proposition with many possible pitfalls [2, 27]. The task is further complicated by the fact that the VCG class of mechanisms are essentially the only class which guarantee fully efficient outcomes as dominant strategy equilibria [8]. Thus, to make progress, the notion of equilibrium must be weakened, and/or some efficiency must be lost. Previous results in the economics literature have considered weakening the notion of equilibrium; for example, Maskin has shown that if we only consider Nash equilibria, efficiency can be guaranteed if certain conditions are satisfied by players' characteristics [24]. However, no guidance is available as to how to design such mechanisms with low complexity.

In this chapter we weaken the requirement of full efficiency. The basic technique we consider is one of restricting the strategy spaces of the players (either buyers or sellers). With the proper choice of restriction, we can achieve two goals simultaneously. First, by ensuring that strategy spaces are relatively simple, we can restrict attention to mechanisms with low complexity. Second, if strategies of players are restricted, we can reduce their opportunities to game the system; this will lead to provable bounds on efficiency loss at Nash equilibria.

In the remainder of the chapter, we provide an overview of the progress made in our earlier work [15–17]. In Section 12.2, we consider a setting of multiple consumers and inelastic supply, motivated by rate allocation in communication networks. For a single link of fixed capacity, we investigate a resource allocation mechanism proposed by Kelly [18]. Network users choose bids, which denote the total amount they are willing to pay. A price is then chosen to clear the market; for the case of a single link, this allocation mechanism allocates fractions of the resource to the users in *proportion* to their bids. Kelly has previously shown that if users are *price taking*—that is, if they do not anticipate the effects of their actions on the market-clearing price—the resulting *competitive equilibrium* allocation is fully efficient. Our key result in this section is that when users are price anticipating, aggregate utility falls by no more than 25% relative to the maximum possible.

In Section 12.3, we consider the same basic mechanism as in Section 12.2, but now consider a setting where supply is elastic; this is the model considered by Kelly et al. [19]. In this case the link is characterized by a cost depending on the total allocated rate, rather than a fixed capacity. Again, Kelly et al. have previously shown that if users are price taking, this mechanism maximizes aggregate surplus (i.e., aggregate utility minus cost). For this setting we establish that when users are price anticipating, aggregate surplus falls by no more than approximately 34% relative to the maximum possible.

Sections 12.2 and 12.3 establish efficiency loss results for a specific market mechanism. In Section 12.4, we characterize the mechanism studied in Section 12.2 as the "best" choice of mechanism under reasonable assumptions. Formally, we show that in a class of market-clearing mechanisms satisfying certain simple mathematical assumptions and for which there exist fully efficient competitive equilibria, the mechanism of Section 12.2 uniquely minimizes efficiency loss when market participants are price anticipating. These results justify the attention devoted to understanding the particular market mechanism studied in Sections 12.2 and 12.3; furthermore, they clearly delineate conditions which must be violated if we hope to achieve higher efficiency guarantees than those provided by the results of Sections 12.2 and 12.3.

In Section 12.5, we summarize two further directions of research. First, in Section 12.5.1, we discuss the generalization of the models of Sections 12.2 and 12.3 to networks with arbitrary topology. We consider games where users submit individual bids to each link in the network. Such games are then proven to have the same efficiency loss guarantees as the single link games considered in Sections 12.2 and 12.3.

Next, in Section 12.5.2, motivated by power systems, we discuss a setting where multiple producers bid to satisfy an inelastic demand D. We consider a market mechanism where producers submit supply functions restricted to lie in a certain one-parameter family, and a market-clearing price is chosen to ensure that aggregate supply is equal to the inelastic demand. We establish that when producers are price anticipating, aggregate production cost rises by no more than a factor $1 + 1/(N-2)$ relative to the minimum possible production cost, where $N > 2$ is the number of firms competing. Finally, we conclude with some open issues in Section 12.6.

12.2 Multiple Consumers, Inelastic Supply

Suppose R users share a communication link of capacity $C > 0$. Let d_r denote the rate allocated to user r. We assume that user r receives a *utility* equal to $U_r(d_r)$ if the allocated rate is d_r; we assume that utility is measured in monetary units. We make the following assumptions on the utility function.

Assumption 1. *For each r, over the domain $d_r \geq 0$ the utility function $U_r(d_r)$ is concave, strictly increasing, and continuous; and over the domain $d_r > 0$, $U_r(d_r)$ is continuously differentiable. Furthermore, the right directional derivative at 0, denoted $U_r'(0)$, is finite.*

Given complete knowledge and centralized control of the system, it would be natural for the link manager to try to solve the following optimization problem [18]:

$$\text{maximize} \quad \sum_r U_r(d_r) \tag{12.1}$$

$$\text{subject to} \quad \sum_r d_r \leq C; \tag{12.2}$$

$$d_r \geq 0, \quad r = 1, \dots, R. \tag{12.3}$$

Note that the objective function of this problem is the *aggregate utility*. Since the objective function is continuous and the feasible region is compact, an optimal solution $\mathbf{d} = (d_1, \dots, d_R)$ exists. If the functions U_r are strictly concave, then the optimal solution is unique, since the feasible region is convex.

In general, the utility functions are not available to the link manager. As a result, we consider the following pricing scheme for rate allocation. Each user r submits a payment (also called a *bid*) w_r to the link manager; we assume $w_r \geq 0$. Given the vector $\mathbf{w} = (w_1, \dots, w_r)$, the link manager chooses a rate allocation $\mathbf{d} = (d_1, \dots, d_r)$. We assume the manager treats all users alike—in other words, the link manager does not *price discriminate*. Each user is charged the same price $\mu > 0$, leading to $d_r = w_r/\mu$. We further assume the manager always seeks to allocate the entire link capacity C; in this case, we expect the price μ to satisfy:

$$\sum_r \frac{w_r}{\mu} = C.$$

The preceding equality can only be satisfied if $\sum_r w_r > 0$, in which case we have:

$$\mu = \frac{\sum_r w_r}{C}. \tag{12.4}$$

In other words, if the manager chooses to allocate the entire available rate at the link, and does not price discriminate between users, then for every nonzero $\mathbf{w}$ there is a *unique* possible price $\mu > 0$, given by the previous equation.

We can interpret this mechanism as a *market-clearing* process by which a price is set so that demand equals supply. To see this interpretation, note that when a user submits a total payment w_r, it is as if the user has submitted a *demand function* $D(p, w_r) = w_r/p$ for $p > 0$. The demand function describes the rate that the user demands at any given price $p > 0$. The link manager then chooses a price μ so that $\sum_r D(\mu, w_r) = C$, i.e., so that the aggregate demand equals the supply C. For the specific form of demand functions we consider here, this leads to the expression for μ given in (12.4). User r then receives a rate allocation given by $D(\mu, w_r)$, and makes a payment $\mu D(\mu, w_r) = w_r$. This interpretation of the mechanism we consider here will be further explored in Section 12.4, where we consider other market-clearing mechanisms with the users submitting demand functions from a family parametrized by a single scalar.

In the remainder of the section, we consider two different models for how users might interact with this price mechanism. In Section 12.2.1, we consider a model where users do not anticipate the effect of their bids on the price, and provide a result, due to Kelly [18], on the existence of a competitive equilibrium. Furthermore, this competitive equilibrium leads to an allocation which is an optimal solution to (12.1)–(12.3). In Section 12.2.2, we change the model and assume users are price anticipating, and provide a result (due to Hajek and Gopalakrishnan [12]) on the existence and uniqueness of a Nash equilibrium. In Section 12.2.3, we then consider the loss of efficiency at this Nash equilibrium, relative to the optimal solution to (12.1)–(12.3).

12.2.1 Price taking users and competitive equilibrium

In this section, we consider a *competitive equilibrium* between the users and the link manager [23], following the development of Kelly [18]. A central assumption in the definition of competitive equilibrium is that each user does not anticipate the effect of their payment w_r on the price μ, i.e., each user acts as a *price taker*. In this case, given a price $\mu > 0$, user r acts to maximize the following payoff function over $w_r \geq 0$:

$$P_r(w_r; \mu) = U_r\left(\frac{w_r}{\mu}\right) - w_r. \tag{12.5}$$

The first term represents the utility to user r of receiving a rate allocation equal to w_r/μ; the second term is the payment w_r made to the manager. Observe that since utility is measured in monetary units, the payoff is *quasilinear* in money [23].

We say that a pair $(\mathbf{w}, \mu)$, with $\mathbf{w} \geq 0$ and $\mu > 0$, is a *competitive equilibrium* if users maximize their payoff as defined in (12.5), and the network "clears the market" by setting the price μ according to (12.4):

$$P_r(w_r; \mu) \geq P_r(\overline{w}_r; \mu) \quad \text{for } \overline{w}_r \geq 0, \;\; r = 1, \dots, R; \tag{12.6}$$

$$\mu = \frac{\sum_r w_r}{C}. \tag{12.7}$$

Kelly shows in [18] that when users are price takers, there exists a competitive equilibrium, and the resulting allocation is an optimal solution to (12.1)–(12.3). This is formalized in the following theorem, adapted from [18].

Theorem 1 (Kelly, [18]). *Suppose that Assumption 1 holds. Then there exists a competitive equilibrium, i.e., a vector* $\mathbf{w} = (w_1, \dots, w_R) \geq 0$ *and a scalar* $\mu > 0$ *satisfying* (12.6)–(12.7).

In this case, the scalar μ *is uniquely determined, and the vector* $\mathbf{d} = \mathbf{w}/\mu$ *is an optimal solution to* (12.1)–(12.3). *If the functions* U_r *are strictly concave, then* $\mathbf{w}$ *is uniquely determined as well.*

Theorem 1 shows that under the assumption that the users of the link behave as price takers, there exists a bid vector $\mathbf{w}$ where all users have optimally chosen their bids w_r, with respect to the given price $\mu = \sum_r w_r/C$; and at this "equilibrium," aggregate utility is maximized. However, when the price taking assumption is violated, the model changes into a game and the guarantee of Theorem 1 is no longer valid.

12.2.2 Price anticipating users and Nash equilibrium

We now consider an alternative model where the users of a single link are price anticipating, rather than price takers. The key difference is that while the payoff function P_r takes the price μ as a fixed parameter in (12.5), price anticipating users will realize that μ is set according to (12.4), and adjust their payoff accordingly; this makes the model a game between the R players.

We use the notation $\mathbf{w}_{-r}$ to denote the vector of all bids by users other than r; i.e., $\mathbf{w}_{-r} = (w_1, w_2, \ldots, w_{r-1}, w_{r+1}, \ldots, w_R)$. Given $\mathbf{w}_{-r}$, each user r chooses w_r to maximize:

$$Q_r(w_r; \mathbf{w}_{-r}) = \begin{cases} U_r\left(\dfrac{w_r}{\sum_s w_s} C\right) - w_r, \text{ if } w_r > 0; \\ \\ U_r(0), \qquad\qquad\qquad \text{if } w_r = 0 \end{cases} \tag{12.8}$$

over all nonnegative w_r. The second condition is required so that the rate allocation to user r is zero when $w_r = 0$, even if all other users choose $\mathbf{w}_{-r}$ so that $\sum_{s \neq r} w_s = 0$. The payoff function Q_r is similar to the payoff function P_r, except that the user anticipates that the network will set the price μ according to (12.4). A *Nash equilibrium* of the game defined by $(Q_1, \ldots, Q_R)$ is a vector $\mathbf{w} \geq 0$ such that for all r:

$$Q_r(w_r; \mathbf{w}_{-r}) \geq Q_r(\overline{w}_r; \mathbf{w}_{-r}), \quad \text{for all } \overline{w}_r \geq 0. \tag{12.9}$$

Hajek and Gopalakrishnan have shown that there exists a unique Nash equilibrium when multiple users share the link, by showing that at a Nash equilibrium it is *as if* the users are solving another optimization problem of the same form as the problem (12.1)–(12.3), but with “modified” utility functions. This is formalized in the following theorem, adapted from [12].

Theorem 2 (Hajek and Gopalakrishnan, [12]). *Suppose that $R > 1$, and that Assumption 1 holds. Then there exists a unique Nash equilibrium $\mathbf{w} \geq 0$ of the game defined by $(Q_1, \ldots, Q_R)$, and it satisfies $\sum_r w_r > 0$.*

In this case, the vector $\mathbf{d}$ defined by:

$$d_r = \frac{w_r}{\sum_s w_s} C, \quad r = 1, \ldots, R, \tag{12.10}$$

is the unique optimal solution to the following optimization problem:

$$\text{maximize} \quad \sum_r \hat{U}_r(d_r) \tag{12.11}$$

$$\text{subject to} \quad \sum_r d_r \leq C; \tag{12.12}$$

$$d_r \geq 0, \quad r = 1, \ldots, R, \tag{12.13}$$

where

$$\hat{U}_r(d_r) = \left(1 - \frac{d_r}{C}\right) U_r(d_r) + \left(\frac{d_r}{C}\right) \left(\frac{1}{d_r} \int_0^{d_r} U_r(z)\, dz\right). \tag{12.14}$$

Theorem 2 shows that the unique Nash equilibrium of the game is characterized as the solution to the optimization problem above. Other games have also profited from such relationships—notably traffic routing games, in which Nash–Wardrop equilibria can be found as solutions to a related global optimization problem. Roughgarden and Tardos use this fact to their advantage in computing efficiency loss for such games [28]; Correa, Schulz, and Stier Moses also use this relationship to consider routing games in capacitated networks [5]. Finally, we note that for the game presented here, several authors have derived results similar to Theorem 2 [7, 21, 22], though not as general.

12.2.3 Efficiency loss

We let $\mathbf{d}^S$ denote an optimal solution to (12.1)–(12.3), and let $\mathbf{d}^G$ denote the unique optimal solution to (12.11)–(12.13). We now investigate the efficiency loss of this system; that is, the utility loss caused by the price anticipating behavior of the users. More precisely, we will compare the utility $\sum_r U_r(d_r^G)$ obtained when the users fully evaluate the effect of their actions on the price, and the maximum possible aggregate utility $\sum_r U_r(d_r^S)$. (We know, of course, that $\sum_r U_r(d_r^G) \leq \sum_r U_r(d_r^S)$, by definition of $\mathbf{d}^S$.) According to the following theorem, the worst case efficiency loss is exactly 25%; the proof may be found in [17].

Theorem 3. *Suppose that $R > 1$, and that Assumption 1 holds. Suppose also that $U_r(0) \geq 0$ for all r. If $\mathbf{d}^S$ is any optimal solution to* (12.1)–(12.3)*, and $\mathbf{d}^G$ is the unique optimal solution to* (12.11)–(12.13)*, then:*

$$\sum_r U_r(d_r^G) \geq \frac{3}{4} \sum_r U_r(d_r^S).$$

Furthermore, this bound is tight: for every $\varepsilon > 0$, there exists a choice of R, and a choice of (linear) utility functions U_r, $r = 1, \ldots, R$, such that:

$$\sum_r U_r(d_r^G) \leq \left(\frac{3}{4} + \varepsilon\right) \left(\sum_r U_r(d_r^S)\right).$$

We provide some comments on the method for proving a result such as Theorem 3. The first step is to show that the worst case efficiency loss occurs when the utility functions belong to a certain finite-dimensional family; in the current context, it is the family of linear utility functions. Identifying the worst case utility functions amounts to minimizing an efficiency measure over all possible choices of the coefficients of the linear utility functions. It turns out that this minimization can be cast as a sequence of finite-dimensional nonlinear optimization problems (each problem in the sequence corresponding to a different number R of users), which can be studied analytically. In the context of Theorem 3, the worst efficiency loss corresponds to a link of capacity 1, where user 1 has utility $U_1(d_1) = d_1$, and all other users have utility $U_r(d_r) \approx d_r/2$. As $R \to \infty$, at the Nash equilibrium of the game, user 1 receives a rate $d_1^G = 1/2$, while the remaining users uniformly split the rate $1 - d_1^G = 1/2$ among themselves, yielding an aggregate utility of $3/4$.

We note that a similar bound was observed by Roughgarden and Tardos for traffic routing games with affine link latency functions [28]. They found that the ratio of worst case Nash equilibrium cost to optimal cost was $4/3$. However, it is questionable whether a relationship can be drawn between the two games; in particular, we note that while Theorem 3 holds even if the utility functions are nonlinear, Roughgarden and Tardos have shown that the efficiency loss due to selfish users in traffic routing may be arbitrarily high if link latency functions are nonlinear.

12.3 Multiple Consumers, Elastic Supply

In this section, we allow the supply of the scarce resource to be elastic, rather than fixed as in the previous section. Rather than being characterized by a capacity, we will characterize the resource through a *cost function* that gives the cost incurred by the resource as a function of the flow through it. We continue to assume that R users share a single communication link, and that user r receives a utility $U_r(d_r)$ if the allocated rate is d_r. We let $f = \sum_r d_r$ denote the total rate allocated at the link, and let $C(f)$ denote the cost incurred at the link when the total allocated rate is $f \geq 0$. We will assume that both U_r and C are measured in the same monetary units. A natural interpretation is that $U_r(d_r)$ is the monetary value to user r of a rate allocation d_r, and $C(f)$ is a monetary cost for congestion at the link when the total allocated rate is f.

We continue to assume the utility functions U_r satisfy Assumption 1. In addition, we make the following assumption on the cost function C.

Assumption 2. *There exists a continuous, convex, strictly increasing function $p(f)$ over $f \geq 0$ with $p(0) = 0$, such that for $f \geq 0$:*

$$C(f) = \int_0^f p(z)dz.$$

Thus $C(f)$ is strictly convex and strictly increasing.

Given complete knowledge and centralized control of the system, it would be natural for the link manager to try to solve the following optimization problem[18]:

$$\text{maximize} \quad \sum_r U_r(d_r) - C\left(\sum_r d_r\right) \qquad (12.15)$$

$$\text{subject to} \quad d_r \geq 0, \qquad r = 1, \ldots, R. \qquad (12.16)$$

We refer to the objective function (12.15) as the *aggregate surplus*. This is the net monetary benefit to the economy consisting of the users and the single link. Since the objective function is continuous, and U_r increases at most linearly, while C increases superlinearly, an optimal solution $\mathbf{d}^S = (d_1^S, \dots, d_R^S)$ exists; since the feasible region is convex and C is strictly convex, if the functions U_r are strictly concave, then the optimal solution is unique.

We consider the following pricing scheme for rate allocation, a natural analogue of the mechanism presented in Section 12.2. Each user r submits a payment (or *bid*) of w_r to the resource manager. Given the composite vector $\mathbf{w} = (w_1, \dots, w_r)$, the resource manager chooses a rate allocation $\mathbf{d}(\mathbf{w}) = (d_1(\mathbf{w}), \dots, d_R(\mathbf{w}))$. We assume the manager treats all users alike—in other words, the network manager does not *price differentiate*. Thus the network manager sets a single price $\mu(\mathbf{w})$; we assume that $\mu(\mathbf{w}) = 0$ if $w_r = 0$ for all r, and $\mu(\mathbf{w}) > 0$ otherwise. All users are then charged the same price $\mu(\mathbf{w})$, leading to:

$$d_r(\mathbf{w}) = \begin{cases} 0, & \text{if } w_r = 0; \\ \dfrac{w_r}{\mu(\mathbf{w})}, & \text{if } w_r > 0. \end{cases}$$

Notice that, with this formulation, the rate allocated to user r is similar to the rate allocated to user r in the model of Section 12.2. The key difference in this setting is that the aggregate rate is not constrained to an inelastic supply; rather, associated with the choice of price $\mu(\mathbf{w})$ is an aggregate rate function $f(\mathbf{w})$, defined by:

$$f(\mathbf{w}) = \sum_r d_r(\mathbf{w}) = \begin{cases} 0, & \text{if } \sum_r w_r = 0; \\ \dfrac{\sum_r w_r}{\mu(\mathbf{w})}, & \text{if } \sum_r w_r > 0. \end{cases} \qquad (12.17)$$

Let us assume for now that given a price $\mu > 0$, user r wishes to maximize the following payoff function over $w_r \geq 0$:

$$P_r(w_r; \mu) = U_r\left(\frac{w_r}{\mu}\right) - w_r. \qquad (12.18)$$

The first term represents the utility to user r of receiving a rate allocation equal to w_r/μ; the second term is the payment w_r made to the manager.

Notice that as formulated above, the payoff function P_r assumes that user r acts as a *price taker*; that is, user r does not *anticipate* the effect of his choice of w_r on the price μ, and hence on his resulting rate allocation $d_r(\mathbf{w})$. Informally, we expect that in such a situation the aggregate surplus will be maximized if the network manager sets a price equal to marginal cost; that is, if the price function satisfies:

$$\mu(\mathbf{w}) = p(f(\mathbf{w})). \qquad (12.19)$$

The well-posedness of such a pricing mechanism is the subject of the following proposition.

Proposition 1. *Suppose Assumption 2 holds. Given any vector of bids* $\mathbf{w} \geq 0$*, there exists a unique pair* $(\mu(\mathbf{w}), f(\mathbf{w})) \geq 0$ *satisfying* (12.17) *and* (12.19)*, and in this case* $f(\mathbf{w})$ *is the unique solution* f *to:*

$$\sum_r w_r = fp(f). \tag{12.20}$$

Furthermore, $f(\cdot)$ has the following properties: (1) $f(\mathbf{0}) = 0$; (2) $f(\mathbf{w})$ is continuous for $\mathbf{w} \geq 0$; (3) $f(\mathbf{w})$ is a strictly increasing and strictly concave function of $\sum_r w_r$; and (4) $f(\mathbf{w}) \to \infty$ as $\sum_r w_r \to \infty$.

Observe that we can view (12.20) as a market-clearing process. Given the total revenue $\sum_r w_r$ from the users, the link manager chooses an aggregate rate $f(\mathbf{w})$ so that the revenue is exactly equal to the aggregate charge $f(\mathbf{w})p(f(\mathbf{w}))$. Due to Assumption 2, this market-clearing aggregate rate is uniquely determined. Kelly et al. present two algorithms in [19] which amount to dynamic processes of market-clearing; as a result, a key motivation for the mechanism we study in this section is that it represents the equilibrium behavior of the algorithms in [19].

For the remainder of this section, we will assume that $\mu(\mathbf{w})$ is set according to the choice prescribed in Proposition 1, as follows.

Assumption 3. *For all $\mathbf{w} \geq 0$, the aggregate rate $f(\mathbf{w})$ is the unique solution to (12.20): $\sum_r w_r = f(\mathbf{w})p(f(\mathbf{w}))$. Furthermore, for each r, $d_r(\mathbf{w})$ is given by:*

$$d_r(\mathbf{w}) = \begin{cases} 0, & \text{if } w_r = 0; \\ \dfrac{w_r}{p(f(\mathbf{w}))}, & \text{if } w_r > 0. \end{cases} \tag{12.21}$$

Note that we have $f(\mathbf{w}) > 0$ and $p(f(\mathbf{w})) > 0$ if $\sum_r w_r > 0$, and hence d_r is always well defined.

In the remainder of this section, we consider two different models for how users might interact with this price mechanism. In Section 12.3.1, we consider a model where users do not anticipate the effect of their bids on the price, in which case there exists a competitive equilibrium. Furthermore, this competitive equilibrium leads to an allocation which is an optimal solution to (12.15)–(12.16). In Section 12.3.2, we change the model and assume users are price anticipating, in which case there exists a Nash equilibrium. Finally, Section 12.3.3 considers the loss of efficiency at Nash equilibria, relative to the optimal solution to (12.15)–(12.16).

12.3.1 Price taking users and competitive equilibrium

Kelly et al. show in [19] that when users are price takers, and the network sets the price $\mu(\mathbf{w})$ according to (12.17) and (12.19), the resulting allocation is an optimal solution to (12.15)–(12.16). This is formalized in the following theorem, adapted from [19].

Theorem 4 (Kelly et al., [19]). *Suppose Assumptions 1, 2, and 3 hold. Then there exists a vector $\mathbf{w}$ such that $\mu(\mathbf{w}) > 0$, and:*

$$P_r(w_r; \mu(\mathbf{w})) = \max_{\overline{w}_r \geq 0} P_r(\overline{w}_r; \mu(\mathbf{w})), \quad r = 1, \ldots, R. \tag{12.22}$$

For any such vector $\mathbf{w}$*, the vector* $\mathbf{d}(\mathbf{w}) = \mathbf{w}/\mu(\mathbf{w})$ *is an optimal solution to* (12.15)–(12.16)*. If the functions* U_r *are strictly concave, such a vector* $\mathbf{w}$ *is unique as well.*

Theorem 4 shows that with an appropriate choice of price function (as determined by (12.17) and (12.19)), and under the assumption that the users of the link behave as price takers, there exists a bid vector $\mathbf{w}$ where all users have optimally chosen their bids w_r, with respect to the given price $\mu(\mathbf{w})$; and at this "equilibrium," the aggregate surplus is maximized. However, when the price taking assumption is violated, the model changes into a game and the guarantee of Theorem 4 is no longer valid.

12.3.2 Price anticipating users and Nash equilibrium

We now consider an alternative model where the users of a single link are price anticipating, rather than price taking, and play a game to acquire a share of the link. Throughout the remainder of this section as well as in Section 12.3.3, we will assume that the link manager sets the price $\mu(\mathbf{w})$ according to the unique choice prescribed by Proposition 1, as follows.

We adopt the notation $\mathbf{w}_{-r}$ to denote the vector of all bids by users other than r; i.e., $\mathbf{w}_{-r} = (w_1, w_2, \ldots, w_{r-1}, w_{r+1}, \ldots, w_R)$. Then given $\mathbf{w}_{-r}$, each user r chooses $w_r \geq 0$ to maximize:

$$Q_r(w_r; \mathbf{w}_{-r}) = U_r(d_r(\mathbf{w})) - w_r, \tag{12.23}$$

over nonnegative w_r. The payoff function Q_r is similar to the payoff function P_r, except that the user now anticipates that the network will set the price according to Assumption 3, as captured by the allocated rate $d_r(\mathbf{w})$. A *Nash equilibrium* of the game defined by $(Q_1, \ldots, Q_R)$ is a vector $\mathbf{w} \geq 0$ such that for all r:

$$Q_r(w_r; \mathbf{w}_{-r}) \geq Q_r(\overline{w}_r; \mathbf{w}_{-r}), \quad \text{for all } \overline{w}_r \geq 0. \tag{12.24}$$

The proof of the following proposition can be found in [16].

Proposition 2. *Suppose that Assumptions 1, 2, and 3 hold. Then there exists a Nash equilibrium* $\mathbf{w}$ *for the game defined by* $(Q_1, \ldots, Q_R)$*.*

12.3.3 Efficiency loss

We let $\mathbf{d}^S$ denote an optimal solution to (12.15)–(12.16), and let $\mathbf{w}$ denote any Nash equilibrium of the game defined by $(Q_1, \ldots, Q_R)$. We now investigate the associated efficiency loss. In particular, we compare the aggregate surplus $\sum_r U_r(d_r(\mathbf{w})) - C(\sum_r d_r(\mathbf{w}))$ obtained when the users fully evaluate the effect of their actions on the price, and the aggregate surplus $\sum_r U_r(d_r^S) - C(\sum_r d_r^S)$ obtained by choosing an allocation which maximizes aggregate surplus. According to the following theorem, the efficiency loss is no more than approximately 34%, and this bound is essentially tight; the proof can be found in [16].

Theorem 5. *Suppose that Assumptions 1, 2, and 3 hold. Suppose also that* $U_r(0) \geq 0$ *for all* r*. If* $\mathbf{d}^S$ *is any optimal solution to* (12.15)–(12.16)*, and* $\mathbf{w}$ *is any Nash equilibrium of the game defined by* $(Q_1, \ldots, Q_R)$*, then:*

$$\sum_r U_r(d_r(\mathbf{w})) - C\left(\sum_r d_r(\mathbf{w})\right) \geq \left(4\sqrt{2}-5\right)\left(\sum_r U_r(d_r^S) - C\left(\sum_r d_r^S\right)\right). \tag{12.25}$$

In other words, there is no more than approximately a 34% efficiency loss when users are price anticipating.

Furthermore, this bound is tight: for every $\delta > 0$*, there exists a choice of* R*, a choice of (linear) utility functions* U_r*,* $r = 1, \ldots, R$*, and a (piecewise linear) price function* p *such that a Nash equilibrium* $\mathbf{w}$ *exists with:*

$$\begin{aligned}\sum_r U_r(d_r(\mathbf{w})) - C\left(\sum_r d_r(\mathbf{w})\right)&\\ \leq \left(4\sqrt{2}-5+\delta\right)&\left(\sum_r U_r(d_r^S) - C\left(\sum_r d_r^S\right)\right).\end{aligned} \tag{12.26}$$

Let us remark here that, according to the proof of Theorem 5, the worst possible efficiency loss is achieved along a sequence of games where:

1. The price function p has the following form, with $b \to \infty$:
$$p(f) = \begin{cases}(2-\sqrt{2})f, & \text{if } 0 \leq f \leq 1;\\ 2-\sqrt{2}+b(f-1), & \text{if } f \geq 1;\end{cases}$$
2. The number of users becomes large ($R \to \infty$); and
3. User 1 has linear utility with $U_1(d_1) = d_1$, and all other users r have linear utility with $U_r(d_r) = \alpha_r d_r$, where $\alpha_r \approx p(1) = 2 - \sqrt{2}$.

(Note that formally, we must take care that the limits of $R \to \infty$ and $b \to \infty$ are taken in the correct order; in particular, in the proof we first have $R \to \infty$, and then $b \to \infty$.) In this limit, we find that at the Nash equilibrium the aggregate allocated rate is 1, and the Nash equilibrium aggregate surplus converges to $4\sqrt{2} - 5$.

12.4 A Characterization Theorem

In this section we revisit the resource allocation problem of Section 12.2, and address the following question: can we identify a mechanism that minimizes the efficiency loss, in the presence of price anticipating users, within a class of mechanisms with certain desirable properties?

Formally, we consider a collection of users bidding to receive a share of a finite, infinitely divisible resource of capacity C. Each user has a utility function $U : \mathbb{R}^+ \to \mathbb{R}^+$ (where $\mathbb{R}^+ = [0, \infty)$) that satisfies Assumption 1. More specifically, U belongs to the set $\mathcal{U}$ utility functions defined by

$$\mathcal{U} = \big\{U : \mathbb{R}^+ \to \mathbb{R}^+ \mid U \text{ is continuous, strictly increasing, concave on } [0,\infty), \\ \text{and continuously differentiable on } [0,\infty), \text{with } U'(0) < \infty\big\}.$$

We let R denote the number of users, and let $\mathbf{U} = (U_1, \ldots, U_R)$ denote the vector of utility functions, where U_r is the utility function of user r. We call a pair $(R, \mathbf{U})$, where $R > 1$ and $\mathbf{U} \in \mathcal{U}^R$, a *utility system*; our goal will be to design a resource allocation mechanism with attractive efficiency guarantees for all utility systems.

We assume once more that utility is measured in monetary units; thus, if user r receives a rate allocation d_r, but must pay w_r, his net net payoff is:

$$U_r(d_r) - w_r.$$

Given a utility system $\mathbf{U} \in \mathcal{U}^R$, the social objective is to maximize aggregate utility, as defined in the problem (12.1)–(12.3); we repeat that problem here, and refer to it as the problem *SYSTEM*$(C, R, \mathbf{U})$, to emphasize that the problem is specified by C, R, and the utility system (U).

$$\text{maximize} \quad \sum_{r=1}^{R} U_r(d_r) \qquad (12.27)$$

$$\text{subject to} \quad \sum_{r=1}^{R} d_r \le C; \qquad (12.28)$$

$$\mathbf{d} \ge 0. \qquad (12.29)$$

We will say that $\mathbf{d}$ *solves SYSTEM*$(C, R, \mathbf{U})$ if $\mathbf{d}$ is an optimal solution to (12.27)–(12.29), given the utility system $(R, \mathbf{U})$.

In general, the utility system $(R, \mathbf{U})$ is unknown to the mechanism designer, so a mechanism must be designed to elicit information from the users. We will focus on mechanisms in which each user i submits a demand function, within a one-parameter family of admissible demand functions. In particular, each user has a one-dimensional strategic variable, denoted by θ_i.

Definition 1. *Given $C > 0$, a* smooth market-clearing mechanism for C *is a differentiable function $D : (0,\infty) \times [0,\infty) \to \mathbb{R}^+$ such that for all R, and for all nonzero $\boldsymbol{\theta} \in (\mathbb{R}^+)^R$, there exists a unique solution $p > 0$ to the following equation:*

$$\sum_{r=1}^{R} D(p, \theta_r) = C.$$

We let $p_D(\boldsymbol{\theta})$ denote this solution, and refer to it as the market-clearing price.

Note that the market-clearing price is undefined if $\boldsymbol{\theta} = \mathbf{0}$. As we will see below, when we formulate a game between consumers for a given mechanism D, we will assume that the payoff to all players is $-\infty$ if the composite strategy vector is $\boldsymbol{\theta} = \mathbf{0}$. Note that this is slightly different from the definition in Section 12.2, where the payoff to a player who submits $\theta = 0$ is set to zero. We will discuss this distinction further later; we simply note for the moment that it does not affect the results of this section.

Our definition of smooth market-clearing mechanisms generalizes the mechanism discussed in Section 12.2. We recall that in that development, each user submits a demand function of the form $D(p,\theta) = \theta/p$, and the link manager chooses a price $p_D(\boldsymbol{\theta})$ to ensure that $\sum_{r=1}^R D(p,\theta_r) = C$. Thus, for this mechanism, we have $p_D(\boldsymbol{\theta}) = \sum_{r=1}^R \theta_r/C$ if $\boldsymbol{\theta} \neq 0$. Another related example is provided by $D(p,\theta) = \theta/\sqrt{p}$; in this case it is straightforward to verify that $p_D(\boldsymbol{\theta}) = (\sum_{r=1}^R \theta_r/C)^2$, for $\boldsymbol{\theta} \neq 0$.

We will further restrict attention to a particular class of mechanisms denoted $\mathcal{D}$, which we define as follows.

Definition 2. *The class $\mathcal{D}$ consists of all functions $D(p,\theta)$ such that the following conditions are satisfied:*

1. *For all $C > 0$, D is a smooth market-clearing mechanism for C (cf. Definition 1).*
2. *For all $C > 0$, and for all $U_r \in \mathcal{U}$, the payoff of a price anticipating user is concave; that is, for all R, and for all $\boldsymbol{\theta}_{-r} \in (\mathbb{R}^+)^R$, the function:*

$$U_r(D(p_D(\boldsymbol{\theta}),\theta_r) - p_D(\boldsymbol{\theta})D(p_D(\boldsymbol{\theta}),\theta_r)$$

is concave in $\theta_r > 0$ if $\boldsymbol{\theta}_{-r} = \mathbf{0}$, and concave in $\theta_r \geq 0$ if $\boldsymbol{\theta}_{-r} \neq \mathbf{0}$.

3. *The demand functions are nonnegative; i.e., for all $p > 0$ and $\theta \geq 0$, $D(p,\theta) \geq 0$.*

The first condition requires a mechanism in $\mathcal{D}$ to be a smooth market-clearing mechanism for any $C > 0$; in particular, the market-clearing price $p_D(\boldsymbol{\theta})$ must be uniquely defined for any $C > 0$. (Note that in the notation we suppress the dependence of the market-clearing price $p_D(\boldsymbol{\theta})$ on the capacity C.) The second condition allows us to characterize Nash equilibria in terms of only first order conditions; indeed, some such assumption needs to be in place in order to guarantee existence of pure strategy Nash equilibria [26]. Finally, the third condition is a normalization condition, which ensures that a user is never required to *supply* some quantity of the resource (which would be the case if we allowed $D(p,\theta) < 0$).

In order to state the main result of this section, we must define *competitive equilibrium* and *Nash equilibrium*. Given a utility system $(R,\mathbf{U})$, a capacity $C > 0$, and a smooth market-clearing mechanism $D \in \mathcal{D}$, we say that a nonzero vector $\boldsymbol{\theta} \in (\mathbb{R}^+)^R$ is a competitive equilibrium if $\mu = p_D(\boldsymbol{\theta})$ satisfies:

$$\theta_r \in \arg\max_{\overline{\theta}_r \geq 0} \left[U_r(D(\mu,\overline{\theta}_r)) - \mu D(\mu,\overline{\theta}_r)\right], \qquad \forall r. \tag{12.30}$$

Similarly, we say that a nonzero vector $\boldsymbol{\theta} \in (\mathbb{R}^+)^R$ is a Nash equilibrium if:

$$\theta_r \in \arg\max_{\overline{\theta}_r \geq 0} Q_r(\overline{\theta}_r;\boldsymbol{\theta}_{-r}), \qquad \forall r, \tag{12.31}$$

where

$$Q_r(\theta_r;\boldsymbol{\theta}_{-r}) = \begin{cases} U_r(D(p_D(\boldsymbol{\theta}),\theta_r)) - p_D(\boldsymbol{\theta})D(p_D(\boldsymbol{\theta}),\theta_r), & \text{if } \boldsymbol{\theta} \neq \mathbf{0}; \\ -\infty, & \text{if } \boldsymbol{\theta} = \mathbf{0}. \end{cases} \tag{12.32}$$

Notice that the payoff is $-\infty$ if the composite strategy vector is $\boldsymbol{\theta} = \mathbf{0}$, since in this case no market-clearing price exists.

Our interest is in the worst-case ratio of aggregate utility at any Nash equilibrium to the optimal value of *SYSTEM*$(C, R, \mathbf{U})$ (termed the the "price of anarchy" by Papadimitriou [25]). Formally, for $D \in \mathcal{D}$ and a capacity $C > 0$ we define a constant $\rho(C, D)$ as follows:

$$\rho(C,D) = \inf \left\{ \frac{\sum_{r=1}^{R} U_r(D(p_D(\boldsymbol{\theta}),\theta_r))}{\sum_{r=1}^{R} U_r(d_r)} \,\middle|\, R > 1, \mathbf{U} \in \mathcal{U}^R, \right.$$
$$\left. \mathbf{d} \text{ solves } \textit{SYSTEM}(C,R,\mathbf{U}) \text{ and } \boldsymbol{\theta} \text{ is a Nash equilibrium} \right\}.$$

Note that since all $U \in \mathcal{U}$ are strictly increasing and nonnegative, and $C > 0$, the aggregate utility $\sum_{r=1}^{R} U_r(d_r)$ is strictly positive for any utility system $(R, \mathbf{U})$ and any optimal solution $\mathbf{d}$ to *SYSTEM*$(C, R, \mathbf{U})$. However, Nash equilibria may not exist for some utility systems $(R, \mathbf{U})$; in this case we set $\rho(C, D) = -\infty$.

The following theorem shows that among smooth market-clearing mechanisms for which there always exists a fully efficient competitive equilibrium, the mechanism proposed in Section 12.2 minimizes efficiency loss when users are price anticipating. The proof can be found in Chapter 5 of [15].

Theorem 6. *Let $D \in \mathcal{D}$ be a smooth market-clearing mechanism such that for all capacities $C > 0$ and utility systems $(R, \mathbf{U})$, there exists a competitive equilibrium $\boldsymbol{\theta}$ such that $(D(p_D(\boldsymbol{\theta}), \theta_r), r = 1, \ldots, R)$ solves SYSTEM$(C, R, \mathbf{U})$. Then for any capacity C and utility system $(R, \mathbf{U})$, there exists a unique Nash equilibrium. Furthermore, $\rho(C, D) \leq 3/4$ for all $C > 0$ and all $D \in \mathcal{D}$, and this bound is met with equality if and only if $D(p, \theta) = \Delta\theta/p$ for some $\Delta > 0$.*

Theorem 6 suggests that the best efficiency guarantee we can hope to achieve is 75%, if we are restricted to market-clearing mechanisms with scalar strategy spaces. A key restriction in the mechanisms we consider is that a single price is chosen to clear the market. If the market designer is granted the latitude to *price discriminate* (i.e., to charge a different price to each user), better efficiency guarantees are possible. The most famous mechanisms which ensure such a guarantee are the Vickrey–Clarke–Groves class of mechanisms, for which fully efficient dominant strategy equilibria exist [4, 11, 32]. More recently, in a networking context, Sanghavi and Hajek [30] have shown that if users choose their payments (as in the Kelly mechanism), but the link manager is allowed to choose the allocation to users as an arbitrary function of the payments, it is possible to ensure no worse than a 13% efficiency loss. Furthermore, Yang and Hajek [34] have shown that if a mechanism allocates resources in proportion to the users' strategies (i.e., user r receives a fraction $\theta_r/(\sum_{s=1}^{R}\theta_s)$ of the resource), then by using differentiated pricing, it is possible to guarantee arbitrarily small efficiency loss at the Nash equilibrium. The mechanisms proposed by both Sanghavi and Hajek [30] as well as Yang and Hajek [34] require price discrimination, since the ratio of payment to allocation is not necessarily identical for all users (as must be the case in the market-clearing mechanisms studied here).

12.5 Further Directions

In addition to the results outlined above, several additional threads are included in this body of research. In this section, we describe two extensions: (1) resource allocation in general networks; and (2) a setting of multiple producers competing to satisfy an inelastic demand.

12.5.1 General networks

The models presented in Sections 12.2 and 12.3 only consider resource allocation for a single link. We now consider extensions to the network case, following [16] and [17]. We consider networks consisting of a set of links; each user has a set of paths available through the network to send traffic, and each path uses a subset of the links. In a setting of inelastic supply, each link j is characterized by a fixed capacity C_j. In a setting of elastic supply, each link j is characterized by a cost function $C_j(\cdot)$. We continue to assume that each user r receives a utility $U_r(d_r)$ from a total rate allocation d_r; however, note that in a network context d_r is the total rate delivered to user r across all paths available to user r through the network.

We extend the single link market mechanisms to multiple links by treating each link as a separate market. Thus we consider a game where each user requests service from multiple links by submitting an individual bid to each link. Links then allocate rates using the same scheme as in the single link model, and each user sends the maximum rate possible, given the vector of rates allocated from links in the network. Although this definition of the game is natural, we demonstrate that Nash equilibria may not exist in the setting of inelastic supply, due to a discontinuity in the payoff functions of individual players. (This problem also arises in the single link setting, but is irrelevant there as long as at least two players share the link.) To address the discontinuity, we extend the strategy space by allowing each user to request a nonzero rate without submitting a positive bid to a link, if the total payment made by other users at that link is zero; this extension is sufficient to guarantee existence of a Nash equilibrium. In the setting of elastic supply, Nash equilibria are always guaranteed to exist, without having to extend the strategy space. Finally, we show that in this network setting, if link capacities are inelastic then the total utility achieved at any Nash equilibrium of the game is no less than 3/4 of the maximum possible aggregate utility; and if link supplies are elastic then the aggregate surplus achieved at any Nash equilibrium of the game is no less than a factor $4\sqrt{2}-5$ of the maximal aggregate surplus. These results extend the efficiency loss results from the single link setting to general networks.

The mechanisms we have studied require each user to submit a separate bid for each link that the user may use. An alternative mechanism had been proposed earlier by Kelly [18] whereby a user submits a single total payment, and the network determines both the rate allocations, as well as the divisions of the users' total payments among the links; in the single link case, this scheme reduces to that studied in Section 12.2. But Hajek and Yang [13] have shown that Kelly's mechanism can result in Nash equilibria in which the aggregate utility is an arbitrarily small fraction of the optimal aggregate utility. It remains an open problem whether there exists a network resource allocation mechanism in which each user submits a single number, representing total payment, and which has some nontrivial efficiency guarantees.

12.5.2 Multiple producers, inelastic demand

The models presented thus far consider consumers competing for resources in scarce supply. Motivated by current problems in market design for electric power systems, we consider a model where multiple producers compete to satisfy an inelastic demand. Demand for electricity, particularly in the short run, is characterized by low elasticity with respect to price, i.e., changes in price do not lead to significant changes in the level of demand; see, e.g., [31], Section 1-7.3. A basic model for electricity market operation involves *supply function bidding* [20]: each generator submits a supply function expressing their willingness to produce electricity as a function of the market clearing price. A single price is then chosen to ensure that supply matches the inelastic demand.

Most previous work on supply function bidding has focused almost entirely on using the supply function equilibrium (SFE) framework of Klemperer and Meyer [20] for its predictive power. In such models, generators can submit nearly arbitrary supply functions; the Nash equilibria of the resulting game are used to give insight into expected behavior in current markets. In other words, by solving the SFE model for an appropriate set of assumptions, most previous work hopes to lend insight into the operation of power markets which require generators to submit complete supply schedules as bids [1, 6, 9, 10, 29, 33]. But because there may be a multiplicity of equilibria, an explicit understanding of efficiency losses in these games has not been developed. Papers such as the work of Rudkevich et al. [29] do suggest, however, that in the presence of inelastic demand, price anticipating behavior can lead to significant deviations from perfectly efficient allocations.

For this reason we take a different approach (see Chapter 4 of [15]). We consider *restrictions* on the supply functions which can be chosen by the generators, and aim to design these restrictions so that nearly efficient allocations are achieved even if firms are price anticipating. Formally, we assume that each firm n has a convex cost function C_n, as a function of the electricity generated. An efficient production vector minimizes the aggregate cost $\sum_n C_n(s_n)$, subject to the constraint that the total produced electricity $\sum_n s_n$ must equal the demand D. We then consider the following market. Each firm submits a supply function of the form $S(p, w) = D - w/p$, where

D is the fixed (exogenous) demand and w is a nonnegative scalar chosen by the firm. The market then chooses a price so that aggregate supply is equal to demand.

If we assume that firms are price taking, it is possible to show that there exists a competitive equilibrium; furthermore, at this competitive equilibrium the resulting allocation minimizes aggregate production cost. If we assume instead that firms are price anticipating, we can establish existence of a Nash equilibrium and uniqueness of the resulting production vector, as long as more than two firms compete. Next, we consider the aggregate production cost at a Nash equilibrium relative to the minimal possible aggregate production cost. As long as more than two firms are competing, we show that the ratio of Nash equilibrium production cost to the minimal production cost is no worse than $1 + 1/(N - 2)$, where N is the number of firms in the market. Furthermore, we demonstrate that this efficiency loss result carries over even to a setting where demand is inelastic but stochastically determined, by showing that in such an instance it is *as if* firms play a game with deterministic demand but different cost functions. Finally, a characterization theorem, similar to the one in Section 12.4, is also available, indicating that the mechanism under study has the best possible efficiency guarantees, within a class of mechanisms in which the generators are restricted to submitting a supply function chosen from within a restricted, one-parameter family.

These results, which have been established in [15], suggest that market power can be controlled, and efficient allocations guaranteed, by restricting the supply functions available to generators in electricity markets. Restricted families of supply functions have also been considered elsewhere in the literature, e.g., in [3]. However, these models are typically used as approximations to unconstrained supply function bidding, and thus the resulting efficiency loss has not been studied. Still, this work leaves many open questions; in particular, the dynamics of power systems, together with their complex network structure, has not been captured in the models developed (in contrast to the telecommunications models previously discussed). Furthermore, the work described here depends on convexity assumptions on the cost functions of the producers, and such assumptions may generally not hold in electricity markets [14]. Finally, away from a Nash equilibrium, e.g., if some generators do not act rationally, the remaining generators may have to produce electricity at highly undesirable or even impossible levels. Addressing these types of questions is the subject of current research.

12.6 Open Issues

We have discussed the efficiency properties of Nash equilibria associated with certain resource allocation mechanisms. For the case where there is a single available resource (respectively, a single demand to be satisfied), the mechanisms involve the submission of a demand (respectively, a supply) function, which can be specified in terms of a single parameter, followed by market-clearing. In each case, we have provided a tight bound on the worst case efficiency loss. It remains to understand the worst case efficiency loss when mechanisms belonging to broader classes are considered. For example, in the context of Section 12.2 what efficiency guarantees are possible if users can choose a demand function from within a two-parameter family of demand curves?

Another research direction relates to the study of natural adjustment dynamics in the context of various mechanisms. Indeed, a desirable mechanism should not only have efficiency guarantees for the resulting Nash equilibria. It should also allow for simple adjustment algorithms whereby the different players can converge, in a stable manner, to such a Nash equilibrium.

Acknowledgment. This research was supported by the National Science Foundation under a Graduate Research Fellowship and grant ECS-0312921, by the Defense Advanced Research Projects Agency under the Next Generation Internet Initiative, and by the Army Research Office under grant DAAD10-00-1-0466. The material of Section 12.3 was reported in a paper jointly authored with Shie Mannor; the authors would also like to acknowledge helpful conversations with Frank Kelly.

References

[1] E.J. Anderson and A.B. Philpott, Using supply functions for offering generation into an electricity market, *Operations Research*, **50** (3):477–499, 2002.

[2] L.M. Ausubel and P. Milgrom, The lovely but lonely Vickrey auction, 2004, in preparation.

[3] R. Baldick, R. Grant, and E.P. Kahn, Theory and application of linear supply function equilibrium in electricity markets, *Journal of Regulatory Economics*, 2004, to appear.

[4] E.H. Clarke, Multipart pricing of public goods, *Public Choice*, **11**:19–33, 1971.

[5] J.R. Correa, A.S. Schulz, and N. Stier Moses, Selfish routing in capacitated networks, *Mathematics of Operations Research*, 2004, to appear.

[6] C.J. Day, B.F. Hobbs, and J.-S. Pang, Oligopolistic competition in power networks: a conjectured supply function approach, *IEEE Transactions on Power Systems*, **17** (3):597–607, 2002.

[7] R.J. Gibbens and F.P. Kelly, Resource pricing and the evolution of congestion control, *Automatica*, **35**:1969–1985, 1999.

[8] J.R. Green and J.-J. Laffont, *Incentives in Public Decision-Making*, North-Holland Publishing Company, Amsterdam, The Netherlands, 1979.

[9] R.J. Green, Increasing competition in the British electricity spot market, *Journal of Industrial Economics*, **44** (2):205–216, 1996.

[10] R.J. Green and D.M. Newbery, Competition in the British electricity spot market, *Journal of Political Economy*, **100** (5):929–953, 1992.

[11] T. Groves, Incentives in teams, *Econometrica*, **41** (4):617–631, 1973.

[12] B. Hajek and G. Gopalakrishnan, Do greedy autonomous systems make for a sensible Internet? Presented at the Conference on Stochastic Networks, Stanford University, 2002.

[13] B. Hajek and S. Yang, Strategic buyers in a sum-bid game for flat networks, 2004, submitted.

[14] B.F. Hobbs, M.H. Rothkopf, L.C. Hyde, and R.P. O'Neill, Evaluation of a truthful revelation auction in the context of energy markets with nonconcave benefits, *Journal of Regulatory Economics*, **18** (1):5–32, 2000.

[15] R. Johari, Efficiency Loss in Market Mechanisms for Resource Allocation, PhD thesis, Massachusetts Institute of Technology, 2004.

[16] R. Johari, S. Mannor, and J.N. Tsitsiklis, Efficiency loss in a network resource allocation game: the case of elastic supply, Publication 2605, MIT Laboratory for Information and Decision Systems, 2004.

[17] R. Johari and J.N. Tsitsiklis, Efficiency loss in a network resource allocation game, *Mathematics of Operations Research*, **29** (3):407–435, 2004.

[18] F.P. Kelly, Charging and rate control for elastic traffic, *European Transactions on Telecommunications*, **8**:33–37, 1997.

[19] F.P. Kelly, A.K. Maulloo, and D.K. Tan, Rate control for communication networks: shadow prices, proportional fairness, and stability, *Journal of the Operational Research Society*, **49**:237–252, 1998.

[20] P.D. Klemperer and M.A. Meyer, Supply function equilibria in oligopoly under uncertainty, *Econometrica*, **57** (6):1243–1277, 1989.

[21] R.J. La and V. Anantharam, Charge-sensitive TCP and rate control in the Internet, In *Proceedings of IEEE INFOCOM*, 1166–1175, 2000.

[22] R.T. Maheswaran and T. Basar, Nash equilibrium and decentralized negotiation in auctioning divisible resources, *Group Decision and Negotiation*, **12** (5):361–395, 2003.

[23] A. Mas-Colell, M.D. Whinston, and J.R. Green, Microeconomic Theory, Oxford University Press, Oxford, United Kingdom, 1995.

[24] E. Maskin, Nash equilibrium and welfare optimality, *Review of Economic Studies*, **66** (1):23–38, 1999.

[25] C. Papadimitriou, Algorithms, games, and the Internet, In *Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing*, 749–753, 2001.

[26] J. Rosen, Existence and uniqueness of equilibrium points for concave n-person games, *Econometrica*, **33** (3):520–534, 1965.

[27] M.H. Rothkopf, T.J. Teisberg, and E.P. Kahn, Why are Vickrey auctions rare?, *Journal of Political Economy*, **98** (1):94–109, 1990.

[28] T. Roughgarden and É. Tardos, How bad is selfish routing?, *Journal of the ACM*, **49** (2):236–259, 2002.

[29] A. Rudkevich, M. Duckworth, and R. Rosen, Modeling electricity pricing in a deregulated generation industry: the potential for oligopoly pricing in a poolco, *Energy Journal*, **19** (3):19–48, 1998.

[30] S. Sanghavi and B. Hajek, Optimal allocation of a divisible good to strategic buyers, 2004, submitted.

[31] S. Stoft, Power System Economics: Designing Markets for Electricity, *IEEE Press*, Piscataway, New Jersey, 2002.

[32] W. Vickrey, Counterspeculation, auctions, and competitive sealed tenders, *Journal of Finance*, **16** (1):8–37, 1961.

[33] N.-H.M. von der Fehr and D. Harbord, Spot market competition in the U.K. electricity industry, *Economic Journal*, **103** (418):531–546, 1993.

[34] S. Yang and B. Hajek, An efficient mechanism for allocation of a divisible good and its application to network resource allocation, 2004, in preparation.

13

Decentralized Resource Allocation Mechanisms in Networks: Realization and Implementation

Tudor Mihai Stoenescu[1] and Demosthenis Teneketzis[2]

[1] Social and Information Science Laboratory
California Institute of Technology
tudor@caltech.edu

[2] Department of Electrical Engineering and Computer Science
University of Michigan
teneketzis@eecs.umich.edu

Summary. We discuss how decentralized network resource allocation problems fit within the context of mechanism design (realization theory and implementation theory), and how mechanism design can provide useful insight into the nature of decentralized network resource allocation problems. The discussion is guided by the unicast problem with routing and Quality of Service (QoS) requirements, and the multi-rate multicast service provisioning problem in networks. For these problems we present decentralized resource allocation mechanisms that achieve the solution of the corresponding centralized resource allocation problem and are informationally efficient. We show how the aforementioned mechanisms can be embedded into the general framework of realization theory, and indicate how realization theory can be used to establish the mechanisms' informational efficiency in certain instances. We also present a conjecture related to implementation in Nash equilibria of the optimal centralized solution of the unicast service provisioning problem.

13.1 Introduction: Motivation and Challenges

Today's fast paced world requires a vast amount of information exchange in order to operate efficiently. With the various technological advances the number of types of services being offered (e.g. telephone connections, live audio broadcasting, live video broadcasting, library database access, e-mail, world wide web), is constantly increasing. Each type of service imposes different Quality of Service (QoS) requirements (e.g. delay, percentage of data packet loss, jitter) on the delivery methods. To address these needs extensive communication networks were developed in the past century. Many of these networks (such as telephone networks) were initially designed for the delivery of certain types of information and were later adapted to accommodate new information exchange needs.

Most of today's networks, called integrated services networks, support the delivery of a variety of services to their users. One of the main challenges in integrated services networks is the design of resource allocation strategies which guarantee the delivery of different services, each with its own QoS requirement, and maximize some performance criterion (e.g. the network's utility to its users). The challenge in determining such resource allocation strategies comes from the fact that the network is an informationally decentralized system.

The topic of resource allocation for informationally decentralized systems has been explored in great detail by mathematical economists in the context of mechanism design. In this chapter we discuss how decentralized network resource allocation problems fit within the context of mechanism design, and how ideas from mechanism design can provide useful insight into the nature of decentralized network resource allocation problems. We first present a brief history of the development of the ideas that led to the current state-of-the-art of the theory of mechanism design (Section 13.2.1). Then, we present the key features of the two components of mechanism design, namely, realization theory (Section 13.2.2) and implementation theory (Section 13.2.3). To illustrate how network resource allocation problems fit within the context of mechanism design, and how mechanism design can be used to provide insight into the nature of network resource allocation problems, we consider two classes of network problems: unicast service provisioning with routing and QoS requirements, and multi-rate multicast service provisioning. We discuss these problems in Section 13.3 from the realization theory point of view. We investigate unicast resource allocation with routing from the implementation theory point of view in Section 13.4. We conclude in Section 13.5, by summarizing our discussion and identifying some open problems.

13.2 Mechanism Design

13.2.1 Historical background

Traditionally, economic analysis treated economic systems as one of the "givens." That is, it was assumed that for a given problem the structure of the economic system considered in order to generate a solution is fixed.

At the turn of the last century, economists started to question the effect that the structure of the system has upon the solution of the problem. Although the search for a "better system" has been around at least since Plato's *Republic*, this issue became more relevant with the emergence of the socialist and capitalist economic systems.

One of the major issues that arose from the debate surrounding the virtues of the socialist and the capitalist systems was the methodology in which resources should be allocated. From the early stages of the debate most economists envisioned that the resources in a socialist system would be allocated by the use of a centralized coordinator, while in a capitalist system the resources would generally be allocated through the use of a market. This debate attracted a lot of attention, with prominent economists like Bukharin [15, 16], Dickinson [26], Doob [29], Kautsky [61], Lange [70], Lenin [72], Lerner [73, 74], Marschak [80], Neurath [93], and Taylor [130] arguing in favor of the socialist system; Pierson [105], von Hayek [135–139], and L. von Mises [140] arguing in favor of the capitalist system; and Barone [10], Pareto[99–101], and Walras [141] contributing to the mathematical foundations.[3] Yet, with all these contributions, very few fundamental results on resource allocation theory were available until the 1930s. One of the major reasons for this was the lack of mathematical tools required to tackle such problems. The research efforts in the 1930s along with the subsequent mathematization of classical welfare economics, von Hayek's work, and the developments on mathematical programming and game theory set the mathematical foundations for the development of the theory of mechanism design.

In the 1930s three major research efforts relevant to the design of allocation mechanisms began: i) the development of resource allocation methods for the socialist economy (with major contributors being Lange [70], Lerner [73, 74] and Taylor [130]); ii) the efforts of Hotelling [37, 38] and Lerner [73] on marginal cost pricing and consumer-producer surplus; and iii) the development of the "new welfare economics" (with major contributions by Hicks [36], Kaldor [55], and Scitovsky [21]). A decade later researchers began developing the mathematization of "classical welfare economics" [3, 4, 7, 9, 24, 25, 66, 71].

In the late 1930s and early 1940s, von Hayek made the following key observations: (i) the amount of available information required and the amount of calculations needed by a central-control system in order to determine an optimal allocation would be enormous; and (ii) the economic incentives provided by the market economy could not be reproduced by any of the socialist models. Von Hayek argued that even with the use of "fast" algorithms, the problem required to be solved may be overwhelming and no human or computer could calculate a solution. Von Hayek also argued that the process of placing the "right" information in the hands of the computing and decision making agencies may be very difficult. Since information is dispersed throughout the economy (with no agent having full knowledge of the state of the economy), it must be communicated among the economic agents in order for a solution to be determined. This information exchange may be very costly, and in many cases it may be impossible for a central-control agent to have full knowledge of the state of the economy.

Following the initial results of the 1930s, three major lines of research had a great influence on the development of resource allocation mechanisms and helped to establish a more comprehensive understanding of the main features of such problems:

- activity analysis and linear programming (Dantzig, Kantorovich, Koopmans),
- game theory and iterative solution procedures (von Neumann and Morgenstern, George Brown, Julia Robinson),
- investigation of the relationship between linear/nonlinear programming, two person zero-sum games and Lagrange multipliers (Gale, Kuhn, Tucker).

The early breakthroughs in the field of linear programming greatly influenced the mathematization of classical welfare economics. Although linear programming models are not able to handle goal conflicts due to the multiplicity of consumers as

[3] For a more detailed presentation of the historical aspects of the socialist controversy we refer the reader to [125].

well as constraints arising from the decentralization of information, they are still a very important step in analyzing and understanding multi-objective problems.

Game theory is concerned with the interactive behavior of “rational” man. It is the study of mathematical models of conflict among rational decision makers. Game theory provided general mathematical techniques for analyzing situations where two or more decision makers’ decisions influence one another’s welfare. As such, game theory offered insights of fundamental importance for researchers in many branches of social sciences and technology, including resource allocation mechanisms.

Understanding the interplay between mathematical programming techniques and zero sum games, along with the use of Lagrange multipliers (interpreted as shadow prices in economic systems) helped to develop tools for analyzing general mechanism design problems and provided a better perspective on solution methods.

The original work in classical welfare economics, von Hayek’s observations, and the development of linear programming, game theory and Lagrange multipliers, set the foundations for the formal development of mechanism design in terms of both the “realization” and the “implementation” of Social Choice Rules (SRC’s) (goal correspondences) by decentralized economic systems. Realization theory and implementation theory are the two basic components of mechanism design. We briefly present the key features of realization and implementation theory next.

13.2.2 Realization theory

Formally, resource allocation problems can be described by the following triple: environment, action space, and goal correspondence. We define the *environment* **E** of such problems to be the set of individual endowments, the technology, and preferences, taken together. More generally, the environment is defined as the set of circumstances that cannot be changed either by the designer of the mechanism or by the agents. The *action space* **A** of the problem is considered to be the set of all possible actions, (e.g. resource exchanges) conducted by the various agents. Finally, the *goal correspondence* π is the map from **E** to **A** which assigns for every $e \in \mathbf{E}$ the set of actions in **A** which are solutions to the resource allocation problem.

The setup described above corresponds to the case in which one of the agents has enough information about the environment so as to determine the actions that would satisfy the goal correspondence (i.e. the information in the systems is centralized). Generally this is not the case. Usually, different agents have different information about the environment (i.e. we have an informationally decentralized system). For this reason it is desired/necessary to devise a message exchange process among the various agents that eventually enables them to jointly take an action which corresponds to a solution of the centralized problem. We call such a process of communication, decisions and actions a *resource allocation mechanism*.

The function of a resource allocation mechanism is to guide the agents (economic or otherwise) to make decisions that determine the flow of resources. More specifically, mechanisms provide rules, called *response rules* (or *communication rules*), according to which agents communicate messages to other agents. These messages are generally generated based on the agents' "private" information about the environment and prior messages received from other agents. To provide for a transition from the dialogue to decisions and actions, the mechanism must also have an *outcome rule* which specifies what actions are to be taken given the course of the dialogue. Generally, the mechanism rules may be deterministic or probabilistic; mathematically they are expressed as functions or correspondences.

For simplicity we are going to consider the case in which our mechanism can be represented by a tâtonnement process. This process consists of a communication stage in which agents exchange formal messages in an iterative fashion, followed by a decision process, and finally a translation of decisions into actions. The case in which the communication, decisions, and actions overlap in time (a non-tâtonnement process) require more general theory and will not be discussed here.

The first effort to formally study resource allocation mechanisms can be traced back to Hurwicz's work [40–43]. Hurwicz models the communication process by means of a language and response functions, specifying how each agent determines the message to be emitted at each stage of the iterative exchange of messages. After the process of communication terminates, decisions are determined on the basis of the state of information at the final stage of communication.

Formally, the mechanism model proposed by Hurwicz can be described by the triple $(\mathcal{M}, \mu, h)$: a message space $\mathcal{M}$, an equilibrium message correspondence μ, and an outcome correspondence h. The *message space* is the set of messages that may be exchanged by the agents. The *equilibrium message correspondence* describes the sets of messages that the agents "agree" upon given any particular environment. The *outcome function* describes the set of actions that are taken based on a particular set of "equilibrium" messages. The formulation above is depicted graphically in Fig. 13.1 (cf. [110]).

$$\begin{tikzcd}
\mathbf{E} \arrow[rr, "\pi"] \arrow[dr, "\mu"'] & & \mathbf{A} \\
& \mathbf{M} \arrow[ur, "h"'] &
\end{tikzcd}$$

Fig. 13.1. Message exchange for a decentralized system.

Realization theory is concerned with the existence and design of mechanisms $(\mathcal{M}, \mu, h)$ such that the diagram in Fig. 13.1 commutes. Hurwicz's setup is quite general and can incorporate many types of mechanisms. Given a specified goal correspondence (alternatively called *social choice rule* or *social welfare correspondence*) there may be several mechanisms $(\mathcal{M}, \mu, h)$ such that the diagram of Fig. 13.1 commutes. Each of these mechanisms may have different "communication" and "information processing" characteristics. For example, in a market mechanism with an "auctioneer" the messages exchanged could be prices and demands. In this model the auctioneer updates the prices according to the "excess demand" while the agents update their demands based on the prices. In such a situation the message space $\mathcal{M}$ is small but the "information processing" required until a final action is taken is large, mainly because the process is iterative and the number of required iterations may be very large (in theory they may be infinite). On the other hand, in a "central command" type of mechanism the messages are the signals the agents send to a "central authority" so as to describe their environments. After receiving the agents' messages, the central authority calculates an optimal allocation of resources and sends the order for action to the agents. In this situation, if the space of environments is "rich" the "dimensionality" [69] of the message space required for communication is very large. On the contrary, the amount of information processing is small because it takes only one iteration to implement a centralized decision.

The characterization and classification of mechanisms in terms of their "communication" and "information processing" requirements is an open research area. So far, research concentrated mainly on the "communication" requirements specifically on the "dimensionality" [39–44, 81, 88] of the message space $\mathcal{M}$ required so that the diagram in Fig. 13.1 should commute. Mechanisms $(\mathcal{M}, \mu, h)$ that posses the aforementioned commutative property, have a message space $\mathcal{M}$ of minimum "dimensionality," and satisfy some additional requirements (described below), have been called "informationally efficient." The characterization and comparison of mechanisms according to their "information processing" requirements has received very little attention. In the sequel we will state and discuss more precisely the conditions under which "realization" theory was developed.

The following requirements are generally imposed on Hurwicz's models:

R1. For each element of the environment $e \in \mathbf{E}$ there exists a non-empty set of possible *feasible* actions. The notion of feasibility can usually be split into two categories *individual feasibility* and *compatibility*. In particular:

1. In standard models of production economies, an agent's individually feasible actions are defined to be the set of actions formed by the agent's production function. Within the context of the network problems considered in Section 13.3 a user's individual feasible actions are formed by the set of non-negative demand vectors. On the other hand, the network's individual feasible actions are formed by the set of amounts of services that are delivered and satisfy certain QoS requirements and the network's capacity constraints.
2. We call an action incompatible if given two different input-output vectors of two agents, one calls for an input which the other does not propose to supply. Within the context of the network problems considered in Section 13.3 an action is incompatible if in equilibrium a user requests an amount of service which differs from the amount of service the network intends to supply.

R2. For each element of the environment the set of feasible actions that satisfy the goal correspondence is non-empty.

R3. The actions generated by π must satisfy some sort of optimality criterion. Examples of such criteria are: efficiency of production (defined by Koopmans [65]), optimality (introduced by Pareto [100] under the name of "ophelimity"[4] maximizing), and socially welfare maximizing (defined by Bergson [13], Samuelson [113] and Arrow [6, 8]).

R4. For any environment $e \in \mathbf{E}$, $\mu(e) \neq \emptyset$; that is, for any environment there exists a set of messages to which all agents "agree."

R5. The maps π,μ and h satisfy the following relationship:

$$h(\mu(e)) \subseteq \pi(e) \quad \forall\, e \in \mathbf{E}. \tag{13.1}$$

In other words, for any system environment, in equilibrium, the messages exchanged by the system agents enable the agents to take actions which achieve an optimal centralized solution. Mechanisms satisfying this assumption are also referred to as *non-wasteful*.

R6. The non-wasteful criterion established above can sometimes be inadequate. For example we can have the case where the equilibria of a given process always favors one group of participants at the expense of others. We call such mechanisms *biased*. To avoid biased mechanisms we require *unbiasedness*.

A formal test for unbiasedness can be viewed as follows: Suppose that we think of our process as being formed of two stages. In the first stage the process "distributes parameters" (e.g. resources, information etc.) to the various agents, while in the second stage we have a tâtonnement process. If for any environment $e \in \mathbf{E}$ and any goal realizing action $a \in \pi(e)$ there exists a set of distributional parameters such that at the end of the tâtonnement process the agents take action a, then the process is called unbiased.

R7. For any environment, the rules of the process lead the system to a uniquely determined allocation. This requirement may be difficult to satisfy even in the case of market-based economies. In such economies there may be multiple allocations which are optimal for a fixed set of prices, however, all of these allocations have the same utility for all the agents. We call such processes, where equilibrium indeterminacies are trivial in nature, *essential single-valued*.

R8. There are two types of information regarding the environment agents have access to: direct and indirect. The agent's direct information is obtained through observations of the environment. The indirect information is gathered by the agent through the exchange of messages with other agents. We assume that an agents' direct information is information pertaining only to himself and not to other agents. We will refer to processes satisfying this property as *informationally consistent*.

When considering the equilibrium messages generated by agents, we call a process *privacy preserving* when all the agents generate their messages based only on

[4] Pareto was troubled with the concept of 'utility.' In its common usage utility meant the well-being of the individual or society. Pareto realized that when people make economic decisions they are guided by what they think is desirable for them whether or not that corresponds to their well-being. Thus, he introduced the term "ophelimity" to replace the worn-out 'utility.' Later, preferences replaced Pareto's ophelimity.

their own information about the environment. Hurwicz's model restricts attention to *privacy preserving* resource allocation mechanisms.

R9. Assume that the agents communicate with one another through a *communication alphabet* which permits them to communicate in one shot their direct information (profile) to the other agents. Such a language in most cases is too complex for consideration and hence is undesirable. Restricting the language may also not be enough to alleviate this problem. Agents may be able to encode in a relatively "simple" language their full profile. This type of encoding though may be done by the use of equilibrium message and outcome correspondences which are highly discontinuous and resemble such functions as the Peano space filling curves [5]. Such mechanisms are generally highly unstable (hence undesirable) since minor perturbations/errors in communication will lead to drastically different/nonoptimal actions. In order to alleviate problems such as the above, we introduce the following requirement: We impose extra conditions, such as spot threadiness, on the correspondences μ and h.

Definition 1. *A correspondence* $F : E \twoheadrightarrow M$ *is* spot threaded *if for every* $e \in E$ *there exists an open set* $U_e \subseteq E$*, and a continuous function* $f : U_e \to M$ *such that* $f(e') \in F(e')$ *for all* $e' \in U_e$.

We note that the first three requirements **R1**–**R3** are constraints on the type of problems considered, and they are defined independently of the mechanism. The next four requirements **R4**–**R7** are imposed on the mechanisms to be considered and are generally referred to as *(Pareto) satisfactoriness*[5]. Mechanisms that satisfy **R8**–**R9** are called *regular*.

Realization theory was developed using subsets of the requirements above. To proceed with a more formal description of the results on communication and information processing requirements we need the following definitions:

Definition 2. *We say a mechanism* $(\mathcal{M}, \mu, h)$ *is* goal realizing *if it satisfies requirements* **R1**, **R2**, **R4** *and* **R5**.

Definition 3. *We say that a mechanism* $(\mathcal{M}, \mu, h)$ *is* informationally efficient *if it is goal realizing and regular and it has a message space of a dimensionality which is minimal among all the other goal realizing and regular mechanisms.*

To the best of our knowledge, this definition of informational efficiency is different from that appearing in most of the literature on realization theory. Our definition, compared to the definition appearing in the literature, imposes more requirements on the properties an informationally efficient mechanism must satisfy. We show that the decentralized network resource allocation mechanisms we present in Section 13.3 are informationally efficient according to our definition of informational efficiency.

Most of the research on realization theory has dealt with the discovery of goal realizing mechanisms that have a message space of minimum dimension among the message spaces of goal realizing and regular mechanisms. Some of the key results

[5] In many models requirement **R7** is omitted in the definition of Pareto satisfactoriness.

are: (i) the competitive process is Pareto satisfactory over classical environments[6] [88, 89]; (ii) for classical environments, the competitive mechanisms are goal realizing and have a message space of minimum dimension among the message spaces of goal realizing and regular mechanisms [40, 44, 88][7]; (iii) competitive mechanisms are informationally efficient for classical environments where the utility functions are of the Douglas Cobb form [88]; (iv) for environments with public goods the Lindahl mechanism is goal-realizing and has a message space of minimum dimension among the message spaces of all goal-realizing and regular mechanisms [43, and references therein]; (v) if the dynamics of allocation mechanisms are considered explicitly and stability is required then the size of the message space has to increase [53, 90].

Work in [56, 108, 109] addressed issues related to the complexity of information processing of goal-realizing mechanisms that have message spaces of minimum dimension among the message spaces of goal-realizing and regular mechanisms.

13.2.3 Implementation theory

It is well known that the theory of organizational control systems is concerned with two types of rules: operational and enforcement. The operational rules describe how the system "should" operate, while the enforcement rules assure that the operational rules are followed. Enforcement rules fall within two categories: explicit and implicit. While explicit enforcement rules generally use monitoring techniques in order to control agents' behavior in a system, implicit enforcement motivates agents behavior by providing appropriate incentives.

In the previous section we defined a mechanism to be a set of operational rules, according to which the system's agents generate messages which lead to desired actions. The question that arises is: Can we expect the agents to follow such rules? The answer to the question above is provided by the theory of implementation.

The theory of implementation is generally concerned with strategic behavior of allocation procedures, and generally studies implicit enforcing rules. It is concerned with the design/discovery of "game forms" that implement, in some behavioral equilibrium (solution concept), social choice rules/goal correspondences.

Specifically, N-agent "game forms" are defined as pairs of the form $(\mathcal{M}, h)$, where $\mathcal{M} = \prod_{i=1}^{N} \mathcal{M}_i$, $\mathcal{M}_i$ is the message space of agent i, $i = 1, 2, \ldots, N$, and $h : \mathcal{M} \to \mathbf{A}$. Thus, for each profile $m := (m_1, m_2, \ldots, m_N)$ of messages, $h(m) \in \mathbf{A}$ represents the resulting outcome or allocation. A game form is different from a game as the consequence of a profile of messages is an outcome (allocation) rather than a vector of utility payoffs. Once a preference profile, i.e. a complete, binary and reflexive preordering $\mathcal{R}(e_i)$, that describes the ith agent's preferences over alternatives in $\mathbf{A}$ when i's environment is $e_i \in \mathbf{E}_i$, is specified for each $i = 1, 2, \ldots, N$, a game form induces a game.

[6] A *classical environment* is defined to be a convex economy (i.e. concave utility functions and convex constraint sets), free of externalities (an externality is present when wellbeing of an agent is directly affected by the actions of another agent).

[7] In [40, 44, 88] it is not required that the competitive mechanism should be regular in the whole space of environments.

The principal difference between a game form and Hurwicz's original model, described by $(\mathcal{M}, \mu, h)$, is the following: In game forms the message correspondence (described by μ in Hurwicz's model) is not a design variable, but is induced by the outcome function and the behavioral equilibrium concept (e.g. Nash, Bayesian Nash, maxmin, undominated etc.).

A *solution concept* (or *equilibrium concept*) specifies the strategic behaviors of agents (individuals, users) faced with a game form $(\mathcal{M}, h)$ given a preference profile $\mathcal{R}(e) := (\mathcal{R}(e_1), \mathcal{R}(e_2), \ldots, \mathcal{R}(e_N))$. Hence, a solution concept is a correspondence Λ that identifies a subset of $\mathcal{M}$ for any given specification $(\mathcal{M}, h, \mathcal{R}(e))$. We define

$$\mathfrak{Q}_\Lambda := \{a \in \mathbf{A} | \exists m \in \Lambda(\ (\mathcal{M}, h, \mathcal{R}(e))\ ) \text{ s.t. } h(m) = a\} \tag{13.2}$$

as the set of outcomes associated with the solution concept Λ.

To illustrate (13.2) consider a pure strategy Nash equilibrium as the solution concept. For any given $(\mathcal{M}, h, \mathcal{R}(e))$ a pure Nash equilibrium is a message $m := (m_1, m_2, \ldots, m_N) \in \mathcal{M}$ such that

$$h(m)\mathcal{R}(e_i)h(\overline{m}_i, m_{-i}) \tag{13.3}$$

for all $i = 1, 2, \ldots, N$, and all $\overline{m}_i \in \mathcal{M}_i$, where

$$m_{-i} := (m_1, m_2, \ldots, m_{i-1}, m_{i+1}, \ldots, m_n).$$

Denote the messages satisfying (13.2) by $\mathbf{NE}(\ (\mathcal{M}, h, \mathcal{R}(e))\ )$. Then the set of associated outcomes is

$$\mathfrak{Q}_{\mathbf{NE}}(\ (\mathcal{M}, h, \mathcal{R}(e))\ ) := \{a \in \mathbf{A} | \exists m \in \mathbf{NE}(\ (\mathcal{M}, h, \mathcal{R}(e))\ ) \text{ s.t. } h(m) = a\}. \tag{13.4}$$

To precisely define how social choice correspondences are implicitly enforced via game forms in some behavioral equilibrium we need the following:

Definition 4. *A social choice correspondence $\pi : \mathbf{E} \to \mathbf{A}$ is implemented by the game form $(\mathcal{M}, h)$ via the solution concept Λ if*

$$\mathfrak{Q}_\Lambda(\ (\mathcal{M}, h, \mathcal{R}(e))\ ) = \pi(e)$$

for all $e \in \mathbf{E}$.

Definition 5. *A social choice correspondence $\pi : \mathbf{E} \to \mathbf{A}$ is said to be implementable via the solution concept $\mathfrak{Q}$ if there exists a game form $(\mathcal{M}, h)$ that implements it.*

The form of implementation above is called *full implementation* or *strong implementation* since it requires that the outcomes of a game form coincide with those of the social choice correspondence. A weaker form of implementation (called *weak implementation*) is one where for every $e \in \mathbf{E}$,

$$\mathfrak{Q}_\Lambda(\ (\mathcal{M}, h, \mathcal{R}(e))\ ) \subseteq \pi(e).$$

A comparison between full implementation and weak implementation is presented in Thompson [133].

Within the context of implementation theory there have been significant developments in the characterization of social choice rules that can be implemented in dominant strategies [20, 32]; in Nash equilibria [83–85, 112, 143]; or in refined Nash equilibria such as subgame perfect equilibria [2, 87], undominated Nash equilibria [1, 46, 48, 96], trembling hand perfect Nash equilibria [123]; or in Bayesian Nash equilibria [45, 95, 97, 106]. Excellent survey articles on implementation theory are [47, 85, 94]. These articles summarize the state of the art on implementation theory up to the time of their publication.

Direct revelation game forms (otherwise called *direct mechanisms* or *direct revelation mechanisms*) is a particular class of game forms that have a natural appeal and have received significant attention. In direct revelation game forms $M_i = E_i$ for each agent i. In effect then each agent reports his own environment, but not necessarily his true one. The interest in direct revelation game forms stems from the *revelation principle*. The revelation principle is the observation that if a game form (M, h) implements a social choice correspondence π $(\pi : E \twoheadrightarrow A)$, then there exists a direct revelation game form (E, h^*) which has the following properties: (1) announcing one's true characteristic is an equilibrium message; and (2) $h^*(e_1, e_2, \ldots, e_N) = h^*(e) \in \pi(e)$ for all $e \in E$. Even though the direct revelation game form has the aforementioned properties, it does not necessarily implement the social choice correspondence π. This is because the direct mechanism may have multiple equilibria which give rise to outcomes which, for some $e \in E$ are not in $\pi(e)$ (see [20]). Thus, one cannot conclude from the revelation principle that all one ever needs to consider are direct revelation game forms. Only under certain conditions (see [20]) a social choice rule π can be implemented by a direct revelation game form. Most of the literature on implementation in dominant strategies and in Bayesian equilibria has used *truthful implementation*, an implementation concept that requires *only* that the truthful equilibrium of a direct revelation game form (E, h^*) be in the choice set, i.e., $h^*(e) \in \pi(e) \; \forall \; e \in E$.

13.3 Mechanism Design in Networks: A Realization Theory Point of View

To illustrate how mechanism design can be used in networks we present two classes of network resource allocation problems and discuss them from the realization theory point of view. The two classes of problems are: 1) resource allocation in unicast with routing and end-to-end Quality of Service requirements; and 2) rate allocation in multi-rate multicast service provisioning. For these two problems we present two distinct pricing mechanisms which achieve the solution of the centralized resource allocation problem, satisfy the informational constraints imposed by the decentralization of information in networks, and are informationally efficient. We show how these mechanisms can be embedded into Hurwicz's abstract framework; in [125] we have shown how ideas from realization theory can be used to establish the pricing mechanisms' informational efficiency.

13.3.1 Unicast with routing and Quality of Service requirement

Problem formulation

We consider a set of users/agents, denoted by $\mathbf{N} = \{1, 2, \dots, N\}$, requesting various services from a network. For each user $i \in \mathbf{N}$ we denote by $\mathbf{M}_i$ the set of types of services requested by that user. For each $i \in \mathbf{N}$ each service $j \in \mathbf{M}_i$ must satisfy some sort of end-to-end Quality of Service (QoS) requirements denoted by $\mathcal{F}_{ij}$. Assume that user i's preference over the set of services it requests is summarized by a utility function $U_i(\overline{x}_i)$, where $\overline{x}_i \in \mathbb{R}_+^{\mathbf{M}_i}$. We consider the network to be the $(N+1)^{th}$ agent.

The network is formed by a set of links $\mathbf{L}$. For every $l \in \mathbf{L}$, $\mathbf{K}_l$ denotes the set of resources on link l. Denote by $\mathbf{K} \triangleq \bigotimes_{l \in \mathbf{L}} \mathbf{K}_l$ the set of resources available at the different links of the network, and by the vector $c_{\mathbf{K}}$ the amount resources at those links. Define $\mathbf{T_L}$ to be the topology of the network and $\mathbf{R_{T_L}}$ to be the set of possible routes over which each service requested from the network can be delivered. For each user $i \in \mathbf{N}$ and for each service type $j \in \mathbf{M}_i$ denote by $\mathbb{F}_{ij}(\mathbf{R_{T_L}}, \mathbf{K})$ the set of all resource allocations along all the possible routes of service j that guarantee the end-to-end QoS requirements $\mathcal{F}_{ij}$. Also, for $t \in \mathbf{R_{T_L}}$, $i \in \mathbf{N}$, and $j \in \mathbf{M}_i$ denote by $\mathbb{F}_{ijt}(\mathbf{R_{T_L}}, \mathbf{K})$ the set of all resource allocations along route t that guarantee the end-to-end QoS requirements $\mathcal{F}_{ij}$.

The goal of the network is to allocate resources to the various services in order to maximize a social welfare function described by the sum of the user utilities, while satisfying the QoS requirements imposed by the offered services. Hence, the goal of the network is:

$$\max_{x, \mathbf{R_{T_L}}} \sum_{i \in \mathbf{N}} U_i(\overline{x}_i) \qquad \mathbf{P}$$

subject to:

$$\overline{x}_i \in \mathbb{R}_+^{\mathbf{M}_i}; \qquad \mathbf{P.a}$$

$$r^{i,j} \in \mathbb{F}_{ij}(\mathbf{R_{T_L}}, \mathbf{K}); \qquad \mathbf{P.b}$$

$$\sum_{i \in \mathbf{N}} \sum_{j \in \mathbf{M}_i} x_j^i r_{l,k}^{i,j} \le c_{l,k}; \qquad \mathbf{P.c}$$

For each user $i \in \mathbf{N}$, each service $j \in \mathbf{M}_i$, any set of routes $\mathbf{R_{T_L}}$ and any resource availability $\mathbf{K}$, the set $\mathbb{F}_{ij}(\mathbf{R_{T_L}}, \mathbf{K})$ is well defined, compact and non-empty; **P.d**

The users' utility functions are concave, strictly increasing and continuously differentiable. **P.e**

In **P.b** and **P.c**, $r_{l,k}^{i,j}$ stands for the amount of resource of type k on link l assigned to the service of type j requested by user i, $r^{i,j}$ represents the vector of resources allocated to user i for service j, and $c_{l,k}$ is the amount of resource of type $k \in \mathbf{K}_l$.

In addition, the following informational constraints are present:

The network has no information about the users' utility functions or the number of users interested in services. **P**.**f**

Each user's preferences over the particular services is his private information. The users are unaware of the topology of the network, the amount of resources available on each link, and method by which the network delivers their services. The users are also unaware of the number of other users requesting services from the network, or their utility functions. **P**.**g**

The assumption that the network manager has complete knowledge of the network topology and resources is not an unrealistic one. For example, a corporate intranet or VPN (virtual private network) may have a single provider of resources and services, who is likely to have such knowledge about the network, and who will assume the roll of network management in collecting aggregate excess demand on links and adjusting link prices. In particular, some resource/service providers use very sophisticated network management tools to monitor in real time the proper functions of a network (e.g., events such as congestion, fault, server ups and downs), and to issue appropriate response/commands. Such monitoring requires complete knowledge of the network (e.g., topology, resources, router/link capacities), as well as separate network management protocols to pass information to and from the management site. These tools can easily be used to acquire information on aggregate excess demands and to adjust link prices.

The goal in unicast with routing and QoS requirements is to determine a mechanism that allocates resources in order to generate services for individual users, satisfies the QoS requirements for all the services delivered, is social welfare maximizing[8], and satisfies the aforementioned informational constraints. To achieve this goal we present a market mechanism, which results in a solution of the centralized optimization problem **P–P.e** and satisfies the informational constraints **P.f–P.g**.

Unicast service provisioning has received significant attention. Most of the results on decentralized resource allocation in unicast service provisioning, currently available in the literature are based on pricing mechanisms [17, 19, 22, 33, 49, 54, 62, 64, 76, 78, 79, 91, 92, 103, 104, 131, 142]. These publications have addressed, either by analysis [19, 22, 33, 49, 62, 64, 76, 78, 92, 131, 142], or simulation and analysis [17, 33, 91, 103, 104], a subset of the issues outlined in the goal of the unicast problem stated in the previous paragraph. A significant number of publications have dealt with single link networks [22, 91, 104, 142], or with the allocation of a single resource per connection [22, 33, 49, 64, 78, 91, 92, 104, 142].

Market mechanism

We proceed as follows: First, we describe a competitive market economy consisting of service providers, users and an auctioneer. Then, within the context of this

[8] The social welfare function in this work is characterized by the sum of individual user utility functions.

market we specify a procedure, used by the auctioneer, which leads to a resource allocation that achieves a solution of Problem **P**.

Description of the market

In our market, for conceptual clarity, we assume that the network consists of a service provider and an auctioneer. Under this assumption, the economy consists of the following three types of agents: a service provider, users and an auctioneer. The auctioneer sets the prices per unit of resource at each link. The price of resource k at link l is denoted by $\lambda_{l,k}$. The service provider and the users are price takers. They act as if their behavior has no effect on the equilibrium prices reached by the market allocation process. The service provider uses the network's resources and the prices $\lambda_{l,k}$, specified by the auctioneer, to set up services and the corresponding prices for each unit of these services. Then, it announces the price per unit of service for each service to the users. Based on the announced prices, each user decides the type of services and the amount of each service it should request.

We observe that the price taking assumption and the fact that we try to maximize the sum of the users' utilities imply that: (i) the service providers will not attempt to make a profit; and (ii) the service prices are directly derived from the resource prices.

Below we describe each type of agent in more detail.

Service providers: The users request services from the service providers. Each of these requests is described by the origin, destination and the minimal level of quality of service required. The services are indexed by the (i, j) pair, with $i \in \mathbf{N}$ representing the user and $j \in \mathbf{M}_i$ representing the service type. For each pair (i, j) there exists a set $\mathbf{T}^{i,j}$ of possible routes that can be used. Denote by $\mathbf{V}_{ijt}$ the set of links forming route $t \in \mathbf{T}^{i,j}$. The service provider allocates resources $r^{i,j,t}(\lambda) \in \mathbf{V}_{ijt}$ so that the minimum cost for the service and the lowest acceptable level of quality of service are attained. We assume that each service can not be distributed over multiple routes.

Since the service provider is not a profit maker, it allocates resources for each type of connection by solving:

$$r^{i,j,t}(\lambda) \in \underset{r^{i,j,t} \in \mathbf{F}^{i,j,t}}{\operatorname{argmin}} \sum_{l \in \mathbf{V}_{ijt}} \sum_{k \in \mathbf{K}} \lambda_{l,k} r^{i,j,t}_{l,k} \tag{13.5}$$

where $i \in \mathbf{N}$, $j \in \mathbf{M}_i$, $t \in \mathbf{T}^{i,j}$. For each (i, j) pair, equation (13.5) generates a set of allocations that result in a minimum price per unit of service for each route $t \in \mathbf{T}^{i,j}$. Then, the service provider computes the price per unit of service for route t,

$$p^i_{j,t}(\lambda) = \sum_{l \in \mathbf{V}_{ijt}} \sum_{k \in \mathbf{K}} \lambda_{l,k} r^{i,j,t}_{l,k}, \tag{13.6}$$

where $r^{i,j,t}_{l,k}$ are determined by (13.5). Finally, the service provider computes, for each $i \in \mathbf{N}, j \in \mathbf{M}_i$,

$$p^i_j(\lambda) = \min_{t \in \mathbf{T}^{i,j}} p^i_{j.t}(\lambda) \tag{13.7}$$

and announces the prices $p^i_j(\lambda), i \in \mathbf{N}, j \in \mathbf{M}_i$, to the users. If for some (i, j) there are two or more routes of minimum price, the service provider picks one of these routes.

Users: Users request one way connections from the service provider. Based on the price $p^i_j(\lambda)$, announced by the service provider, the users demand a number of connections determined by

$$x^i(p) \in \operatorname*{argmax}_{x^i \in \mathbf{X}^i} \left[u_i(x^i) - \sum_{j \in \mathbf{M}_i} x^i_j p^i_j(\lambda) \right], \qquad \forall\, i \in \mathbf{N}. \tag{13.8}$$

Auctioneer: The role of the auctioneer is to regulate the prices of the resources. He does this based on the aggregate excess demand vector $z(\lambda, \mathbf{t})$:

$$z_{l,k}(\lambda, \mathbf{t}) \triangleq \sum_{i \in \mathbf{N}} \sum_{j \in \mathbf{M}_i} \left(x^i_j(\lambda) r^{i,j,t}_{l,k}(\lambda) \right) - c_{l,k} \tag{13.9}$$

where $l \in \mathbf{L}$, $k \in \mathbf{K}$, and $r^{i,j,t}_{l,k}$ is determined by (13.5)-(13.7).

The tâtonnement process

We present a tâtonnement process, specified by an algorithm, called **Algorithm 1**, that describes how the market works. The algorithm proceeds iteratively as follows:

Step 1: The auctioneer announces prices λ for the resources at each node of the network. The users announce their desired services to the service provider.

Step 2: Based on the auctioneer's announcement, the service provider computes the minimum price per unit of service according to (13.5)–(13.7). The service provider announces these prices to the users.

Step 3: Based on the prices p announced by the service provider, the users request services in the amount $x(p)$ satisfying (13.8).

Step 4: Based on the service demand vector $x(p)$, the auctioneer computes, through (13.9) the excess demand vector $z(\lambda)$.

Step 5: If $z(\lambda) \leq 0$ then the process ends. Otherwise the auctioneer changes the prices λ of resources according to a specific mechanism based on Scarf's algorithm which is described in detail in [125, 129, 132], announces new prices, say λ', and the process is repeated from Step 2 on.

Embedding into Hurwicz's framework

We embed the unicast routing problem, formulated in Section 13.3.1, within the framework of Hurwicz's model described in Section 13.2.2.

The resource allocation problem:

As presented in Section 13.2.2, a resource allocation problem can be described by the following triple: environment, action space, and goal correspondence.

Environment: "Characteristics" of a particular agent i, say e^i, is called the *local environment* of i. The set of all possible environments of i is denoted by E^i. The (*system*) *environment* is a tuple consisting of the local environments of all agents and is denoted by e. The set of possible system environments is denoted by $E := (\otimes_{i \in \mathbf{N}} E^i) \otimes E^{N+1}$.

For the network Problem **P** the local environment E^i of each user i is the set of differentiable concave functions on $\mathbb{R}_+^{\mathbf{K}}$. The local environment of the network is the set $E^{N+1} \triangleq \{\{\mathbf{T_L}\} \times \{\mathbf{R_{T_L}}\} \times \{\mathbf{K}\} \times \mathbb{F}_{ij}(\mathbf{R_{T_L}}, \mathbf{K}) \times \{c_{\mathbf{K}}\}\}$.

Action Space: The set of possible actions taken by the system is called the *action space* of the system, and is denoted by $\mathbf{A}$. For Problem **P** the action space is the feasible region of **P**.

Goal Correspondence: The relation between the environments and the (desired) actions of the system is represented by a point-to-set map, called the *goal correspondence / social choice rule / social welfare maximizing rule*, and is denoted by π. For Problem **P**, $\pi : E \twoheadrightarrow \mathbf{A}$ is defined as follows:

$$\pi(e) := \text{argmax } P(e). \tag{13.10}$$

Mechanism specification

A mechanism in equilibrium correspondence form is characterized by the following triple $(\mathcal{M}, \mu, h)$, where $\mathcal{M}$ is the message space, μ is the equilibrium correspondence, and h is the outcome function.

Message Space: The set of messages chosen for communication by the designer is called the *message space*, and is denoted by $\mathcal{M}$. The *size of a finite dimensional message space* $\mathcal{M}$ is defined to be the dimension [69] of the smallest real vector space in which there is an open set W such that $\mathcal{M} \subseteq W$. The size of $\mathcal{M}$ is denoted by $\dim \mathcal{M}$.

In the allocation mechanism described above (see [129] for details), two types of messages are exchanged among agents:

- The prices per unit of service for each service are communicated by the network to the users. We denote by $\overline{p} := (p_1, p_2, \ldots, p_q)$ the vector of prices, where q is the number of network services available for delivery.
- The demands for services communicated by the users to the network. We denote by $\overline{x} := \{x_j^i | i \in \mathbf{N},\ j \in \mathbf{M}_i\}$ the vector of user demands.

Thus, the message space for Problem **P** has dimension equal to the number of user demands $\sum_{i+1}^{N} |\mathbf{M}_i|$ (where $|.|$ denotes the cardinality of the set) plus the number of different services supplied by the network.

Equilibrium Correspondence: The relation between the environment and a mechanism's equilibrium messages is represented by a point-to-set map, called *equilibrium correspondence*, and denoted by $\mu(: E \twoheadrightarrow \mathcal{M})$. The *individual equilibrium correspondence* of participant i, denoted by $\mu^i : E^i \twoheadrightarrow \mathcal{M}$, represents a relationship between the local environment of i and the terminal messages emitted by i.

To capture the private nature of the initial distribution of information, we require the following *privacy-preserving* property to be satisfied:

$$\mu(e) = \cap_i \mu^i(e^i). \tag{13.11}$$

To determine the equilibrium correspondence for the first N agents (the users) we define the Lagrangian function $\Lambda(\overline{x}, \lambda)$ (see [12]) of Problem **P**:

$$\Lambda(\overline{x}, \lambda) := \sum_{i \in \mathbf{N}} \sum_{j \in \mathbf{M}_i} x_j^i r_{l,k}^{i,j} + \lambda_{l,k}(c_{l,k} - \sum_{i \in \mathbf{N}} \sum_{j \in \mathbf{M}_i} x_j^i r_{l,k}^{i,j}) \tag{13.12}$$

with $\lambda_{l,k}$ being the Lagrangian multiplier corresponding to the k^{th} resource on the l^{th} link.

Using the first order optimality conditions, we define the equilibrium correspondence of the first N agents (the users) as follows:

$$\mu^i(U_i(\overline{x}_i)) := \{(\overline{p}, \overline{x}) \in \mathcal{M} | \frac{\partial}{\partial \overline{x}_j^i} U_i(\overline{x}_i) - p_{i,j} \leq 0; \overline{x}_j^i(\frac{\partial}{\partial \overline{x}_j^i} U_i(\overline{x}_i) - p_{i,j}) = 0\} \tag{13.13}$$

where $\overline{p}$ is the vector of prices for the services requested by the users, $\overline{x}$ is the vector of demands requested by the users, $\overline{x}_i$ is the vector of demands requested by user i and $\overline{x}_j^i$ is the demand of user i for service j.

From equation (13.12) and the Karush–Kuhn–Tucker (KKT) conditions, the equilibrium correspondence for agent $N+1$ (i.e. the network) is:

$$\begin{aligned} &\mu^{N+1}(\mathbf{T_L} \times \mathbf{R_{T_L}} \times \mathbf{K} \times c_{\mathbf{K}} \times \mathbb{F}_{ij}(\mathbf{R_{T_L}}, \mathbf{K})) \\ &\quad := \{(\overline{p}, \overline{x}) \in \mathcal{M} | \sum_{i \in \mathbf{N}} \sum_{j \in \mathbf{M}_i} x_j^i r_{l,k}^{i,j} \leq c_{l,k},\ r^{i,j} \in \mathbb{F}_{ij}(\mathbf{R_{T_L}}, \mathbf{K}), \\ &\qquad\qquad \lambda_{l,k}(c_{l,k} - \sum_{i \in \mathbf{N}} \sum_{j \in \mathbf{M}_i} x_j^i r_{l,k}^{i,j}) = 0, \\ &\qquad\qquad p_{i,j} = \sum_{l \in L, k \in K} r_{l,k}^{i,j} \lambda_{l,k}\}. \end{aligned} \tag{13.14}$$

Outcome Function: A function which translates messages into actions is called an *outcome function*, and it is denoted by $h(: \mathcal{M} \to \mathcal{A})$. In Problem **P** the outcome function is $h(\overline{p}, \overline{x}) := \overline{x}$.

Key results

The main features of the market mechanism of Section 13.3.1 are:

- The mechanism achieves the solution of the centralized resource allocation problem **P** and satisfies the informational constraints **P.f** and **P.g**. (see [125, 129, 132]).

- The dimension of its message space is a lower bound on the dimension of the message space of any goal-realizing and regular mechanism for routing in unicast with QoS requirements. (See [125].)
- In the case of rate allocation with fixed routes and without QoS the mechanism is informationally efficient.[9] (See [125, 128].)

13.3.2 Multi-rate multicast

Problem formulation

Let $\mathbf{N} = \{1, 2, \ldots, N\}$ denote the set of users/agents requesting various services from a network. We assume that the network consists of a set of L unidirectional links, with a topology denoted by $\mathbf{T_L}$, and each link $l \in L$ having finite capacity c_l. There is a set M of multicast groups. Each multicast group is a tree. Each multicast tree $m \in M$ is specified by $\{s_m, R_m, L_m\}$, where s_m is the unique source node, R_m is the set of receiver nodes, and L_m is the set of links used by the group.

We denote by $R \triangleq \cup_{m \in M} R_m$ the set of all receivers over all the multicast groups, and by $R_{l,m}$ the set of all the receivers of multicast group $m \in M$ using link $l \in L$.

We assume that a unique user is connected to each receiver node $r \in R$. Each user r has a utility function $U_r(x_r)$, where x_r is the rate at which r receives data. This utility function can be interpreted either in terms of the perceived quality of the service received or the amount paid in order to receive the service.

We make the following assumptions:

Assumption 1. *The utility functions $U_r(x_r)$ are strictly concave, differentiable and increasing.*

Assumption 2. *The rates x_r are assumed to be continuous variables.*

Assumption 3. *Rate allocations are done along fixed multicast trees with a fixed number of users.*

Under the assumptions above, we consider the following network multi-rate multicast problem:

$$\max_{x_r, r \in R} \sum_{r \in R} U_r(x_r) \qquad \textbf{Max 1}$$

subject to:

$$\sum_{m \in M} \max_{r \in R_{l,m}} x_r \le c_l, \qquad \forall\, l \in L, \qquad \textbf{Max 1.a}$$

$$x_r \ge 0, \qquad \forall\, r \in R, \qquad \textbf{Max 1.b}$$

[9] Our notion of informational efficiency imposes more requirements on the properties of a mechanism than the standard notion of informational efficiency. To the best of our knowledge an analysis similar to that presented in [125, 128] was conducted only within the framework of production economies where the agent utility functions are of the Douglas Cobb form [88] or quadratic form [81].

the informational constraints **P.f** and **P.g** of Section 13.3.1, and the following additional constraint:

Users are unaware of the method used for service delivery (e.g. unicast vs. multicast). **P.h**

Constraint **Max 1.a** is known as the *capacity constraint*. For this constraint to be satisfied, on each link, the sum of the rates used by each multicast tree can not exceed the link capacity. The capacity constraint ensures that for all the multicast trees, the rate on each branch of a tree is less than or equal to the rate on its parent branch.

Assumption **P.h** is critical in what follows, as it justifies the price-taking assumption we make in the sequel. If the method of service delivery is known to the users, then the mechanism proposed in Section 13.3.2 for the solution of problem **Max 1** together with **P.f–P.g** may not be appropriate for multi-rate multicast service provisioning. This is because in this situation common links have features of public goods, and the mechanism proposed in Section 13.3.2 leads to the "free rider" problem [82, Chapter 11], which in the case of multi-rate multicast service provisioning manifests itself as follows: users who use a common link and demand less than the rate in the link do not participate in the price-sharing of the link (see [125, 126]).

The multi-rate multicast problem with the features above is an informationally decentralized resource allocation problem where there are two distinct types of agents: network (network manager) and users. A major difference between the multi-rate multicast (under the assumptions above) and unicast is the fact that users connected to the same multicast tree receive service over "common links." Thus, to determine optimal (with respect to the performance criterion defined in **Max 1**) rate allocations in multi-rate multicast service provisioning one must find how the prices of "common links" should be shared by their users.

The goal in multi-rate multicast is to develop a mechanism for rate allocation along the various multicast trees in order to: (i) generate services that maximize a social welfare function consisting of the sum of individual user utility functions; and (ii) satisfy the informational constraints **P.f–P.h**.

Multicast service provisioning problems have received significant attention. Within the context of single rate and multi-rate multicast service provisioning, studies have addressed issues of bandwidth/rate allocation [23, 31, 57, 59, 111, 115–119, 122, 134], routing [27, 28, 102, 121, 144] and reliability [30, 34, 60]. Most of the literature on rate allocation is done via the notion of fairness [23, 31, 111, 115–119, 122, 134], specifically, max-min fairness [14] and proportional fairness [63]. In particular, [119] develops a unified framework for diverse fairness objectives via the notion of fair allocation of utilities. A more general approach to rate allocation is via utility maximization. Utility maximizing is more general because rate allocation with the fairness property is utility maximizing when the utility has a special form [23, 86, 119, 122]. Although utility maximization has been extensively studied within the context of unicast rate allocation to achieve congestion control [11, 58, 62, 64, 67, 68, 75, 129, 132], relatively fewer studies approached the multi-rate multicast allocation problem via a general utility maximization formulation, with the notable exceptions being [23, 57, 59]. Problem **Max 1** together with Assumptions 13.3.2–13.3.2 and constraints **Max 1.a**, **Max 1.b**, **P.f–P.h** is similar in spirit to those formulated and analyzed in [23, 57, 59]. However, the decentralized resource allocation mechanism presented in this chapter is different from the mechanisms proposed in [23, 57, 59]. The development here follows [125, 126], where all the details of the proposed algorithm and of the key results can be found.

In the next section we present a market-based pricing mechanism that satisfies the informational constraints imposed by the nature of the network problem, and achieves a solution of the centralized optimization problem **Max 1**. This market mechanism is based on a price splitting algorithm and on properties of price splitting, all of which are presented formally in [126].

Market mechanism

We proceed as follows: We first describe a competitive market economy consisting of two types of agents: network and users. Then, within the context of this market we specify an iterative procedure (a tâtonnement process) which leads to an allocation that achieves a solution to Problem **Max 1**.

Description of the market

The market economy is composed of two types of agents: network (or network manager) and users. The network communicates directly with each user, and the users do not communicate with one another. The messages exchanged by the market agents are service prices and service demands.

For conceptual clarity we decompose the network manager into two distinct entities: service provider and auctioneer. The market features and the relations among the market agents are as follows: The resource traded at each link is the available communication rate. The rate price at link $l \in L$ is denoted by λ_l. The prices λ_l, $l \in L$, are set by the auctioneer. Based on λ_l, $l \in L$, the service provider sets up prices per unit of rate along each path of each multicast tree and communicates these prices to the users. Based on the service prices announced by the service provider the users demand a certain amount of service from the network in order to maximize their utility functions. Based on user demands the auctioneer updates the price per unit of rate at each link of the network.

We make the assumption that the service provider and users are price takers. They act as if their behavior has no effect on the equilibrium prices reached by the market allocation process. As pointed out in Section 13.3.2, this assumption is justified by **P.h**, that is, the fact that the users are unaware of the type of service received and they do not know the number of other users requesting service from the network. The price taking assumption and the fact that we try to maximize the users' utilities imply that: (i) the service provider will not attempt to make a profit; and (ii) the service prices are directly derived from resource prices.

Below we describe each type of agent in more detail.

Service provider: The service provider receives from the auctioneer a rate price λ_l for each link l of the network. Based on these prices, it has to compute the price per unit of rate for each user.

A major challenge in solving multi-rate multicast problems through pricing is the determination of the set of user service prices from the set of link prices. This challenge comes from the fact that for each link which is common to multiple users of a multicast tree one needs to determine the portion of the price which is incurred by each of the users sharing the link. These price shares need to be determined in a way that satisfies the informational constraints imposed by the nature of the network multi-rate multicast problem.

In [126] we present a distributed algorithm which for a fixed set of link prices λ computes a set of link price shares $\gamma(\lambda)$. Based on these price shares $\gamma(\lambda)$ the algorithm also computes the service prices $p(r, \lambda) \triangleq p(r, \gamma(\lambda))$ which generate demands that maximize the total user utility along any multicast tree for the fixed set of link prices λ.

Users: Users are price takers and request service from the service providers. For each user r of the multicast tree $m \in M$ the service provider announces a service price $p(r, \lambda)$. Based on $p(r, \lambda)$, user r determines its desired service rate by solving:

$$x_r(p(r,\lambda)) \triangleq \underset{x}{\operatorname{argmax}}\{U_r(x) - p(r,\lambda) \times x\}. \tag{13.15}$$

Auctioneer: The role of the auctioneer is to regulate the prices of resources, based on the aggregate excess demand vector $z(\lambda)$,

$$z_l(\lambda) \triangleq \sum_{m \in M} \max_{r \in R_{l,m}} x_r(p(r,\lambda)) - c_l \tag{13.16}$$

at every link $l \in L$.

The tâtonnement process

We present a tâtonnmment process, described by an algorithm, called **Algorithm 2**, that describes how the market works. The algorithm proceeds iteratively as follows:

Step 1: The multicast trees are fixed.

Step 2: The auctioneer announces prices $\lambda := \{\lambda_l, l \in L\}$ per unit of rate at each link of the network.

Step 3: The service provider receives the link prices λ announced by the auctioneer. Given the link prices λ, the service provider communicates with the users via an iterative process in order to determine the optimal service prices $p(\lambda) := \{p_i(\lambda), i = 1, 2, \ldots, N\}$. During the iterative process the service provider and the users exchange prices per unit of service p and service demands $x(p)$, with $x(p)$ satisfying (13.15). This iterative process is described in detail in [126, Section IV, Appendix A]. During the iterative process between the service provider and users, the auctioneer checks if the sign of the excess demand function $z(\lambda)$ is positive on some link or negative on all links.

Step 4: If at the end of Step 3 $z(\lambda) \leq 0$ the process ends. Otherwise the auctioneer changes the prices λ of resources according to a specific mechanism based on Scarf's algorithm (which is described in detail in [125, 126, 132]), announces new prices, say λ', and the process is repeated from Step 3 on.

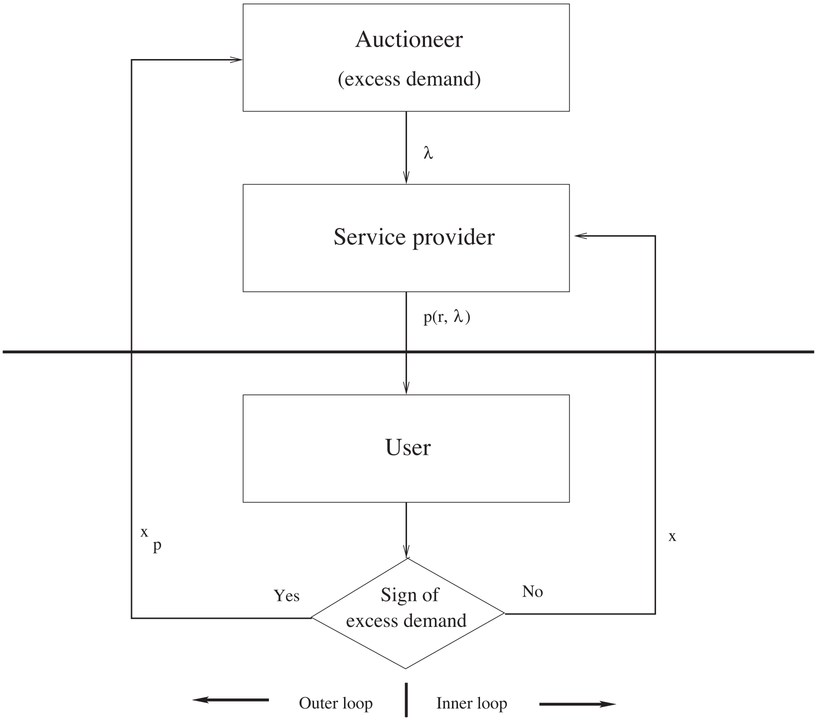


Fig. 13.2. Market mechanism.

The steps above are pictorially shown in Fig. 13.2. The figure illustrates the fact that the algorithm contains two loops: an outer loop and an inner loop. The inner loop describes the iterative process used by the service provider to determine user service prices $p(\lambda)$ (hence user demands) for fixed link prices λ set by the auctioneer. For fixed λ the inner loop also determines how prices of links that are common to many users are optimally shared by these users. The outer loop determines the iterative process used by the auctioneer to determine link prices based on excess demand. The iterative process of the inner loop is guided by the results of [126, Section III] and is presented in [126, Section IV, Appendix A]. The iterative process of the outer loop is guided by Scarf's Algorithm [120].[10]

Embedding into Hurwicz's framework

We embed the multi-rate multicast problem, formulated in Section 13.3.2, within Hurwicz's abstract framework described in Section 13.2.2.

The resource allocation problem:

Environment: For the network Problem **Max 1**, the local environment E^i of each user i is the set of differentiable concave functions on $\mathbb{R}_+$. The local environment of the network is the set $E^{N+1} \triangleq \{\{\mathbf{T_L}\} \times \{L\} \times \{M\} \times \{c_L\}\}$. The *system environment* is denoted by $E := (\otimes_{i \in \mathbf{N}} E^i) \otimes E^{N+1}$.

Action Space: The action space is the feasible region of Problem **Max 1**.

Goal Correspondence: For problem **Max 1**, $\pi : E \twoheadrightarrow \mathbf{A}$ is defined as follows:

$$\pi(e) := \text{argmax } \textbf{Max 1 } (e). \tag{13.17}$$

The environment, action space and goal correspondence describe the resource allocation problem.

Mechanism specification

Message Space: In the pricing mechanism proposed in Section 13.3.2 for solving Problem **Max 1**, two types of messages are exchanged among agents:

- To each user i the network communicates a service price p_i.
- Each user i communicates a service demand x_i to the network.

Thus, the message space for Problem **Max 1** has dimension equal to the number of user demands $\sum_{m \in M} |R_m|$ (where $|.|$ denotes the cardinality of the set) plus the number of service prices[11].

Equilibrium Correspondence: Using the first order optimality conditions, we define the equilibrium message correspondence of the first N agents (the users) as follows:

$$\mu^r(U_r(x_r)) := \{(p_r, x_r) \in \mathcal{M} | \frac{\partial}{\partial x_r} U_r(x_r) - p_r \leq 0; x_r(\frac{\partial}{\partial x_r} U_r(x_r) - p_r) = 0\}. \tag{13.18}$$

[10] It may be possible to use algorithms other then Scarf's at the outer loop, however, to prove convergence of such algorithms we may need to impose additional constraints on the users' utility functions (e.g. second order differentiability of the utility functions).

[11] In this setup since no two services are identical (i.e. no two services are part of the same multicast tree and are delivered over the same links) the number of service prices is equal to the number of user demands.

To present the equilibrium message correspondence for the network, we consider the following problem:

$$\max_{x_r, r\in R} \sum_{r\in R} U_r(x_r) \qquad \textbf{Max 2}$$

such that:

$$\sum_{m\in M} x_{r_{l,m}} \leq c_l, \qquad \forall\, l \in L,\ \forall\, r_{l,m} \in R_{l,m}. \tag{13.19}$$

$$x_{r_{l,m}} \geq 0, \qquad \forall\, l \in L,\ \forall\, r_{l,m} \in R_{l,m} \tag{13.20}$$

where $r_{l,m}$ denotes a receiver on the mth multicast tree that employs link l.

Let $|M|$ denote the number of multicast trees in the network. We define the set $\Phi(l) \triangleq \{\big(r_{l,1}, \ldots, r_{l,|M|}\big) : r_{l,i} \in R_{l,i}, 1 \leq i \leq |M|\}$, to be the set of $|M|$-tuples, each tuple consisting of one receiver from each multicast tree, and every receiver of each tuple is downstream from link $l \in L$ on its respective multicast tree. We note that the number of elements in $\Phi(l)$ corresponds to the number of constraints for link l in the set of equations (13.19). We denote by r_l an element of $\Phi(l)$, and by $r_{l,m}$ a receiver on the m^{th} multicast tree of r_l. Note that if for some multicast tree $m \in M$ and some link $l \in L$, $R_{l,m} = \emptyset$, i.e. link l is not part of the multicast tree m, then we let the $r_{l,m}$ entry of the r_l tuple be empty, i.e. no receiver from multicast tree m is assigned to any of the r_l tuples. We define the set $\Phi(l,r) \triangleq \{\big(r_{l,1}, \ldots, r_{l,|M|}\big) : r \in \big\{r_{l,1}, \ldots, r_{l,|M|}\big\}, r_{l,i} \in R_{l,i}, 1 \leq i \leq |M|\}$ to be a subset of $\Phi(l)$ where all the tuples contain receiver r.

Using the notation above we can rewrite equation (13.19) as follows:

$$\sum_{m\in M} x_{r_{l,m}} \leq c_l, \qquad \forall\, l \in L,\ \forall\, r_l \in \Phi(l).$$

Then, the Lagrangian function for Problem **Max 2** can be expressed as:

$$\Lambda(x,\gamma) \triangleq \sum_{r\in R} U_r(x_r) - \sum_{l\in L} \sum_{r_l\in\Phi(l)} \zeta_{r_l}\big(\sum_{m\in M} x_{r_{l,m}} - c_l\big) \tag{13.21}$$

where $\zeta \triangleq \{\zeta_{r_l} : \zeta_{r_l} \in R_+, r_l \in \Phi(l), l \in L\}$.

Consequently, the first order optimality conditions are:

$$\zeta_{r_l}\big(\sum_{m\in M} x_{r_{l,m}} - c_l\big) = 0, \tag{13.22}$$

$$\Big(\frac{\partial U_r(x_r)}{\partial x_r} - \sum_{l\in\mathcal{L}_r} \sum_{r_l\in\Phi(l,r)} \zeta_{r_l}\Big) = 0 \tag{13.23}$$

where $\mathcal{L}_r$ is the set of links connecting receiver r to the source.

From equations (13.22) and (13.23), the equilibrium correspondence for agent $N+1$ (i.e., the network) is:

$$\begin{aligned}&\mu^{N+1}(\{\mathbf{T_L}\} \times \{L\} \times \{M\} \times \{c_L\})\\ &\quad := \{(\overline{p}, \overline{x}) \in \mathcal{M} | \sum_{m\in M} \max_{r\in R_{l,m}} x_r \leq c_l,\ \zeta_{r_l}(\sum_{m\in M} x_{r_{l,m}} - c_l) = 0,\\ &\qquad\qquad p_r = \sum_{r_l \in \Phi(l,r)} \zeta_{r_l}\}\end{aligned} \tag{13.24}$$

where $\overline{p} := (p_1, p_2, \ldots, p_{|R|})$ is the vector of prices for the services requested by the users, and $\overline{x} := (x_1, x_2, \ldots, x_{|R|})$ is the vector of demands requested by the users.

Outcome Function: In Problem **Max 1** the outcome function is $h(\overline{p}, \overline{x}) := \overline{x}$.

Key results

The main features of the market mechanism of Section 13.3.2 are:

- It achieves the solution of the centralized problem **Max 1**, and satisfies the informational constraints **(P.f)** and **(P.g)** (see [125, 126]).
- It is informationally efficient (see [125]).

13.3.3 Discussion

In Sections 13.3.1 and 13.3.2, we presented an approach for optimal resource allocation for both unicast with routing and QoS requirements, and multi-rate multicast service provisioning. The main features of this approach are:

(1) The objective to maximize the total value of the network to its users.
(2) The agents are price takers in the markets in which they participate.
(3) The users' utility functions u_i are quasi-linear, continuously differentiable, and strictly concave.
(4) There is no cost associated with the supply of network resources.

We now briefly discuss and critique each one of the features above separately. For more details we refer the reader to [125, 126, 129, 132].

(1) In the problems considered we assumed that the objective function of interest was to maximize the sum of individual network users' utility functions. It may not be obvious why this is a reasonable objective to consider.

It is important to realize that our point of view is primarily *normative*, not *descriptive*. That is, we have taken a particular objective function—one which we believe is often reasonable—and studied whether a network resource pricing scheme exists that can achieve an optimum for that particular function, and how one might implement that allocation with a market-based algorithm. Thus, we have demonstrated the feasibility of using pricing to achieve a particular performance goal. We are not claiming that this goal describes any particular actual network environment. Nor are we making the stronger normative claim that this objective function *should* be adopted in any particular setting.

We do, in fact, believe that maximizing the sum of user utilities is a reasonable description for a wide variety of network allocation problems. Suppose we are considering a corporate intranet. If the corporation's overall objective is to maximize its profits (in present value), then the appropriate interpretation of our problem is to define each user's "utility" as that user's contribution to corporate profit as a function of the network services it consumes. In other words, the corporation is not (directly) interested in how personally happy an employee is with the network, but on how much the network enhances the employee's productivity. Then the sum of user utilities will be the contribution of network services to corporate profits, which is precisely the firm's objective function for this part of the overall management problem. Although it may seem difficult to come up with a reasonable representation of the effect of network services on each user's contribution to corporate profits, at some level this is precisely the problem corporations need to solve for allocating equipment, office space, subordinates and so forth to each employee—it is well beyond the scope of our research to worry about how the corporation specifically formulates these valuation functions.

Thus, although our method of using prices to allocate network resources cannot be directly applied to every allocation problem with any reasonable objective function, we believe that it has broad applicability to many existing situations. In any case, when our objective function *is* the desired goal, we have carefully analyzed the existence and implementability of a pricing scheme to support that objective.

(2) For the problems presented in Sections 13.3.1 and 13.3.2, we have imposed the price-taking assumption to the agents of the market economy. How useful is the price-taking assumption? It is not essential for a proof that an algorithm exists that will clear the markets and reach some equilibrium allocation of network resources. However, in general, that allocation will not be a solution of our original optimization problem.

As a general matter we could show that equilibrium allocations based on behavior other than price-taking will lead to less efficient allocations, that is, allocations that do not maximize the sum of user utilities subject to the technology constraints. Therefore, we did not consider markets in which agents exhibit different types of strategic behavior, but limited ourselves to the price-taking behavior that we can show can be harnessed to yield a solution to the centralized optimization problem.

Restricting attention to the price-taking case may not in practice be as restrictive as it seems. Consider the example of a corporate intranet with a single monopoly provider of resources and services. If the management instructs the resource and service provider to behave "as if" it is a price taker (and provides compensation incentives that make it in the provider's best interests to do so) then the desired outcome can be achieved. Essentially, this requires compensating the provider based on the value of the allocation to the company as a whole, rather than based on the provider's own local "profits." If the network is to be managed with an agent-based control system, the agents should be programmed to act as price takers, whether or not other programmable strategies might seem more desirable from the local viewpoint of the agents.

In a more open, conventionally market-based system, such as a commercial market for virtual data circuits, it is also possible that at a given moment some participants might have some market power, which is to say that they are cognizant of an opportunity to improve their position by acting strategically with respect to price-setting, rather than as a price taker. In such a setting, it might not be possible to directly control behavior to make those participants behave "as if" they are price takers. However, if there are no artificial barriers to entry by other providers—for example, if it is possible for another competing firm to build an interconnected network of links with buffers and bandwidth—then it will tend to be the case that in a long-run equilibrium surviving agents will be those who behave as price takers (competition will drive others out of the market). Therefore, we believe there are many circumstances under which the conditions will exist, or can be imposed, that are necessary for our approach to provide an equilibrium that is a social welfare maximizing solution of the centralized network problem.

The price-taking assumption is harder to justify in multi-rate multicast service provisioning than in unicast with routing. Treating users as price-takers in Problem **Max 1** is reasonable under the assumption that they are unaware of the method of service delivery (assumption **P.h**). Without such an assumption, a formulation of multi-rate multicast as a public goods problem, albeit a non-typical one, may be more appropriate than that of Section 13.3.2. The investigation of Problem **Max 1** without assumption **P.h** remains an interesting open problem.

(3) Since we assume that the expenditure of the good under study is a small portion of a consumer's total expenditure, the small size of the market under study should lead the prices of the other goods to be approximately unaffected by changes in this market. Because of this fixity of other prices, we are justified in treating the expenditure on these other goods as a single composite commodity, which we call the numeraire. This allows us to express the utility function as a function of the goods under study and the numeraire.

The choice of representing users' preferences by quasi-linear objective functions also imposes the constraint that there are no income effects on network service demand; that is, changes in income or budget available to the users does not change the amount of network services they wish to purchase. This is a typical simplifying assumption in the economic literature when the budget share of the services of interest is small, e.g. when network services are only a relatively small amount of the users' total expenditures.

The rest of the assumptions made for the utility functions are normal assumptions usually made in analysis of economic optimization problems. The continuously differentiable assumption comes from the idea that we may look at a set of users that may have similar utilities as a group, and in this case the group utility will be a smoothed out version of each user's utility. Strictly concave assumption is natural when we are working with goods that are desirable.

(4) In both network problems considered in Sections 13.3.1 and 13.3.2, we assumed that there is no cost in supplying network resources (bandwidth, buffers, etc.) to the market. This cost can be incorporated into our model if we subtract it from the objective function of the optimization problem. We believe that the new problem will have the same qualitative properties with the problem presented in this chapter, thus it may lead to a similar type of result.

13.4 Mechanism Design in Networks: An Implementation Theory Point of View

In the previous section we considered a mechanism to be a set of rules which, if followed, generate allocations that satisfy a goal correspondence. This kind of mechanisms ignore issues of strategic behavior of individual agents. Thus, it may not be possible to contractually enforce such mechanisms. To design resource allocation mechanisms that are contractually enforceable we have to take into account the divergence of individual preferences from the overall performance objective.

In this section we discuss game forms that implement social choice rules characterized by socially welfare maximizing solutions. We concentrate on Nash implementation and relate our discussion to the unicast problem with routing.

There are two distinct ways in which one can think of implementation of social welfare maximization rules in Nash equilibria. We present them below.

When Nash implementation is the solution concept, an individual (user) needs to know not only his own preferences, but everyone else's preferences so as to determine his equilibrium message(s). Thus, for Nash implementation purposes in unicast and routing, an environment of a user is an entire profile of utility functions and $\{\{\mathbf{T_L}\} \otimes \{\mathbf{R_{T_L}}\} \otimes \{\mathbf{K}\} \otimes \mathbb{F}_{ij}(\mathbf{R_{T_L}}, \mathbf{K}) \otimes \{c_{\mathbf{K}}\}\}$, defined in Section 13.3.2. Consequently, the space of user's environments is

$$\hat{E} := (\otimes_{i\in \mathbf{N}} E^i) \otimes \{\{\mathbf{T_L}\} \otimes \{\mathbf{R_{T_L}}\} \otimes \{\mathbf{K}\} \otimes \mathbb{F}_{ij}(\mathbf{R_{T_L}}, \mathbf{K}) \otimes \{c_{\mathbf{K}}\}\}, \quad (13.25)$$

where all the components of the right-hand-side of (13.25) are defined in Section 13.3.2. When the action space $\mathbf{A}$ is the feasible region of Problem $\mathbf{P}$, and the message space $\mathcal{M}$ is

$$\mathcal{M} = \hat{E} \times \mathbf{A} \ \times \mathbb{N} \quad (13.26)$$

where $\mathbb{N}$ is the set of natural numbers, then the goal correspondence $\pi : \otimes_{i=1}^{N} E^i \twoheadrightarrow \mathbf{A}$, described by the centralized solution of Problem $\mathbf{P}$, can be implemented in Nash equilibria by a game form $(\mathcal{M}, h)$ where the outcome function h is defined in [84, Theorem 3]. Such an implementation is possible for the following reason. For the unicast problem with routing, π is a Pareto correspondence; Pareto correspondences are *monotonic* and possess the *no veto power* property [83, 84]; therefore, π can be implemented in Nash equilibria by the aforementioned game form whenever the number of users in the network is greater than or equal to three [83, 84]. However, the game form described above is infinite dimensional. Thus, the approach above to Nash implementation leads to game forms that are infeasible on information grounds.

An alternative way of proceeding with Nash implementation of the goal correspondence π (which could potentially result in game forms with finite dimensional message spaces) is the following. Consider that users know their own environment, but not those of other users or the network. All users are involved in an unspecified message exchange process in which they grope their way to a stationary message and in which the Nash property is a necessary condition for stationarity. Experimental evidence [124] has shown that such an approach to Nash implementation is reasonable.

An important open issue within the context of the second approach to Nash implementation is: What is the minimum dimensionality of the message space of game forms that implement social welfare maximizing rules (e.g. π) in Nash equilibria? It is expected that, in general, an implementing mechanism with the Nash property in equilibrium messages will require a larger message space than the one that suffices for decentralized realization without regard to individual incentives. Reichelstein and Reiter [107] have shown that the statement above is true in the case of Nash implementation of Walrasian allocations in exchange environments. The following example from [107] illustrates the fact that Nash implementations require larger message spaces than the corresponding decentralized realizations.

Example 4.1

Consider a resource allocation problem with two agents $\{1, 2\}$ and two goods $\{X, Y\}$, where good X represents a desirable service and good Y has the interpretation of money. Assume that the agents' preference over the goods are described, at least locally, by quasi-linear[12] convex utility functions of the form

$$U_i(x, y|e_i) \triangleq e_i \times x - \frac{x^2}{2} + y, \quad e_i \in E_i,\ i \in \{1, 2\}.$$

The agents' private objective is to maximize their individual utility function, while the social objective is to achieve a resource allocation which maximizes the sum of individual utility functions.

Assume that the goods are distributed among the agents and the agents are permitted to trade. From the realization theory point of view there exists a goal realizing mechanism with a message space of dimension two [43]. Specifically, in the case in which both agents are truthful, the mechanism in which agent 1 sets the price for good X and agent 2 makes a request[13] for good X based on the price set by agent 1 is social welfare maximizing.

In [107] the authors show that there is no mechanism of dimension 2 which implements in Nash equilibria the social welfare maximizing rule for this problem. In particular, they show that the pricing mechanism where agent 1 sets the price

[12] Let $X := \{x_1, x_2, \ldots, x_L\}$ be a set of commodities. A function $U(x)$ is called *quasi-linear* with respect to commodity L if it is of the form $U(x) = \mathcal{U}(x_1, x_2, \ldots, x_{L-1}) + x_L$. Commodity x_L is called the *numeraire commodity*. The numeraire commodity generally has the interpretation of money.

[13] If agent 2 makes a negative request for good X it means that he would like to sell that amount of good X to agent 2.

for commodity X and agent 2 makes a request for this commodity based on the price, does not have a Nash equilibrium which is social welfare maximizing. The authors present a mechanism of dimension 3 which implements the social welfare maximizing rule in Nash equilibria. This mechanism works as follows:

The agents message spaces are described by

$$M_1 = \{m_1 | m_1 \in \mathbb{R}_+\}, \tag{13.27}$$

$$M_2 = \{(m_2, m_3) | (m_2, m_3) \in \mathbb{R}^2_+\}. \tag{13.28}$$

The amount of commodity X exchanged is characterized by the outcome functions

$$h^x_1(m) = m_1 - m_2, \tag{13.29}$$

$$h^x_2(m) = m_2 - m_1, \tag{13.30}$$

while the amount of the numeraire commodity Y exchanged by the agents is described by:

$$h^y_1(m) = -m_3(m_1 - m_2), \tag{13.31}$$

$$h^y_2(m) = -m_1(m_2 - m_1) - (m_1 - m_3)^2. \tag{13.32}$$

In this mechanism m_3 has the interpretation of the price of good X. Agent 1 maximizes his utility function based on m_3 and sends his message m_1 to agent 2. Agent 2 does not use m_3 as the price for commodity X, but rather it uses message m_1. Based on m_1, agent 2 maximizes his utility by choosing m_2. Agent 2 receives a quadratic penalty for announcing a price which is not equal to m_1. This penalty, along with the fact that he does not set his own price for good X, forces agent 2 to be truthful in his messages. In [107] the authors prove that this mechanism implements the social welfare maximizing rule in Nash equilibria. □

Recent game theoretic studies in network unicast problems (without routing)[35, 50–52, 114] have shown that: When the dimension of the message space of the game form is the same as that of the pricing mechanism which suffices for decentralized realization, the game form does not implement the social welfare maximizing rule. Specifically the Nash equilibria determined in [35, 50–52, 114] are not social welfare maximizing.

Example 4.1, the results in [35, 50–52, 114], as well as the results on implementation in other solution concepts such as Bayesian Nash equilibria [98], and refinements of Nash equilibria (specifically, subgame perfect equilibria [87], and undominated Nash equilibria [18]) reveal that:

I1: Games that are induced by game forms whose message space has the same dimension as that of the standard pricing mechanism have multiple equilibria, some of which do not result in welfare maximizing solutions. Consequently, such game forms can not implement (in the corresponding solution concept) social welfare maximizing rules.

I2: Implementation of social welfare maximizing rules in some solution concept (cf. Section 13.2.3) requires a message space the dimension of which is larger than that of the message space which suffices for decentralized realization.

I3: The increase in the dimension of the message space must accomplish the following:

1. It must eliminate the equilibria that do not result in welfare maximizing allocations (cf. **I1**).
2. It must maintain the equilibria that result in welfare maximizing allocations.
3. If must not introduce additional equilibria, unless these equilibria result in welfare maximizing allocations.
4. It must induce price-taking behavior among the players (users).

We have the following conjecture concerning the Nash implementation of the centralized solution of the unicast network resource allocation problem with routing and QoS requirements.

Conjecture: In the case of unicast service provisioning, with N users and L services, there exists a game form which implements the centralized solution of Problem **P** in Nash equilibria and has a message space of dimension equal to the dimension of the pricing mechanism plus $\lceil \frac{L}{N} \rceil$. Any mechanism with a message space of smaller dimension can not implement the centralized solution of Problem **P** in Nash equilibria [127].

13.5 Conclusion

Our goal was to discuss: (i) how decentralized network resource allocation problems fit within the context of mechanism design; and (ii) how mechanism design can provide guidelines for the determination of resource allocation strategies that realize (in an informationally efficient manner) social welfare maximizing resource allocation rules, and implement them in some appropriate behavioral equilibrium concept (e.g. Nash equilibrium) in an informationally efficient manner. Our discussion was guided by two classes of network resource allocation problems (unicast with routing and QoS requirements, and multi-rate multicast) that received significant attention in the engineering world. The results we presented reveal the connection between network resource allocation and mechanism design. The discussion also revealed that: (1) the aforementioned network problems are better understood from the realization theory point of view than from the implementation theory viewpoint. (2) A formulation that is appropriate for the multi-rate multicast problem when the users are aware of the method of service delivery remains an interesting open problem.

In our opinion, two problems of fundamental importance are: (1) the characterization and classification of mechanisms in terms of their "communication" and "information processing" requirements; and (2) how can the theory of implementation guide the design of minimal message space mechanisms that implement, in some appropriate solution concept, social welfare maximizing network resource allocation rules.

Acknowledgment. This research was supported in part by ONR Grant N00014-03-1-0232. The authors gratefully acknowledge stimulating discussions with Jeffrey MacKie-Mason and John Ledyard.

References

[1] D. Abreu and H. Matsushima, Virtual implementation in iteratively undominated strategies: Complete information, *Econometrica*, **60** (5):993–1008, 1992.

[2] D. Abreu and A. Sen, Subgame perfect implementation: A necessary and almost sufficient condition, *Journal of Economic Theory*, **50**:285–299, 1990.

[3] M. Allais, *A la Recherche d'une Discipline Economique, I*, Paris, 1943.

[4] M. Allais, *Economie Pure et Rendement Social*, Paris, 1945.

[5] T. Apostol, *Mathematical Analysis*, Addison-Wesley, 1974.

[6] K. Arrow, A difficulty in the concept of social welfare, *Journal of Political Economy*, **58**:328–346, 1950.

[7] K. Arrow, An extension of the basic theorems of welfare economics, In *J. Neyman, Ed., Proceedings of the Second Berkeley Symposium*, 507–532, Berkeley, CA, 1951.

[8] K. Arrow, *Social Choice and Individual Values*, Yale University Press, 1951.

[9] K. Arrow and F. Hann, *General Competitive Analysis*, Holden-Day, San Francisco, 1971.

[10] E. Barone, Il ministro della produzione nello stato collettivista, *Giornale degli Economisti*, 1908.

[11] Y. Bartal, J. Byers and D. Raz, Global optimization using local information with applications to flow control, In *Proceedings of the* 38^{th} *Ann. IEEE Symp. on Fundations of Computer Science (FOCS)*, Miami, FL, October 1997.

[12] M. Bazaraa, H. Sherali and C. Shetty, *Nonlinear Programming Theory and Algorithms*, John Wiley, New York, 1993.

[13] A. Bergson, A reformulation of certain aspects of welfare economics, *Quarterly Journal of Economics*, **52**:310–334, 1938.

[14] D. Bertsekas and R. Gallagher, *Data Networks*, 2nd Edition, Prentice Hall, Englewood Cliffs, 1992.

[15] N. Bukharin, *Economics of the Transition Period*, 1920.

[16] N. Bukharin and E. Preobrazhensky, *The ABC of Communism*, Penguin Books, 1969.

[17] R. Cocchi, D. Estrin, S. Shenker and L. Zhang, Pricing in computer networks: motivation, formulation and example, *IEEE/ACM Transaction on Networking*, **1** (6):614–627, December 1993.

[18] L. Corchon, *The Theory of Implementation of Socially Optimal Decisions in Economics*, McMillan, New York, 1996.

[19] C. Courcoubetis, F. Kelly and R. Weber, Measurement-based usage charges in communications networks, *Operations Research*, **48** (4):535–548, 2000.

[20] P. Dasgupta, P. Hammond and E. Maskin, The implementation of social choice rules: Some general results on incentive compatibility, *Review of Economic Studies*, **46**:185–216, 1979.

[21] T. de Scitovsky, A note on welfare propositions in economics, *Review of Economic Studies*, **9**:77–88, 1941.

[22] G. de Veciana and R. Baldick, Resource allocation in multiservice networks via pricing, *Computer Networks and ISDN Systems*, **30**:951–962, 1998.

[23] S. Deb and R. Srikant, Congestion control for fair resource allocation in networks with multicast flows, *IEEE/ACM Transactions on Networking*, **12** (2):261–273, 2004.

[24] G. Debreu, Coefficient of resource utilization, *Econometrica*, **19**:273–292, 1951.

[25] G. Debreu, *Theory of Value*, John Wiley, New York, 1959.

[26] H. Dickinson, Price formation in a socialist community, *The Economic Journal*, 237–250, 1933.

[27] M. Donahoo, K. Calvert and E. Zegura, Center selection and migration for wide-area multicast routing, *Journal of High Speed Networks*, **6** (2), 1997.

[28] M. Donahoo and E. Zegura, Core migration for dynamic multicast routing, In *International Conference on Computer Communications and Networks (IC-CCN)*, Washington, DC, 1996.

[29] M. Doob, Economic theory and the problems of a socialist economy, *The Economic Journal*, 588–598, 1933.

[30] N. Duffield, J. Horowitz, D. Towsley, W. Wei and T. Friedman, Multicast-based loss inference with missing data, *IEEE Journal on Selected Areas in Communications*, 2002.

[31] E. Graves, R. Srikant and D. Towsley, Decentralized computation of weighted max-min fair bandwidth allocation in networks with multicast flows, In *Proceedings Tyrrhenian International Workshop on Digital Communications (IWDC)*, Taormina, Italy, 2001.

[32] J. Green and J. Laffont, *Incentives in Public Decision Making*, North-Holland, Amsterdam, 1979.

[33] A. Gupta, D. Stahl and A. Whinston, A stochastic equilibrium model of internet pricing, *Journal of Economic Dynamics and Control*, **21**:697–672, 1997.

[34] R. Gupta and J. Walrand, Average bandwidth and delay for reliable multicast, In E. Gelenbe, Ed., *System Performance Evaluation - Methodologies and Applications*, chapter 13, 191–202, CRC Press, 2000.

[35] B. Hajek and S. Yang, Strategic buyers in a sum bid game for flat networks, 2004, preprint.

[36] J. Hicks, The foundations of welfare economics, *Economic Journal*, **69**:696–712, 1939.

[37] H. Hotelling, Edgeworth's taxation paradox and the nature of supply and demand functions, *Journal of Political Economy*, **40**:577–616, 1932.

[38] H. Hotelling, The general welfare in relation to problems of taxation and of railway and utility rates, *Econometrica*, **6** (3):242–269, 1938.

[39] L. Hurwicz, Optimality and informational efficiency in resource allocation processes, In K. Arrow, S. Karlin and P. Suppes, Eds., *Mathematical Methods in the Social Sciences*, Stanford University Press, 1960.

[40] L. Hurwicz, On informationally decentralized systems, In B. McGuire and R. Radner, Eds., *Decision and Organization*, Volume in Honor of Jacob Marschak, 297 – 336, North Holland, 1972.

[41] L. Hurwicz, The design of mechanisms for resource allocation, *American Economic Review*, **63** (2):1–30, 1973.

[42] L. Hurwicz, On the dimensional requirements of informationally decentralized Pareto satisfactory processes, *Studies in Resource Allocation Processes*, 1977.

[43] L. Hurwicz, On informational decentralization and efficiency in resource allocation mechanisms, In S. Reiter, Ed., *MAA Studies in Mathematical Economics*, **25**:238–350, Mathematical Association of America, 1986.

[44] L. Hurwicz, S. Reiter and D. Saari, On constructing mechanisms with message spaces of minimal dimension for smooth performance functions, mimeo, Northwestern University, 1985.

[45] M. Jackson, Bayesian implementation, *Econometrica*, **59**:461–478, 1991.

[46] M. Jackson, Implementation of undominated strategies, *Review of Economic Studies*, **59**:757–775, 1992.

[47] M. Jackson, A crash course in implementation theory, *Social Choice and Welfare*, **18**:655–708, 2001.

[48] M. Jackson, T. Palfrey and S. Srivastava, Undominated Nash implementation in bounded mechanisms, *Games and Economic Behavior*, **6**:474–501, 1994.

[49] H. Jiang and S. Jordan, The role of price in the connection establishment process, *European Transactions on Telecommunications*, **6** (4):421–429, July-Aug. 1995.

[50] R. Johari, S. Mannor, and J. Tsitsiklis, Efficiency loss in a network resource allocation game: the case of elastic supply, 2004, preprint.

[51] R. Johari and J. Tsitsiklis, Efficiency loss in a network resource allocation game, *Mathematics of Operations Research*, **29** (3):407–435, 2003.

[52] R. Johari and J. Tsitsiklis, Efficiency loss in a Cournot mechanism for network resource allocation, 2004, preprint.

[53] J. Jordan, The informational requirements of local stability in decentralized allocation mechanisms, In T. Groves, R. Radner and S. Reiter, Eds., *Information, incentives and economic mechanisms: essays in honor of Leonid Hurwicz*, 183–212, University of Minnesota Press, 1987.

[54] S. Jordan and H. Jiang, Connection establishment in high speed networks, *IEEE Selected Areas in Communications*, **13** (7):1150–1161, 1995.

[55] N. Kaldor, Welfare propositions in economics and inter-personal comparisons of utility, *Economic Journal*, **49**:549–552, 1939.

[56] H. Kanemitsu, Informational efficiency and decentralization in optimal resource allocation, *The Economic Studies Quarterly*, **16**:22–40, 1966.

[57] K. Kar, S. Sarkar and L. Tassiulas, Optimization based rate control for multirate multicast sessions, In *Proceedings of INFOCOM*, Alaska, 2001.

[58] K. Kar, S. Sarkar and L. Tassiulas, A simple rate control algorithm for maximizing total user utility, In *Proceedings of INFOCOM*, Alaska, 2001.

[59] K. Kar, S. Sarkar and L. Tassiulas, A scalable low overhead rate control algorithm for multirate multicast sessions, *IEEE Journal of Selected areas in Communication*, Special Issue in Network Support for Multicast Communications, **20** (8):1541–1557, October 2002.

[60] S. Kasera, G. Hjalmtysson, D. Towsley and J. Kurose, Scalable reliable multicast using multiple multicast channels, *IEEE/ACM Transactions on Networking*, **8** (3):294–310, 2000.

[61] K. Kautsky, *The Social Revolution*, Charles Kerr & Co., 1903.

[62] F. Kelly, On tariffs, policing and admission control for multiservice networks, *Operations Research Letters*, **15**:1–9, 1994.

[63] F. Kelly, Charging and rate control for elastic traffic, *European Transactions on Telecommunication*, **8** (1):33–37, 1997.

[64] F. Kelly, A. Maulloo, and D. Tan, Rate control for communication networks: shadow prices, proportional fairness and stability, *Operational Research Society*, **49**:237–252, 1998.

[65] T. Koopmans, Analysis of production as an efficient combination of activities, In T. Koopmans, Ed., *Activity Analysis of Production and Allocation*, (13):33–37, Cowles Commission Monograph, New York, 1951.

[66] T. Koopmans, *Three Essays on the State of Economic Science*, McGraw-Hill, New York, 1957.

[67] S. Kunniyur and R. Srikant, End to end congestion control schemes: Utility functions, random losses and ECN marks, In *Proceedings of INFOCOM*, Tel Aviv, Israel, 2000.

[68] R. La and V. Anantharam, Charge-sensitive TCP and rate control on the internet, In *Proceedings of INFOCOM*, Tel Aviv, Israel, 2000.

[69] S. Lang, *Linear Algebra*, Springer-Verlag, New York, 1991.

[70] O. Lange, *On the Economic Theory of Socialism*, University of Minnesota Press, Minneapolis, 1938.

[71] O. Lange, The foundations of welfare economics, *Econometrica*, **10**:215–228, 1942.

[72] V. Lenin, *State and Revolution*, International Publishers, 1918.

[73] A. Lerner, The concept of monopoly and the measurement of monopoly power, *The Review of Economic Studies*, **1**:157–175, 1934.

[74] A. Lerner, Economic theory and socialist economy, *The Review of Economic Studies*, **2** (1):51–61, 1934.

[75] S. Low and D. Lapsley, Optimization flow control I: Basic algorithm and convergence, *IEEE/ACM Transactions on Networking*, **7** (6):861–874, 1999.

[76] S. Low and P. Varaiya, A new approach to service provisioning in ATM networks, *IEEE/ACM Transactions on Networking*, **1**:547–553, 1993. see also [77] for corrections.

[77] S. Low and P. Varaiya, Corrections to: A new approach to service provisioning in ATM networks, *IEEE/ACM Transactions on Networking*, **2**:312, 1994.

[78] J. MacKie-Mason and H. Varian, Pricing congestible network resources, *Journal of Selected Areas in Communications*, **13** (7):1141–1149, 1995.

[79] J. MacKie-Mason and H. Varian, Some FAQs about usage-based pricing, *Computer Networks and ISDN Systems*, **28**:257–265, 1995.

[80] J. Marschak and E. Lederer, Der neue mittelstand, *Grundriss der Nationalokonomik*, 1926.

[81] T. Marschak and S. Reichelstein, Network mechanisms, informational efficiency, and hierarchies, *Journal of Economic Theory*, **78**:106–141, 1998.

[82] A. Mas-Colell, M. Whinston, and J. Green, *Microeconomic Theory*, Oxford University Press, New York, 1995.

[83] E. Maskin, The theory of implementation in Nash equilibrium: A survey, In L. Hurwicz, D. Schmeidler, and H. Sonnenschein, Eds., *Social Goals and Social Organization*, Chapter 6, 173–204, Cambridge University Press, 1985.

[84] E. Maskin, Nash implementation and welfare optimality, *Review of Economic Studies*, **66**:23–38, 1999.

[85] E. Maskin and T. Sjőstrőm, Implementation theory, In K. Arrow, A. Sen and K. Suzumura, Eds., *Handbook of Social Choice and Welfare*, **1**, chapter 5, North Holland, 2002.

[86] L. Massoulie and J. Roberts, Bandwidth sharing: Objectives and algorithms, In *Proceedings of INFOCOM*, New York, USA, 1999.

[87] J. Moore and R. Repullo, Nash implementation: A full characterization, *Econometrica*, **58** (5):1083–1099, 1990.

[88] K. Mount and S. Reiter, The informational size of message spaces, *Journal of Economic Theory*, **8**:161–192, 1974.

[89] K. Mount and S. Reiter, Economic environments for which there are pareto satisfactory mechanisms, *Econometrica*, **45** (4):821–842, 1977.

[90] K. Mount and S. Reiter, On the existence of a locally stable dynamic process with a statically minimal message space, In T. Groves, R. Radner and S. Reiter, Eds., *Information, Incentives and Economic Mechanisms: Essays in Honor of Leonid Hurwicz*, 213–242, University of Minnesota Press, Minneapolis, 1987.

[91] L. Murphy and J. Murphy, Bandwidth allocation by pricing in ATM networks, In *Proceedings of the IFIP Broadband Communications*, 333–351, 1994.

[92] L. Murphy, J. Murphy and E. Posner, Distributed pricing for embedded ATM networks, In *Proceedings of the International Teletraffic Congress*, **ITC-14**, 1994.

[93] O. Neurath, *Durch die Kriegswirtschaft zur Naturalwirtschaft*, Callwey, München, 1919.

[94] T. Palfrey, Implementation theory, In R. Aumann and S. Hart, Eds., *Handbook of Game Theory*, **III**:2271–2326, Elsevier Science, 2002.

[95] T. Palfrey and S. Srivastava, Implementation with incomplete information in exchange economies, *Econometrica*, **57**:115–134, 1989.

[96] T. Palfrey and S. Srivastava, Nash implementation using undominated strategies, *Econometrica*, **59**:479–502, 1991.

[97] T. Palfrey and S. Srivastava, Implementation in bayesian equilibrium: The multiple equilibrium problem in mechanism design, Volume I, In J. Laffont, Ed., *Advances in Economic Theory*, in Econometric Society Monographs, (20):283–323, Cambridge University Press, 1992.

[98] T. Palfrey and S. Srivastava, *Bayesian Implementation*, Fundamentals of Pure and Applied Economics, **53**, Harwood academic, 1993.

[99] V. Pareto, La courbe des revenus, *Le Monde economique*, 1896.

[100] V. Pareto, The new theories of economics, *Journal of Political Economics*, **5**:485–502, 1896.

[101] V. Pareto, *Manual of Political Economy*, Augustus Kelly, New York, 1906.

[102] W. Park, H. Owen and E. Zegura, Sonet/SDH multicast routing algorithms in symmetrical three-stage networks, In *IEEE International Conference on Communications*, Seattle, WA, 1995.

[103] C. Parris and D. Ferrari, A resource based pricing policy for real-time channels in a packet-switching network, Technical Report TR-92-018, International Computer Science Institute, Berkeley, CA, 1992.

[104] C. Parris, S. Keshav and D. Ferrari, A framework for the study of pricing in integrated networks, Technical Report TR-92-016, International Computer Science Institute, Berkeley, CA, 1992.

[105] N. Pierson, *Das Wertproblem in der sozialistischen Gesellschaft*, 1902.

[106] A. Postlewaite and D. Schmeideler, Implementation in differential information economies, *Journal of Economic Theory*, **39**:14–33, 1986.

[107] S. Reichelstein and S. Reiter, Game forms with minimal message spaces, *Econometrica*, **56** (3):661–692, 1988.

[108] S. Reiter, Informational efficiency of iterative processes and the size of message spaces, Discussion Paper 11, Center for Mathematical Studies in Economics and Management Science, Northwestern University, 1972.

[109] S. Reiter, The knowledge revealed by an allocation process and the informational size of the message space, Discussion Paper 6, Center for Mathematical Studies in Economics and Management Science, Northwestern University, 1972.

[110] S. Reiter, Information and performance in the (new) welfare economics, *The American Economic Review*, **67** (1):226–234, 1977, Papers and Proceedings of the Eighty-ninth Annual Meeting of the American Economic Association.

[111] D. Rubenstein, J. Kurose and D. Towsley, The impact of multicast layering on network fairness, In *Proceedings of ACM SIGCOMM*, Cambridge, MA, 1999.

[112] T. Saijo, Strategy space reduction in Maskin's theorem: Sufficient conditions for Nash implementation, *Econometrica*, **56**:693–700, 1988.

[113] P. Samuelson, *Foundations of Economic Analysis*, Harvard University Press, Cambridge, MA, 1947.

[114] S. Sanghavi and B. Hajek, Optimal allocation of a divisible good to strategic buyers, 2004, preprint.

[115] S. Sarkar and L. Tassiulas, Fair allocation of resources in multirate multicast trees, In *Proceedings of Globecom*, 1999.

[116] S. Sarkar and L. Tassiulas, Distributed algorithms for computation of fair rates in multirate multicast trees, In *Proceedings of INFOCOM*, Tel Aviv, Israel, 2000.

[117] S. Sarkar and L. Tassiulas, Fair allocation of discrete bandwidth layers in multicast networks, In *Proceedings of INFOCOM*, Tel Aviv, Israel, 2000.

[118] S. Sarkar and L. Tassiulas, Back pressure based multicast scheduling for fair bandwidth allocation, In *Proceedings of INFOCOM*, Alaska, 2001.

[119] S. Sarkar and L. Tassiulas, Fair allocation of utilities in multirate multicast networks: A framework for unifying diverse fairness objective, *IEEE Transactions on Automatic Control*, **47** (6):931–944, 2002.

[120] H. Scarf, *The Computation of Economic Equilibria*, Yale University Press, New Haven and London, 1973.

[121] J. Shapiro, J. Kurose, D. Towsley and S. Zabele, Topology discovery service for router-assisted multicast transport, In *Proceedings of IEEE Open Architectures and Network Programming*, 14–24, 2002.

[122] J. K. Shapiro, D. Towsley and J. Kurose, Optimization-based congestion control for multicast communications, *IEEE Communications Magazine*, **40** (9):90–95, 2002.

[123] T. Sjőstrőm, Implementation in perfect equilibrium, *Social Choice and Welfare*, **10**:97–106, 1993.

[124] V. Smith, Incentive compatible experimental processes for the provision of public goods, In *Research in Experimental Economics*, Eds.: V. Smith, Greenwich, CT: JAI Press, 1979.

[125] T. Stoenescu, Decentralized Resource Allocation in Networks, PhD thesis, University of Michigan, 2004.

[126] T. Stoenescu, M. Liu and D. Teneketzis, A pricing mechanism for optimal rate allocation in multicast service provisioning, *IEEE Transaction in Automatic Control*, 2004, submitted.

[127] T. Stoenescu and D. Teneketzis, Minimal message space Nash implementation of unicast resource allocation problems in networks, in preparation.

[128] T. Stoenescu and D. Teneketzis, Informational efficiency of pricing mechanisms in unicast service provisioning, in preparation.

[129] T. Stoenescu and D. Teneketzis, A pricing methodology for resource allocation and routing in integrated-service networks with quality of service requirements, *Mathematical Methods of Operations Research (MMOR)*, **56** (2), 2002.

[130] F. Taylor, The guidance of production in a socialist state, *The American Economic Review*, **19** (1):1–8, 1929.

[131] P. Thomas and D. Teneketzis, An approach to service provisioning with quality of service requirements in ATM Networks, *Journal of High Speed Networks*, **6** (4):263–291, 1997.

[132] P. Thomas, D. Teneketzis, and J. MacKie-Mason, A market-based approach to optimal resource allocation in integrated-services connection-oriented networks, *Operations Research*, **50** (5):603–616, 2002.

[133] W. Thompson, Concepts of implementation, *Japanese Economic Review*, **47**:133–143, 1996.

[134] H. Tzeng and K. Siu, On max-min fair congestion for multicast ABR service in ATM, *IEEE Journal on Selected Areas in Communication*, **15** (3), 1997.

[135] F. von Hayek, The nature and history of the problem, In F. von Hayek, Ed., *Collectivist Economic Planning: Critical Studies on the Possibilities of Socialism*, 1–40, George Routledge & Sons, London, 1935.

[136] F. von Hayek, The present state of the debate, In F. von Hayek, Ed., *Collectivist Economic Planning: Critical Studies on the Possibilities of Socialism*, 201–243, George Routledge & Sons, London, 1935.

[137] F. von Hayek, Economics and knowledge, *Economica*, **4**:33–54, 1937.

[138] F. von Hayek, The socialist calculation: The competitive solution, *Economica*, **7** (26):125–149, 1940.

[139] F. von Hayek, The use of knowledge in society, *The American Economic Review*, **35** (4):519–530, 1945.

[140] L. von Mises, Die wirtschaftsrechnung im sozialistischen gemeinwesen, *Archiv für Sozialwissenschaft*, **47**:86–121, 1920.

[141] L. Walras, *Éléments d'Économie Politique Pure, ou Théorie de la Richesse Sociale*, Paris: Pichon et Durad-Auzias, Lausanne, 1874.

[142] Q. Wang, J. Peha and M. Sirbu, Optimal pricing for integrated-services networks, In L. W. McKnight and J. P. Bailey, Eds., *Internet Economics*, 353–376, MIT Press, Cambridge, MA, 3rd Edition, 1997.

[143] S. Williams, Necessary and sufficient conditions for existence of a locally stable process, Preprint 88, Institute for Mathematics and Its Applications, University of Minnesota, 1984.

[144] E. Zegura, Routing algorithms in multicast switching topologies, In *Proceedings of the Allerton Conference on Communication, Control and Computing*, Monticello, IL, 1993.

Part IV

Transportation

14

Automated Highway Systems Research: The Influence of Pravin Varaiya

Steven E. Shladover

California PATH Program
Institute of Transportation Studies
University of California, Berkeley
1357 South 46^{th} Street, Building 452
Richmond, CA 94804, USA
steve@path.berkeley.edu

Summary. This chapter provides a review of the research on design of control systems for automated highway applications that has been conducted at the California PATH Program of the University of California, Berkeley. The primary focus is on the research areas that Pravin Varaiya led directly, particularly while serving as PATH Director from 1994 to 1997. He and the researchers working directly with him made significant contributions to the definition of control strategies for coordinating maneuvers of neighboring vehicles and for managing the flows of vehicles along network links, while also developing the hybrid system modeling and simulation tools needed to evaluate the effectiveness of these strategies.

14.1 Introduction

The concept of applying automation technology to the driving of road vehicles has been considered by the transportation community since the late 1930s, when a group of futurists collaborated with General Motors on the development of the “Futurama” exhibit for the 1939–40 New York Worlds Fair. World War II interrupted developments until the late 1940s, when Vladimir Zworykin of RCA Laboratories revived the concept and started developing ideas about how to implement it technically [1]. His work stimulated further activities at General Motors in the 1950s and early 1960s, leading to the development of concept cars, test track prototypes, and publicity films highlighting the safety, comfort and convenience of the “electronic highway” [2]. After the General Motors research focus was reoriented toward nearer-term targets, academic research on highway automation thrived under federal and state sponsorship of the work of Prof. Robert Fenton at Ohio State University from the mid 1960s until about 1980 [3].

The California Department of Transportation (Caltrans) became interested in the potential of highway automation for increasing freeway capacity in the mid-1980s, when it was evident that it would no longer be possible to meet the state’s transportation capacity needs by building additional freeways. In 1986, Caltrans sponsored the

University of California's Institute of Transportation Studies in the creation of a new research program called PATH (initially, Program on Advanced Technology for the Highway, then renamed as Partners for Advanced Transit and Highways in 1992), aimed at developing and testing advanced technologies to reduce traffic congestion [4].

When the new PATH Program began, Pravin Varaiya was one of the first faculty members to become involved. This was an opportunity to develop real-world applications, with direct societal benefits, based on some of his existing research interests in distributed and hybrid control systems. His early involvement provided much of the credibility the new program needed to encourage other Berkeley faculty members, including Charles Desoer, Jean Walrand, Roberto Horowitz, Masayoshi Tomizuka, Karl Hedrick and Shankar Sastry to follow. He also brought a very talented group of students and post-doctoral researchers to work with him on the PATH research projects.

14.2 Hierarchical Architecture for Automated Highway Systems

The vital first step in developing the vibrant PATH research program on Automated Highway Systems (AHS) was Pravin Varaiya's definition of a hierarchical architecture that could accommodate the demanding functional requirements of the system. This architecture [5]–[7] has served as the foundation for nearly 15 years of intensive research at PATH, and has been adopted by many other researchers as well. It has the great virtue of being simple in structure, but so carefully defined that it can be applied to a broader range of transportation applications than AHS. It was essential to begin with an architectural framework like this in order to make it possible to decompose the complexities of AHS into components and subsystems of manageable scope and complexity, so that they could be designed, analyzed and tested by relatively small research teams using desktop computing resources. Varaiya's introductory paper about the architecture and the control problems associated with AHS in the *IEEE Transactions on Automatic Control* [7] has become essential reading for researchers working in this field, and has helped point many researchers toward fruitful research topics.

This architecture, shown in Fig. 14.1, provides a logical decomposition, with the actions involving the fastest updates but the smallest information span and spatial scope toward the bottom, while those involving the broadest spatial scope and information span but the slowest updates are toward the top. The information flows between adjacent layers are clearly defined, making it easy to decouple the layers logically for purposes of design and evaluation, and keeping each layer within a small enough range of time scales and physical scope that it can be modeled without excessive difficulty. Trade-offs involving allocations of functions among individual vehicles, groups of vehicles and the infrastructure can be determined logically within this framework.

The physical layer is where we have the in-vehicle sensing of position relative to the lane and other vehicles, the actuation of steering, engine and brakes, and the

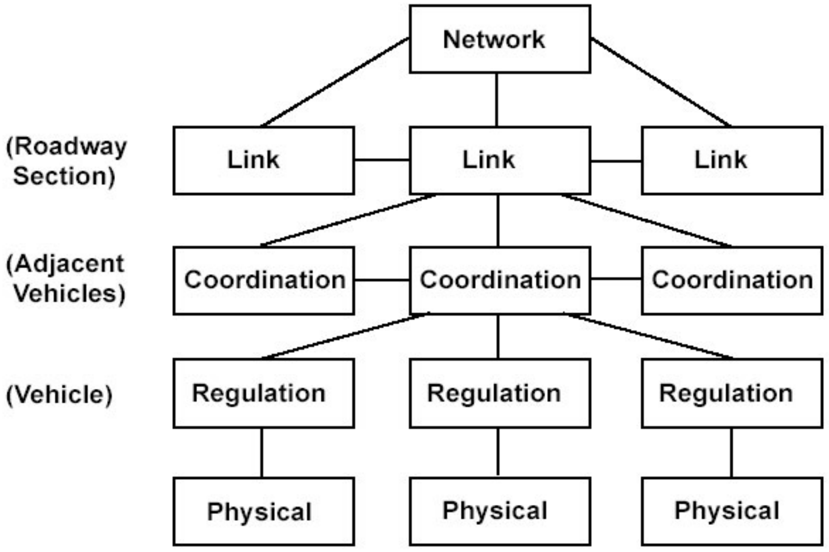


Fig. 14.1. Hierarchical architecture for vehicle automation.

driver-vehicle interface for transitions to and from automatic control. At the regulation layer, we find the classic closing of the vehicle control loops for automatic steering control and the control of vehicle speed and spacing relative to the preceding vehicle(s). The coordination layer is where vehicle trajectories are planned, maneuvers are coordinated among adjacent vehicles (lane changing, joining a platoon or splitting from a platoon), and information is exchanged among groups of vehicles about traffic and road conditions. At the link layer, the emphasis shifts toward aggregate traffic flows rather than individual vehicles or groups of vehicles, with consideration of issues such as managing flows around incidents, balancing traffic across lanes, metering the entry rate of vehicles to maintain good local traffic flow and assigning suitable speed limits to each lane. Finally, at the network layer we have the most aggregate decision making about balancing traffic among alternative routes, rerouting traffic to avoid congested locations, metering access to manage system capacity, and managing serious incidents.

Fig. 14.1 cannot show the full spreading out of the architecture from top layer to bottom layer because of space constraints, but it is important to note that there is a significant spreading from each layer to the next. A large urban region would have a single network-layer operation, but potentially hundreds of links. Each link, with a length of perhaps 1 to 2 km, could have dozens of clusters of closely-coordinated vehicles, each of which could have a dozen or more vehicles.

Since AHS is a safety-critical system, extraordinary measures need to be taken in its design, evaluation, testing and certification in order to provide assurances that people cannot be hurt by it. The hierarchical architecture simplifies this problem by confining the safety-critical elements to the three lowest layers, so that the higher layers can be developed without the extra burden of safety assurance.

In the remainder of this chapter, attention is primarily focused on the coordination and link layers of the architecture and on the modeling and simulation tools for testing system concepts and designs, where Pravin Varaiya's activity has been concentrated and where his influence on other researchers has been most significant.

14.3 AHS Coordination Layer Research

At the coordination layer, the primary issues are the coordination of maneuver plans among vehicles, which have implications not only for capacity but also for the safety of the vehicle operations. The level of coordination may be very low for automation schemes based on use of autonomous vehicles, but it is important to recognize that a certain degree of coordination already exists in conventional driving, implemented through drivers' use of the horn, directional signals and brake lights, but sometimes more subtly through use of gestures, facial expressions and examination of the other driver ("does he see me?"). The design of an automated highway system must ensure that it is no less well coordinated than today's driving.

Pravin Varaiya launched research on coordination layer design by defining a comprehensive set of automated vehicle maneuver protocols for normal driving conditions as 13 state machines, specified in a formal language, "COSPAN," and verified in [8]. This was an extremely important milestone, because it showed that all driving maneuvers could be broken down into a very limited set of simple elemental maneuvers: steady-state cruising in lane, lane changing, joining a platoon (initially, and misleadingly, called "merging"), and splitting from a platoon. In order to simplify the protocol verification, some simplifying constraints were applied, specifically requiring that a platoon could only engage in one maneuver at a time and defining the first vehicle in the platoon (the platoon leader) as the master agent for the platoon as a whole. With these constraints, the full set of protocols, incorporating about 500,000 reachable states and ten million transitions, was verified for completeness and correctness using the COSPAN software verification tool. These protocols became the basis for much subsequent research on both coordination and link layer issues at PATH. The coordination layer was defined as a discrete controller supervising the continuous regulation-layer controllers, together comprising a hybrid system.

The basic architectural framework of [5]–[7] has been extended to address adverse operating conditions and most faults that may occur within the automated highway system by Lygeros et al. [9]. They identified the need to accommodate faults of six different levels of severity, and defined strategies to address those faults, which would be implemented using six new emergency maneuvers to augment the three basic maneuvers that were specified in [8]. The fault classes, in descending order of severity, [with strategies in brackets] are:

- Vehicle stopped or must stop [gentle stop, crash stop, aided stop]
- Vehicle needs assistance to exit [take immediate exit, take immediate exit with escort]

- Vehicle needs no assistance to exit [take immediate exit/normal]
- Vehicle does not need to exit
- Infrastructure failure (not safety critical)
- Driver-vehicle interaction failure

The strategies for handling these faults were defined based on an intuitive understanding of the operation of the automated highway system, trading off system safety and performance, and relying on use of communication for cooperation among vehicles. Significantly, this study did not consider the higher severity class of faults in which a vehicle executes a dangerous maneuver (sudden swerve, accelerate or decelerate), perhaps as a consequence of errors in software design or implementation.

Coordination among vehicles and between vehicles and the roadway infrastructure requires wireless communications, so another important milestone was the definition of the communication protocols needed to implement the maneuver protocols [10]. Even though Pravin Varaiya was not an author of [10], he was certainly acknowledged as an essential influence on this research. The communication protocols were designed to maintain safety while minimizing the impacts of faults on highway operations, but also recognizing that it is not possible to completely preclude injurious crashes when multiple simultaneous faults occur. This chapter included verification to show that the protocols were logically correct and would achieve their desired objectives.

Considerable research has been done on the design and evaluation of the coordinated vehicle maneuvers, within the general protocol framework defined by Pravin Varaiya in [8]. Pravin Varaiya led the initial research on designing the entry and exit maneuvers, in concert with the design of entry and exit ramp geometries [11]. This research showed some of the limitations to the use of a “transition lane” between automated and conventional lanes, pointing toward the need for dedicated entrance and exit ramps for the AHS lanes.

The behavior of the lead vehicle of a platoon needs to be defined at the coordination layer as well as at the regulation layer, since the platoon leader is responsible for the maneuver coordination of the platoon as a whole. Roberto Horowitz and his research team described an initial concept for the platoon leader control in [12], based on definition of an allowable range of speed differences between consecutive platoons (called the “safety region”) in order to avoid high-speed crashes in the event of a hard-braking failure. This study included simulations of two-platoon join and split maneuvers under a variety of conditions, showing how the maneuvers need to take more time in order to reduce crash threats. As an initial study, this work did not address details of vehicle dynamics or sensor data imperfections. Subsequent research by the Horowitz team at PATH [13] led to more detailed definition of both the coordination and regulation layer controllers to avoid any crashes at impact speeds above a specified threshold and to preclude any crashes from propagating from one platoon to the next. This led to the definition of safe inter-platoon spacings and limitations on the maneuvering of the platoon leader, based on consideration of both safety and system performance. Controllers were defined for platoon joining, based on minimizing the time needed for the maneuver, subject to constraints on acceleration, jerk and maximum permissible crash impact speed. Simulations of controller performance showed the join maneuver times to range from 14.7 to 19.4 seconds when joining from initial gaps to 30 m to 60 m respectively, with a brake response delay time of 20 ms.

The coordination layer is where the safety and capacity considerations of vehicle automation intersect, so it is important to develop a strong understanding of the trade-offs between those vital performance measures. The definitive study of these issues until now was conducted by PATH researchers recruited by Pravin Varaiya, under the auspices of the National Automated Highway Systems Consortium [14]. This paper shows a systematic comparison between conventional manual driving, with both alert and typical drivers, and several levels of automated driving (autonomous individual vehicles, two intermediate levels of cooperation, and closely-coupled platoons). These comparisons were based on a "worst case" hazard of a forward vehicle suddenly braking with maximum effort, and then considered the probability and severity of subsequent collisions. The faster responses of the automated systems generally caused them to have much lower collision probabilities than the manual drivers, but with only somewhat milder severity for the cases in which collisions were unavoidable.

The automated platoon systems were significantly different from the autonomous individual vehicles and more loosely coupled cooperative vehicles, because they are designed to make a different trade-off between crash frequency and severity. The very close intra-platoon separations make it impossible to avoid collisions under the severe braking scenario, but they ensure that these collisions occur at very low impact speeds, where injuries are extremely unlikely. Table 14.1 and Fig. 14.2 show the probability density functions of collisions in platoons with nominal vehicle spacings from 1 m to 10 m. The composite measure of collision severity is the expected value of the square of the impact speed, which is proportional to the kinetic energy of the crash. The table shows that the shortest spacings have a higher probability of crash, but much lower severity of crash.

The overall trade-off between highway capacity and safety required an extremely complicated analysis in [14]), which is summarized in Fig. 14.3. In this figure, the crash severity is plotted vertically, with the "best" (least severe) at the top. In this case, the mean-square crash speed is plotted as an expected value to incorporate both the probability and severity of the crash. At the higher values of highway capacity (basically, all values above the current highway capacity of about 2000 vehicles per lane per hour), the platoon cases all show lower crash severities than the individual automated vehicles. This is because in order for the individual vehicles to reach those higher capacity levels they would have had to operate at separations short enough to produce high-speed impacts when crashes could not be avoided. The more frequent crashes of the platooned vehicles were much less severe than the crashes involving the other kinds of automated vehicles, which operated at larger nominal separations. This analysis, despite its complexity and sophistication, was still only able to address the first crash within the platoon, and has not yet been extended to address subsequent crashes that could occur between other pairs of vehicles within the platoon.

Intra-Platoon Spacing (m)	Total Probability of Collision	Expected Collision Severity Surrogate $(\Delta V_{collision})^2$ - (m^2/s^2)
1	0.73	2.94
2	0.62	5.13
3	0.58	7.38
4	0.54	9.87
5	0.51	12.6
6	0.48	15.6
7	0.45	18.9
8	0.42	22.4
9	0.39	26.2
10	0.36	30.2

Table 14.1. Probability and severity of collisions.

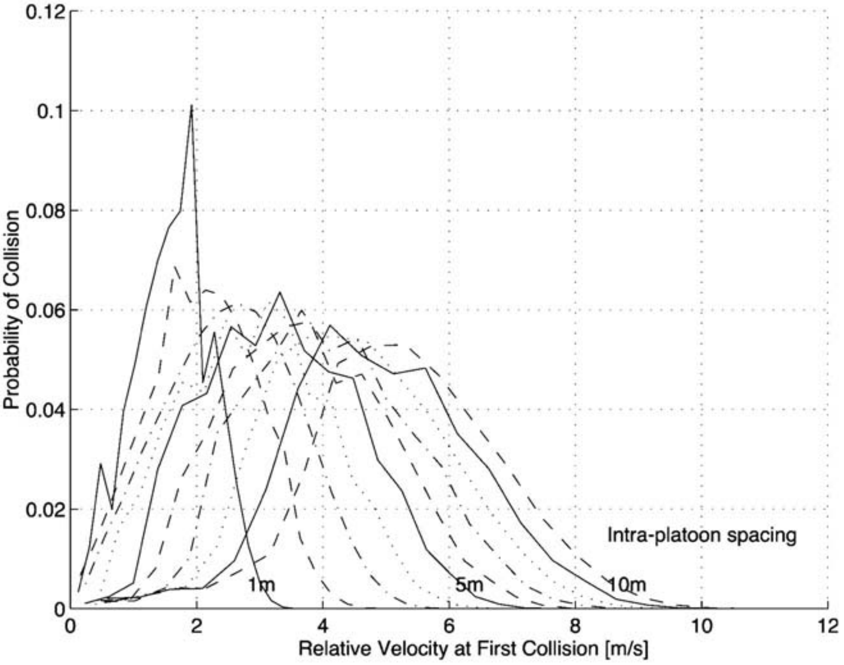


Fig. 14.2. Probability density function of collision severity in sudden hard-braking scenario, as a function of intra-platoon vehicle separation (from Reference [15], also published in Reference [14]).

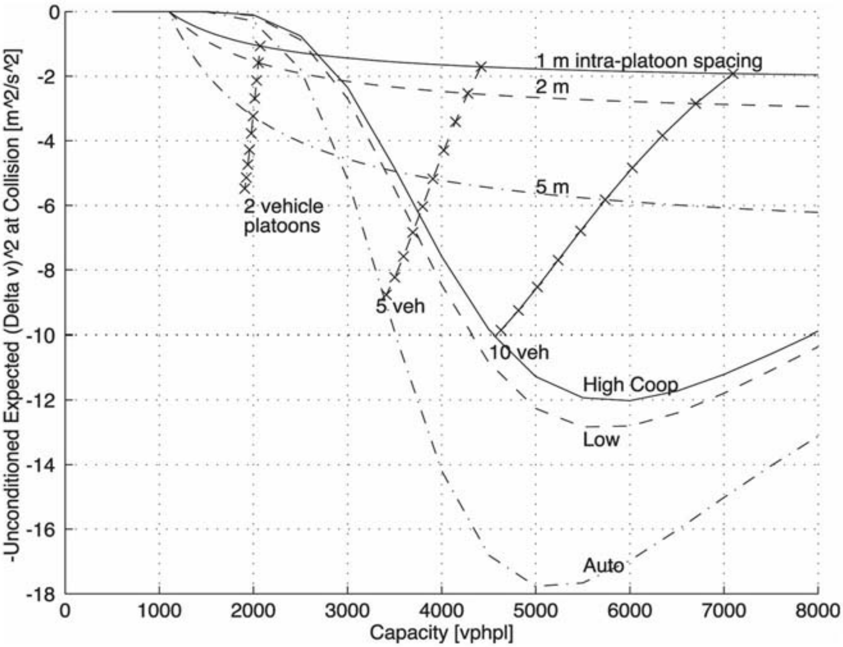


Fig. 14.3. Trade-off between highway capacity and safety for several automated highway operating scenarios (from Reference [15], also published in Reference [14]).

14.4 AHS Link Layer Research

Link layer operations will become important as soon as the first section of automated highway lane(s) needs to be designed. The most basic level of link layer analyses address the capacity increases that could be attained from automation of vehicle operations. Lane capacity analyses date back to the 1960s and 1970s for application to automated guideway transit (AGT) systems. However, much more sophisticated and higher-fidelity studies were performed by PATH researchers for the National Automated Highway Systems Consortium in the mid 1990s, based on consideration of the braking capabilities of production vehicles (automobiles, buses and trucks). Detailed results on the "pipeline" capacity of a simple automated lane, without introduction of merging conflicts, were reported in [15], covering a wide range of assumptions and operating conditions. These showed the effects of different levels of cooperation among the automated vehicles, with the close-coupled platoons providing the highest capacity by a large margin. Introducing even relatively small percentages of buses and trucks to the flow of passenger cars can significantly reduce the achievable capacity because of the poorer performance of these heavier vehicles and their consequent needs for larger safety spacings.

Fig. 14.4 shows a representative example of the results from [15],[16], indicating how the lane capacity depends on speed and the number of vehicles (passenger cars, in this case) in the automated platoon. Comparable results were developed as a function of the size of the intra-platoon gap, as well as other relevant parameters such as the percentages of buses and trucks in the traffic stream, and alternative concepts that would deny access to the automated lanes by vehicles that could not meet certain minimum emergency braking requirements. Nevertheless, this study showed the potential to reach ideal lane capacities in the range of 6000 to 7000 passenger cars per hour before consideration of merge conflicts. More realistic assessments of AHS lane capacity need to include consideration of conflicts with traffic trying to enter at on-ramps, which can further limit capacity, and of the allocation of traffic to lanes in multi-lane AHS applications, particularly in response to incidents that may block one or more lanes. Pravin Varaiya led the first comprehensive treatment of these issues, using a mixture of microscopic and macroscopic simulations [17], which showed successful strategies for assigning target speeds by lane and for recovering from incidents. Almost all subsequent link-layer studies can trace some of their thinking back to this original work.

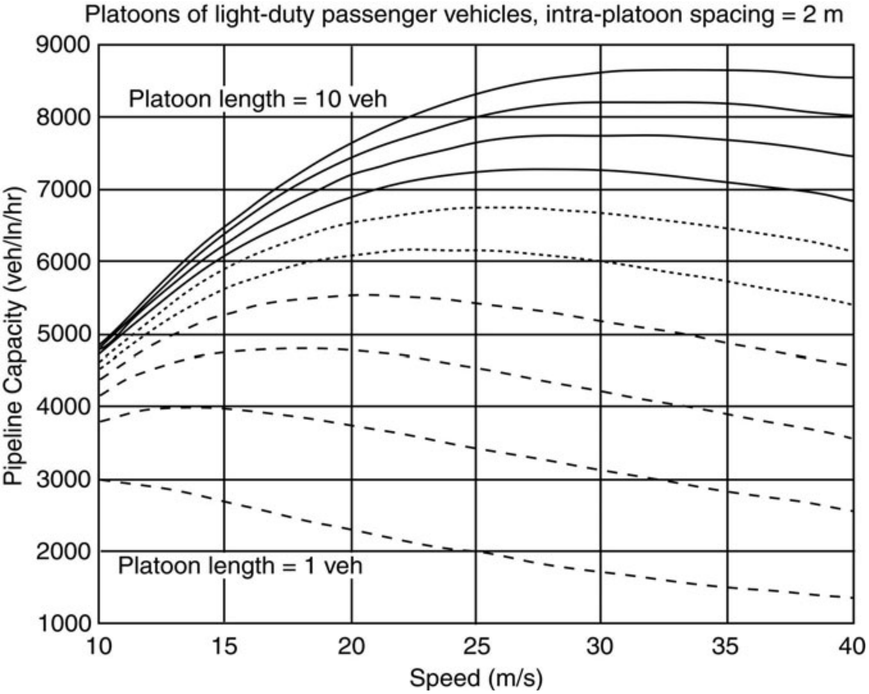


Fig. 14.4. Ideal ("pipeline") automated highway capacity as a function of platoon length and cruising speed (from Reference [15], also published in Reference [16]).

After merging, the other important link layer control function is the assignment of vehicles to lanes in a multi-lane automated highway system. Pravin Varaiya led the development of the basic approach for allocating lane resources to vehicle activities by using the product of space and time needed for each activity [18]. This paper used a linear programming approach to maximize lane capacity, but without considering the effects on travel time. Its case studies addressed the capacity of a single-lane AHS and a merge junction, considering separate examples of manual driving, adaptive cruise control and automated vehicles in 15-car platoons. The results were heavily dependent on the assumptions that were applied about lower-level controller performance and vehicle maneuvering protocols, which were represented in the space estimates for the activities (entry and exit each needing 65 m, platoon join and split maneuvers each needing 28 m, and steady cruising in platoons, ACC and manual driving respectively needing 10 m, 40 m and 50 m). Building from that starting point, a pair of papers by Hall and Li addressed the issue of multiple vehicle classes with different performance characteristics (cars and buses) sharing use of the same AHS facility [19],[20]. Starting from the fundamental safety consideration that vehicles of such different mass and performance should not be combined within the same close-formation platoon, they also considered the possibility of further segregating platoons by destination, but found that that could produce a significant capacity loss [19]. They evaluated a large set of merge entry rules in Monte Carlo simulation, with separate entry queues for the cars and buses, and estimated the queue lengths and times (and queuing ramp lengths needed) for each [20].

Building on the prior Varaiya work, Horowitz' research team addressed lane allocation by seeking to control the density and velocity profiles in each lane of a three-lane AHS [21], using information local to the link communicated among all the local vehicles and the roadway infrastructure.

When an automated highway facility is physically segregated from the normal highway system, it is necessary to provide special access provisions for emergency vehicles to handle crashes, medical emergencies and law enforcement problems. This can be challenging in a physically constrained location when traffic is running close to capacity. Toy et. al [22] defined three types of maneuvers that could be used to help emergency vehicles gain priority access to a problem location, on a highway with at least two lanes but no shoulder, by getting the other vehicles out of the way, and used a flow simulation to show their effectiveness.

An overall link-layer traffic flow control strategy was defined by Alvarez, et.al. [23] as the culmination of the link-layer design work initiated by Pravin Varaiya. This strategy was shown via simulation to be able to stabilize vehicle density and flow rate around the desired profiles by using speed and lane changes as the control signals. This in effect showed that a viable link layer could be designed, complementing other analogous demonstrations of viability of the lower layers in the hierarchical AHS architecture.

14.5 Modeling and Simulation Tools for AHS Design and Evaluation

The design and evaluation of the AHS coordination and link layer concepts and designs could not be conducted entirely in the abstract or using existing analytical tools. From the earliest stages of the PATH Program, we knew that we were going to need new modeling and simulation tools in order to represent AHS systems with sufficient fidelity and efficiency to support multiple iterations of system design. The existing tools were not well suited to support this work for a variety of reasons:

- The existing traffic models are founded on approximations of the behavior of drivers, whose car-following and lane changing behaviors are very different from what the AHS should be doing if it is going to improve highway efficiency, capacity and safety;
- The existing transportation network planning models are still based on higher-level abstractions of driver behavior and traffic flow, and cannot represent the significantly higher capacity that AHS can provide;
- The existing vehicle dynamics and control models were based on continuous dynamic models, but did not address discrete changes of operating mode effectively;
- There were no models available to represent the new phenomena associated with active coordination of vehicle maneuvers at the AHS coordination layer;
- The spatial scope and time scales of the phenomena that need to be represented cover an exceptionally wide range — from an individual vehicle to an entire metropolitan region, and from the millisecond time scale inside the inner control loops onboard a vehicle to a full 24-hour operating day for the transportation network.

Pravin Varaiya recognized AHS as a useful application case study for his research on hybrid systems, because of a variety of complexities inherent in the AHS application:

- vehicles exhibiting continuous, nonlinear dynamic behavior;
- discrete transitions associated with vehicle maneuvers;
- coordination of maneuvers of neighboring vehicles, representing individual agents interacting in complex ways;
- vehicles frequently entering and exiting any defined zone of interest;
- safety-critical consequences of failures necessitating very high confidence in system design;
- realistic system implementations involving large numbers of vehicles and coupled interactions at several levels requiring a computationally efficient simulation in order to be represented effectively.

Varaiya's interests and capabilities in hybrid systems matched well with the needs of the National Automated Highway Systems Consortium for a suite of modeling and simulation tools to support its work. The substantial funding available for development of these tools supported an active cluster of hybrid system research led by Pravin Varaiya, involving a dynamic group of researchers including Akash Deshpande, Aleks Göllü, Farokh Eskafi, Michael Kourjanski, Marco Antoniotti, Luigi Semenzato and Mireille Broucke.

The original framework for object-oriented simulation of hybrid systems, based on distributed control agents working within the previously-defined layered architecture, was called SmartDb, indicating the importance of its underlying database [24]. The first application of this to represent the specific set of AHS maneuvers that were defined in [8] was called SmartPath [25]. SmartPath featured an attractive graphical animation of vehicle maneuvers, but it did not have the flexibility to enable it to represent a broader range of AHS alternatives than those featured in [8]. Nevertheless, the lessons learned in its development were of great value in facilitating the development of the more general hybrid system simulator to follow.

The SmartAHS simulator was developed by the Varaiya-led research team, based on the Object Management Systems (OMS) object-oriented approach [26]. The dynamically reconfigurable hybrid system aspects of the AHS phenomenology motivated the development of a new language for representing more general hybrid systems, initially named the "Hybrid Systems Tool Interface Format"(HSTIF). Since this was unpronounceable, the team decided to scramble the sequence of letters in the name to produce the much easier-to-say name of SHIFT [27]–[28]. Even though it was developed for AHS applications, it has subsequently been used by a variety of researchers to represent a much wider range of dynamic systems, including automated submarines, air traffic control systems and material handling systems.

14.6 Continuing AHS Research Needs

Although great progress has been made in the research directed toward highway automation, there are still some significant research challenges remaining.

14.6.1 Network layer

The network layer of an automated highway system is not dramatically different from a conventional transportation network, except that the availability of automatic control of vehicle motions makes it possible to deterministically assign vehicles to system-optimal routes, rather than depending on the more uncertain choices that are made by human drivers. This could actually make the network-layer control simpler than it is for conventional roadway networks.

The larger challenge at the network layer is in predicting the transportation system impacts of the implementation of a high-capacity automated highway. The uncertainties here are associated with individuals' choices about whether to purchase suitably equipped vehicles and when they choose to use those vehicles on the automated lanes for specific trips. Those decisions in turn affect the long-term patterns of land use and the locational choices that families and businesses make based on their perceptions of the relative accessibility of alternative locations. The extent to which the automated highway ultimately reduces traffic congestion will depend on these (inevitably subjective) decisions about lifestyles and travel behavior.

14.6.2 Link and coordination layers

The research issues at the link and coordination layers are closely related to each other because of the importance of the vehicle maneuvering and communication protocols to both. The first-generation protocols that were described by the Varaiya team in [8] have served as an important foundation for a full generation of research on the operation of these two layers. However, those protocols were built on some extremely conservative assumptions that need to be re-examined in the interest of improving system efficiency. In particular, the following restrictions tend to impede the efficient entry and exit of vehicles to and from an automated lane:

- A platoon may be engaged in only one maneuver at a time.
- A vehicle must complete splits from the other members of a platoon before changing lanes.
- A vehicle must complete its lane change before beginning to join a platoon.
- Split and join maneuvers must transition the full range from the nominal intra-platoon vehicle spacing (in the range of 2 to 4 m) to the nominal inter-platoon spacing (in the range of 60 m), thereby consuming considerable space and time.

These restrictions simplified the verification of the original protocols and helped to ensure their safety, but they also impede the use of promising strategies for cooperative merging and exiting maneuvers (merging vehicle tagging onto the end of a passing platoon or exiting vehicle simply lane changing out of its platoon when a parallel off-ramp becomes available). A second generation of protocols that improves efficiency while maintaining safety needs to be developed, evaluated and verified.

Once the second-generation protocols are defined, many of the link and coordination layer studies will need to be revisited and updated to show how much system efficiency can be improved.

The communication systems to implement coordination among vehicles and between vehicles and the roadway infrastructure also need further development. This is progressing rapidly with the current high interest in “dedicated short-range communications” for safety-critical transportation applications, and is likely to lead to commercially available products within a few years.

14.7 Concluding Remarks

Fifteen years ago, when the PATH research program was starting to ramp up its activities, the state of knowledge on the “system level” issues associated with highway automation was very primitive. Research had already been done on the regulation-layer control of the motions of individual vehicles, including some experiments on test tracks. However, very little thought had been devoted to the higher-level issues associated with how to coordinate the maneuvers of neighboring vehicles to improve their safety and efficiency and how to manage flows of automated vehicles through a roadway network.

The research that Pravin Varaiya led at PATH during the 1990s made major contributions toward showing the technical feasibility of automated highway systems and bringing many other researchers to focus their attention on the related research challenges. The distributed, hierarchical architecture that he defined for AHS (indeed, for all of Intelligent Transportation Systems) remains the foundation for most current thinking on the subject, and has served many researchers well by helping to decompose a large and complicated system of systems into manageable-size pieces. He and his research colleagues have also filled in many of the important gaps in the middle layers of this architecture, showing that:

- all automated vehicle maneuvers can be built up from a common set of simple building blocks;
- these maneuvers can be verified systematically to prove their correctness and safety;
- coordination protocols can be defined to enable vehicles to cooperate with each other efficiently and safely;
- the automated vehicle operations can improve highway lane capacity significantly compared to conventional traffic, without compromising safety;
- link-layer protocols can help automated systems maximize their efficiency even when performance is degraded by vehicle failures or other incidents;
- entry to and exit from high-capacity automated highway lanes should be effected by means of dedicated ramps rather than by merging across adjacent lanes.

Furthermore, his research team has provided the modeling and simulation tools to enable other researchers to explore their own alternative ideas about how to design AHS operations, as well as other complex dynamically-reconfiguring hybrid systems.

Roberto Horowitz and Pravin Varaiya gave their overall perspective on the AHS control system design work that the PATH team did in a paper published in 2000 [29], but even in that overview they introduced some new results on the safety of platoon join maneuvers and link layer control of traffic flow density. Now we have a few more years to look back on those accomplishments and identify what still needs to be done to make progress toward deployment of viable automated transportation systems.

References

[1] V.K. Zworykin and L.E. Flory, Electronic control of motor vehicles on the highways *Highway Research Board Proceedings*, 37^{th} Annual Meeting, 436–451, 1958.

[2] K. Gardels, Automatic Car Controls for Electronic Highways, General Motors Research Laboratories Report GMR-276, 1960.

[3] R.E. Fenton and R.J. Mayhan, Automated highway studies at the Ohio State University—An overview, *IEEE Transactions on Vehicular Technology*, **40** (1):100–113, 1991.

[4] S.E. Shladover, R. Bushey and R. Parsons, California and the Roots of IVHS, *IVHS Review*, **1** (1):27–34, Spring 1993.

[5] P. Varaiya and S.E. Shladover, Sketch of an IVHS Architecture, University of California PATH Program Report, UCB-ITS-PRR-91-3, Institute of Transportation Studies, 1991.

[6] P. Varaiya and S.E. Shladover, Sketch of an IVHS systems architecture, *Vehicle Navigation and Information Systems Conference Proceedings*, Paper No. 912838, 909–928, Dearborn, MI, October 1991.

[7] P. Varaiya, Smart cars on smart roads: problems of control, *IEEE Transactions on Automatic Control*, **38** (2):195–207, Feb. 1993.

[8] A. Hsu, F. Eskafi, S. Sachs and P. Variaya, Protocol design for an automated highway system, *Discrete Event Dynamic Systems*, **2** (1):183–206, 1993.

[9] J. Lygeros, D. N. Godbole and M. Broucke, A Fault tolerant control architecture for automated highway systems, *IEEE Transactions on Control Systems Technology*, **8** (2):205–219, March 2000.

[10] D.N. Godbole, J. Lygeros, E. Singh, A. Deshpande and A.E. Lindsey, Communication protocols for a fault-tolerant automated highway system, *IEEE Transactions on Control Systems Technology*, **8** (5):787–800, Sept. 2000.

[11] D.N. Godbole, F. Eskafi, E. Singh and P. Varaiya, Design of entry and exit maneuvers of IVHS, *Proceedings of the American Control Conference*, 3576–3580, Seattle, WA, 1995.

[12] P.Y. Li, L. Alvarez and R. Horowitz, AHS safe control laws for platoon leaders, *IEEE Transactions on Control Systems Technology*, **5** (6):614–628, November 1997.

[13] L. Alvarez and R. Horowitz, Safe platooning in automated highway systems: Part I: Safety regions design, and Part II: Velocity tracking controller, *Vehicle System Dynamics*, Special Issue on IVHS, **32** (1):23–84, July 1999.

[14] J. Carbaugh, D.N. Godbole and R. Sengupta, Safety and capacity analysis of automated and manual highway systems, *Transportation Research Part C: Emerging Technologies*, **C6**:69–99, 1998.

[15] National Automated Highway Systems Consortium (NAHSC), Milestone 2 Report, Downselect System Configurations and Workshop #3, 1997.

[16] J.B. Michael, D.N. Godbole, J. Lygeros and R. Sengupta, Capacity analysis of traffic flow over a single-lane automated highway system, *ITS Journal*, **4**:49–80, 1998.

[17] B.S.Y. Rao and P. Varaiya, Roadside intelligence for flow control in an intelligent vehicle and highway system, *Transportation Research Part C: Emerging Technologies*, **2C** (1):49–72, March 1994.

[18] M. Broucke and P. Varaiya, Theory of traffic flow in automated highways, *Transportation Research Part C: Emerging Technologies*, **4** (4):181–210, 1996.

[19] R.W. Hall and C. Li, Lane capacity for an automated highway with mixed vehicle classes, *ITS Journal*, **5** (3):217–240.

[20] R.W. Hall and C. Li, Evaluation of priority rules for entrance to automated highways, *ITS Journal*, **6** (2):175–193, 2001.

[21] P.Y. Li, R. Horowitz, L. Alvarez, J. Frankel and A.M. Robertson, An automated highway system link layer controller for traffic flow stabilization, *Transportation Research Part C: Emerging Technologies*, **5C** (1):11–37, February 1997.

[22] C. Toy, K. Leung, L. Alvarez and R. Horowitz, Emergency vehicle maneuvers and control laws for automated highway systems, *IEEE Transactions on Intelligent Transportation Systems*, **3** (2):109–119, June 2002.

[23] L. Alvarez, R. Horowitz and C. Toy, Multi-Destination traffic flow control in automated highway systems, *Transportation Research Part C: Emerging Technologies*, **11C** (1):1–28, Feb. 2003

[24] A. Göllü, A. Deshpande, P. Hingorani and P. Varaiya, SmartDb: An object-oriented simulation framework for intelligent vehicles and highway systems, *Fifth Annual Conference on AI, Simulation and Planning in High-Autonomy Systems*, Gainesville, FL, 1994.

[25] F. Eskafi, D. Khorramabadi and P. Varaiya, An automated highway system simulator, *Transportation Research Part C: Emerging Technologies*, **3C** (1):1–18, Feb. 1995.

[26] A. Deshpande and P. Varaiya, Design and evaluation tools for automated highway systems, *Second European Workshop on Hybrid Systems*, Institut National Polytechnique de Grenoble, 1995.

[27] A. Deshpande and P. Varaiya, The use of SHIFT in system design, *IEEE Conference on Decision and Control*, 708–713, San Diego, CA, 1997.

[28] A. Deshpande, A. Göllü and L. Semenzato, The SHIFT programming language for dynamic networks of hybrid automata, *IEEE Transactions on Automatic Control*, **43** (4):584–587, April 1998.

[29] R. Horowitz and P. Varaiya, Control design of an automated highway system, *Proceedings of the IEEE*, **88** (7):913–925, July 2000.

15

The Traffic Amelioration Potential of Freeway Network Ramp Metering Control

A. Kotsialos and M. Papageorgiou

Dynamic Systems and Simulation Laboratory, Technical University of Crete, 73100 Chania, Greece
appie@dssl.tuc.gr, markos@dssl.tuc.gr

Summary. Recurrent and non-recurrent congestions on freeways may be substantially reduced if today's "spontaneous" infrastructure utilisation is replaced by an orderly, controllable operation via comprehensive application of ramp metering and freeway-to-freeway control, combined with powerful optimal control techniques. This chapter first explains why ramp metering can lead to a dramatic amelioration of traffic conditions on freeways. Subsequently, a large-scale example demonstrates the high potential of advanced ramp metering approaches. It is demonstrated that the proposed control scheme is efficient, fair and real-time feasible.

15.1 Introduction

Urban and interurban freeways had been originally conceived so as to provide virtually unlimited mobility to road users. The on-going dramatic expansion of car-ownership, however, has led to the daily appearance of recurrent and nonrecurrent freeway congestions of thousands of kilometres in length around the world. Ironically, daily recurrent congestions reduce substantially the available infrastructure capacity at the rush hours, i.e. at the time this capacity is most urgently needed, causing delays, increased environmental pollution, and reduced traffic safety. Similar effects are observed in the frequent case of nonrecurrent congestions caused by incidents, road works, etc. It has been recently realized that the mere infrastructure expansion cannot provide a complete solution to these problems due to economic and environmental reasons or, in metropolitan areas, simply due to lack of space.

The traffic situation on today's freeways very much resembles the one in urban road networks prior to the introduction of traffic lights: blocked links, chaotic intersections, reduced safety. It seems like road authorities and road users are still chasing the phantom of unlimited mobility that freeways were originally supposed to provide. What is urgently needed, however, is to restore and maintain the full utilisation of the freeways' capacity along with an orderly and balanced satisfaction of the occurring demand both during rush hours and in case of incidents. Clearly, the passage from chaotic to optimal traffic conditions is only possible if today's "spontaneous" use of the freeway infrastructure is replaced by suitable control

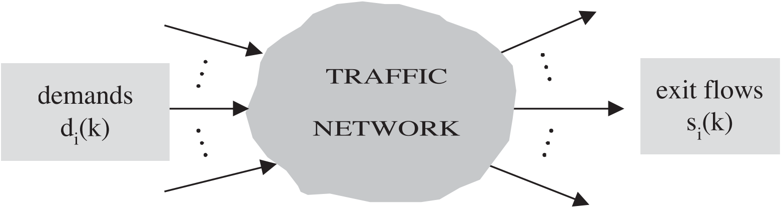


Fig. 15.1. A general traffic network.

actions aiming at the benefit of all users. Ramp metering is the most efficient means to this end, whereby short delays at on-ramps and freeway-to-freeway intersections is the (relatively low) price to pay for capacity flow on the freeway itself, leading to substantial savings for each individual road user.

This chapter first explains, based on simple arguments the reasons why ramp metering may lead to a substantial amelioration of traffic conditions on freeways (section 15.2). Then a hierarchical ramp metering control strategy based on a nonlinear optimal control problem formulation, is presented in section 15.3. A simulation example demonstrates the high amelioration potential of advanced ramp metering algorithms in section 15.4. Finally, section 15.5 summarizes the main conclusions.

15.2 Why Ramp Metering?

15.2.1 A basic property

To be able to answer this question, we will first recall a simple fact. Consider any traffic network (Fig. 15.1) with demand appearing at several locations (e.g. at the on-ramps, in case of a freeway network) and exit flows forming at several destinations (e.g. at the freeway off-ramps). Clearly, the accumulated demand over, say, a day will be equal to the accumulated exit flows, because no vehicle disappears or is generated in the network. Let us assume that the demand level and its spatial and temporal distribution are independent of any control measures taken in the network. Then, we are interested to know how much accumulated time will be needed by all drivers to reach their respective destinations at the network exits (network efficiency!). It is quite evident that this total time spent by all drivers in the traffic network will be longer if, for any reason (e.g. due to lack of suitable control measures), the exit flows are temporarily lower, i.e. if vehicles are delayed within the network on their way to their destinations. As a consequence, any control measure or control strategy that can manage to increase the early exit flows of the network, will lead to a corresponding decrease of the total time spent.

The statements above may be formalized by use of simple mathematics [9], [11]. For the needs of this chapter we will use a discrete-time representation of traffic variables with discrete time index $k = 0, 1, 2, \ldots$ and time interval T. A traffic volume or flow $q(k)$ (in veh/h) is defined as the number of vehicles crossing a corresponding location during the time period $[kT, (k+1)T]$, divided by T.

We consider a traffic network (Fig. 15.1) that receives demands $d_i(k)$ (in veh/h) at its origins $i = 1, 2, \ldots$ and we define the total demand $d(k) = d_1(k)+d_2(k)+\cdots$. We assume that $d(k)$, $k = 0, \ldots, K-1$, is independent of any control measures taken in the network. We define exit flows $s_i(k)$ at the network destinations $i = 1, 2, \ldots,$ and the total exit flow $s(k) = s_1(k)+s_2(k)+\cdots$. We wish to apply control measures so as to minimize the total time spent T_s in the network over a time horizon K, i.e.

$$T_s = T\sum_{k=1}^{K} N(k) \tag{15.1}$$

where $N(k)$ is the total number of vehicles in the network at time k. Due to conservation of vehicles

$$N(k) = N(k-1) + T[d(k-1) - s(k-1)]. \tag{15.2}$$

Substituting (15.2) in (15.1) we obtain

$$T_s = T\sum_{k=1}^{K}\left[N(0) + T\sum_{\kappa=0}^{k-1} d(\kappa) - T\sum_{\kappa=0}^{k-1} s(\kappa)\right]. \tag{15.3}$$

The first two terms in the outer sum of (15.3) are independent of the control measures taken in the network, hence minimization of T_s is equivalent to maximization of the quantity

$$S = T^2\sum_{k=1}^{K}\sum_{\kappa=0}^{k-1} s(\kappa) = T^2\sum_{k=0}^{K-1}(K-k)s(k). \tag{15.4}$$

Thus, minimization of the total time spent in a traffic network is equivalent to maximization of the time-weighted exit flows. In other words, the earlier the vehicles are able to exit the network (by appropriate use of the available control measures) the less time they will have spent in the network.

15.2.2 First answer

We consider (Fig. 15.2) two cases for a freeway on-ramp: (a) without and (b) with metering control. Let q_{in} be the upstream freeway flow, d be the ramp demand, q_{con} be the mainstream outflow in presence of congestion, and q_{cap} be the freeway capacity. It is well-known that the outflow q_{con} in case of congestion is lower by some 5-10% than the freeway capacity q_{cap}. In Fig. 15.2(b), we assume that ramp metering may be used to maintain capacity flow on the mainstream, e.g. by using the control strategy ALINEA [13]. Of course, the application of ramp metering creates a queue at the on-ramp but, because q_{cap} is greater than q_{con} (increased outflow!), ramp metering leads to a reduction of the total time spent (including the ramp waiting

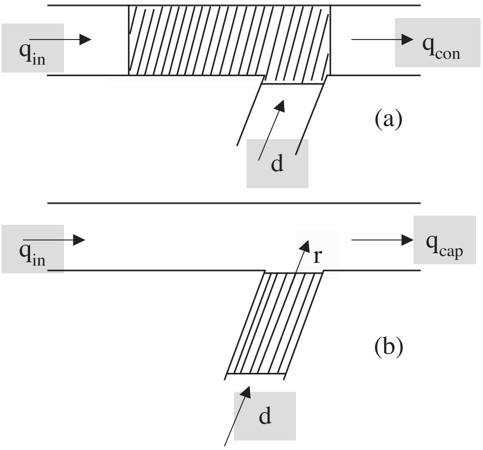


Fig. 15.2. Two cases: (a) without and (b) with ramp metering; grey areas indicate congestion zones.

time). It is easy to show [11] that the amelioration ΔT_s (in %) of the total time spent is given by

$$\Delta T_s = \frac{q_{cap} - q_{con}}{q_{in} + d - q_{con}} 100. \tag{15.5}$$

As an example, if $q_{in} + d = 1.2 q_{cap}$ (i.e. the total demand exceeds the freeway capacity by 20%) and $q_{con} = 0.95 q_{cap}$ (i.e. the capacity drop due to the congestion is 5%) then $\Delta T_s = 20\%$ results from (15.5), which demonstrates the importance of ramp metering.

15.2.3 Second answer

We consider (Fig. 15.3) two cases of a freeway stretch that includes an on-ramp and an off-ramp, namely (a) without and (b) with metering control. In order to clearly separate the different effects of ramp metering, we will assume here that $q_{con} = q_{cap}$, i.e. no capacity drop due to congestion. Defining the exit rate γ $(0 \leq \gamma \leq 1)$ as the portion of the upstream flow that exits at the off-ramp, it is easy to show [11] that the exit flow without control is given by

$$s^{nc} = \frac{\gamma}{1-\gamma}(q_{cap} - d) \tag{15.6}$$

while with metering control we have

$$s^{rm} = \gamma \cdot q_{in}. \tag{15.7}$$

Because $(1-\gamma)q_{in} + d > q_{cap}$ holds (else the congestion would not have been created), it follows that s^{nc} is less than s^{rm}, hence ramp metering increases the

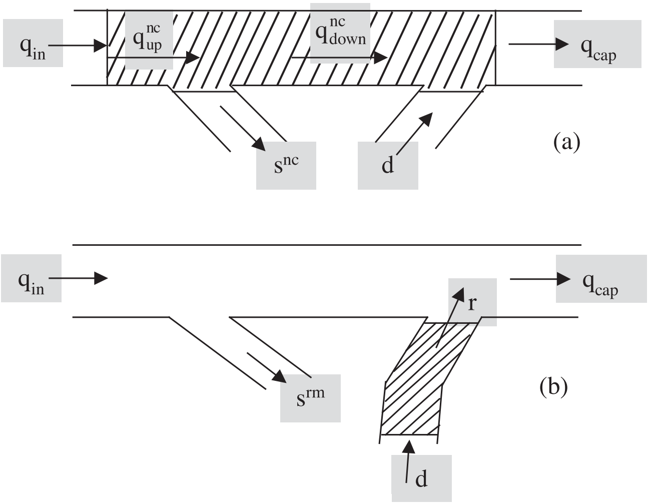


Fig. 15.3. Two cases: (a) without and (b) with ramp metering.

outflow thus decreasing the total time spent in the system. It is easy to show [11] that the amelioration of the total time spent in this case amounts to

$$\Delta T_s = \gamma \cdot 100. \tag{15.8}$$

As an example, if the exit rate is $\gamma = 0.05$ then the amelioration is $\Delta T_s = 5\%$. If several upstream off-ramps are blocked by the congestion in absence of ramp metering (which is typically the case in many freeways during rush hours) then the amelioration achievable via introduction of ramp metering is accordingly higher.

Summing up the effects of sections 15.2.2, 15.2.3 in a freeway network, overall amelioration of total time spent by as much as 50% (i.e. halving of the average journey time) may readily result (see section 4). This may also be demonstrated via suitable treatment of real (congested) freeway traffic data as suggested by Varaiya and coworkers in [1], where it is estimated that the annual congestion delay of 70 million veh·h on Los Angeles freeways could be reduced by 50 million veh·h if the highways were to be operated at 100% efficiency, e.g. via efficient ramp metering.

15.2.4 Further impacts

The road users choose their respective routes towards their destinations so as to minimize their individual travel times. When a control measure (e.g. ramp metering) is introduced that may change the delay experienced in particular network links (e.g. on-ramps), a portion of the drivers will accordingly change their usual route in order to benefit from, or avoid disbenefits due to the new network conditions. For example, in the case of Fig. 15.2(b), the upstream flow q_{in} will probably increase while the ramp demand d will decrease as compared to Fig. 15.2(a). Because the route choice behaviour of drivers is predictable to a large extent (Traffic Assignment problem!), ramp metering may also be used so as to impose an operationally desired traffic flow distribution in the overall network, e.g. avoidance of the rat-running phenomenon, increased or decreased utilisation of underutilised or overloaded, respectively, parallel arterials etc. Clearly, the modified routing behaviour of drivers should be taken into account in the design and evaluation phases of ramp metering control strategies.

Several field evaluation results (see e.g. [7]) demonstrate that ramp metering improves the merging behaviour of traffic flow at freeway intersections which may have a significant positive impact on traffic safety due to fewer lane changes and reduced driver stress. Moreover, the increase of network efficiency related to both answers above, is expected to lead to accordingly improved network traffic safety and reduced pollutant emissions to the environment.

15.3 A Hierarchical Ramp Metering Strategy

15.3.1 Traffic flow modeling

A validated second-order traffic flow model is used for the description of traffic flow on freeway networks. The network is represented by a directed graph whereby the links of the graph represent freeway stretches. Each freeway stretch has uniform characteristics, i.e., no on-/off-ramps and no major changes in geometry. The nodes of the graph are placed at locations where a major change in road geometry occurs, as well as at junctions, on-ramps, and off-ramps.

The time and space arguments are discretized. The discrete time step is denoted by T (typically $T = 5 \dots 15$ s). A freeway link m is divided into N_m segments of equal length L_m (typically $L_m \approx 500$m), such that the stability condition $L_m \geq T \cdot v_{f,m}$ holds, where $v_{f,m}$ is the free-flow speed of link m. This condition ensures that no vehicle traveling with free speed will pass a segment during one simulation time step. Each segment i of link m at time $t = kT$, $k = 0, \dots, K$, where K is the time horizon, is macroscopically characterized via the following variables: the *traffic density* $\rho_{m,i}(k)$ (veh/lane-km) is the number of vehicles in segment i of link m at time $t = kT$ divided by L_m and by the number of lanes Λ_m; the *mean speed* $v_{m,i}(k)$ (km/h) is the mean speed of the vehicles included in segment i of link m at time kT; and the *traffic volume* or *flow* $q_{m,i}(k)$ (veh/h) is the number of vehicles leaving segment i of link m during the time period $[kT, (k+1)T]$, divided by T. The evolution of traffic state in each segment is described by use of two interconnected state equations for the density and mean speed, respectively, [2], [3], [4].

For origin links, i.e., links that receive traffic demand d_o and forward it into the freeway network, a simple queue model is used (Fig. 15.4). The outflow $q_o(k)$ of an origin link is limited by a number of upper bounds; more specifically, $q_o(k)$ cannot be higher than:

(i) the total present ramp demand $d_o(k) + w_o(k)/T$, where $d_o(k)$ is the arriving demand at period k and $w_o(k)$ (veh) is the current ramp queue;

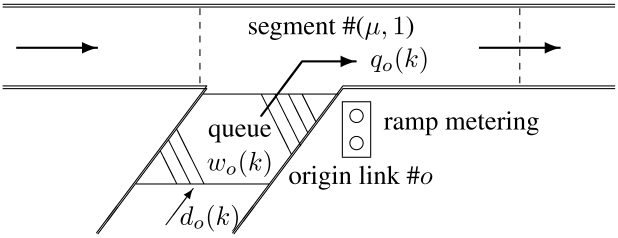


Fig. 15.4. The origin-link queue model.

(ii) the merging flow capacity $Q_o(k)$ which depends on the current density $\rho_{\mu,1}(k)$ of the merge segment; more specifically, $Q_o(k)$ is equal to the constant ramp flow capacity $q_{o,\max}$ so long as $\rho_{\mu,1}(k)$ is less than a critical density $\rho_{\mu,cr}$; if $\rho_{\mu,1}(k) \geq \rho_{\mu,cr}$ then $Q_o(k)$ is linearly decreased with increasing $\rho_{\mu,1}(k)$ and reaches zero when $\rho_{\mu,1}(k)$ attains a maximum value $\rho_{\max}$.

If a ramp is not metered, the corresponding outflow $q_o(k)$ obtains the lowest of both upper bounds above. In case of ramp metering, the outflow $q_o(k)$ is determined by the metering strategy but is eventually limited to the same bounds if necessary. Note that due to bound (ii), a ramp queue may be created even without ramp metering, e.g. if the ramp flow capacity is reduced due to overcritical freeway density $\rho_{\mu,1}$, i.e. due to mainstream congestion. The evolution of the origin queue w_o is described by an additional state equation (conservation of vehicles). Note that the freeway flow $q_{\mu,1}(k)$ in merge segments attains a maximum value $q_{\mu,1}^{\max}$ if the corresponding density $\rho_{\mu,1}(k)$ takes values near a critical density $\rho_{\mu,cr}$.

A similar queue-based approach applies to freeway-to-freeway (ftf) interchanges as well.

Freeway bifurcations and junctions (including on-ramps and off-ramps) are represented by nodes. Traffic enters a node n through a number of input links and is distributed to the output links. The percentage of the total inflow at a bifurcation node n that leaves via the outlink m is the turning rate β_n^m, which can be easily estimated in real time.

15.3.2 Local ramp metering strategies

Most implemented ramp metering systems are based on **local** control strategies that address one single ramp at a time using traffic measurements from the vicinity of the ramp. A most successful local feedback ramp metering algorithm is ALINEA [12], [13] and its recent variations [15], [16]. ALINEA determines a ramp flow $q_o^r(k)$ so as to maintain the traffic conditions in the merge segment $(\mu, 1)$ close to a desired set value by use of an I-type regulator with a control sample time T_c that is a multiple of the model time step T, i.e. $T_c = z_c T$, $z_c \in \mathbb{N}$.

If the set value $\tilde{\rho}_{\mu,1}$ concerns the merge segment density, we have the original ALINEA [12], [13]

$$q_o^r(k_c) = q_o^r(k_c - 1) + K_r\,[\tilde{\rho}_{\mu,1} - \rho_{\mu,1}(k_c)] \tag{15.9}$$

where $k_c = z_c k$ and K_r is the feedback gain factor. If the set value $\tilde{q}_{\mu,1}$ concerns the merge segment outflow, we have the flow-based ALINEA [15]

$$q_o^r(k_c) = q_o^r(k_c - 1) + K_f\,[\tilde{q}_{\mu,1} - q_{\mu,1}(k_c)] \tag{15.10}$$

with K_f the feedback gain factor. Note that the same segment flow $q_{\mu,1}(k)$ may be present for either undercritical or overcritical (congested) density $\rho_{\mu,1}(k)$, hence utilisation of (15.10) is only recommended for set values $\tilde{q}_{\mu,1}$ that are well below the freeway capacity $q_{\mu,1}^{\max}$ [15].

Whichever regulator is used, the resulting flow $q_o^r(k)$ is bounded by the constant ramp flow

capacity $q_o^{\max}$ and a minimum admissible ramp flow $q_o^{\min}$. In order to avoid windup, the term $q_o^r(k_c - 1)$ used in both (15.9) and (15.10) is bounded accordingly.

In order to avoid the creation of large ramp queues that would interfere with the surface street traffic, a queue control policy is employed in conjunction with every local metering strategy. The queue control law takes the form

$$q_o^w(k_c) = -\frac{1}{T_c}\,[w_{o,\max} - w(k_c)] + d_o(k_c - 1) \tag{15.11}$$

where $w_{o,\max}$ is the maximum admissible ramp queue.Thus, the final on-ramp outflow is

$$q_o(k_c) = \max\{q_o^r(k_c), q_o^w(k_c)\} \tag{15.12}$$

which means that queue control may override the ramp metering control whenever necessary to avoid overspilling of the ramp queue.

Typically, ALINEA (15.9) is used with a set value $\tilde{\rho}_{\mu,1} = \rho_{\mu,cr}$ so as to maximise the mainstream flow $q_{\mu,1}$. However, if this is done in several metered ramps independently (no coordination), it may lead to an unbalanced utilisation of the available ramp storage spaces, whereby mainstream congestion may not be avoided due to queue control overrides at some ramps. This is a main motivation for developing coordinated network-wide ramp metering strategies that can exploit the available ramp storage spaces in an optimal way.

15.3.3 Formulation of an optimal control problem for coordinated ramp metering

The traffic flow model described in section 15.3.1 may be extended as follows to include the impact of ramp metering actions. If ramp metering is applied at a ramp o, the outflow $q_o(k)$ is a portion $r_o(k)$ of the flow that would leave the ramp in absence of ramp metering. Thus, $r_o(k) \in [r_{o,\min}, 1]$ is the metering rate for origin link o, i.e. a control variable, where $r_{o,\min}$ is a minimum admissible value while for $r_o(k) = 1$ no ramp metering is applied.

The overall network traffic model has then the general state space form

$$\mathbf{x}(k+1) = \mathbf{f}\left[\mathbf{x}(k), \mathbf{r}(k), \mathbf{d}(k)\right] \tag{15.13}$$

where the state of the traffic flow process is described by the state vector $\mathbf{x} \in \mathbb{R}^N$ and its evolution depends on the system dynamics and the input variables. The input variables are distinguished into control variables $\mathbf{r} \in \mathbb{R}^M$ and uncontrollable external disturbances $\mathbf{d} \in \mathbb{R}^D$. In our case the state vector $\mathbf{x}$ consists of the densities $\rho_{m,i}$ and mean speeds $v_{m,i}$ of every segment i of every link m, and the queues w_o of every origin o. The control vector $\mathbf{r}$ consists of the ramp metering rates r_o of every on-ramp o under control, with $r_{o,\min} \leq r_o(k) \leq 1$. Finally, the disturbance vector consists of the demands d_o at every origin of the network, and the turning rates β_n^m at the network's bifurcations. The disturbance trajectories $\mathbf{d}(k)$ are assumed known over the time horizon K_P. For practical applications, these values may be predicted based on historical data and, if necessary, on real-time estimations, see [17].

The coordinated ramp metering control problem is formulated as a discrete-time dynamic optimal control problem with constrained control variables which can be solved numerically over a given optimization horizon K_P [14]. The chosen cost criterion aims at minimizing the Total Time Spent (TTS) of all vehicles in the network (including the waiting time experienced in the ramp queues). The minimization of TTS is a natural objective for the traffic systems considered here, as it represents the total time spent by all users in the network. Penalty terms are added appropriately to the cost criterion in order for the solution to comply with the maximum queue constraints.

In [6] this nonlinear optimal control problem formulation combined with a powerful numerical optimization algorithm resulted in the AMOC open-loop control tool that is able to consider coordinated ramp metering, route guidance as well as integrated control combining both control measures. In [2], [3], [4] the results of AMOC application to the problem of coordinated ramp metering at the Amsterdam ring-road are presented in detail with special focus on the equity issue. The solution determined by AMOC consists of the optimal ramp metering rate trajectories and the corresponding optimal state trajectories.

Due to various inherent uncertainties the open-loop optimal solution becomes suboptimal when directly applied to the freeway traffic process. In this chapter, the optimal results are cast in a model-predictive frame and are viewed as targets for local feedback regulators which leads to a hierarchical control structure similar to that proposed in [10], albeit with a more sophisticated optimal control approach.

15.3.4 Hierarchical control

The solution provided by AMOC is of an open-loop nature. As a consequence, its direct application may lead to traffic states different than the calculated optimal ones due to errors associated with the system's initial state estimation $\mathbf{x}(0)$, with the prediction of the future disturbances $\mathbf{d}(k)$, $k = 0, \ldots, K_P - 1$, with the model parameters based on which AMOC determines the optimal solution, as well as errors due to unpredictable incidents in the network.

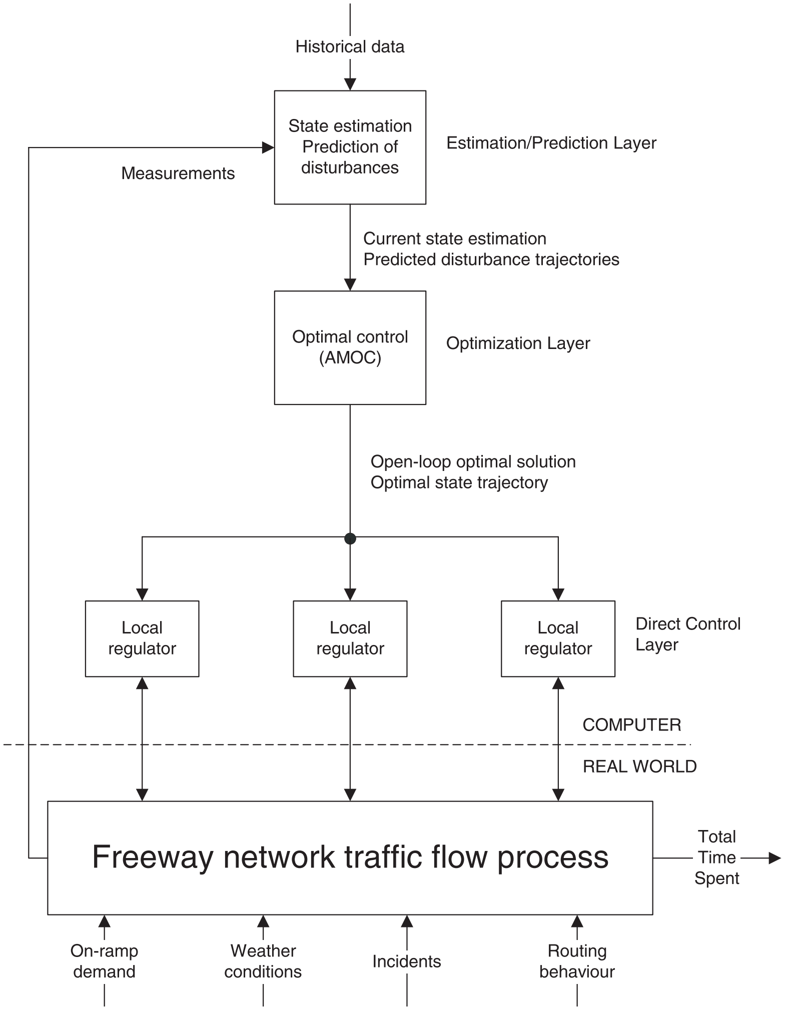


Fig. 15.5. Hierarchical control structure.

Since estimation, modeling and prediction errors are inevitable, a receding horizon approach (model-predictive control) is employed to address any mismatch between the predicted and actual system behavior. This approach is suitably extended to the hierarchical control system depicted in Fig. 15.5, which consists of three layers.

The Estimation/Prediction Layer receives as input historical data, information about incidents and real-time measurements from sensors installed in the freeway network. This information is processed in order to provide the current state estimation and future predictions of the disturbances to the next layer.

The Optimization Layer (AMOC) considers the current time as $t = 0$ and uses the current state estimate as initial condition $\mathbf{x}_0$. Given the predictions $\mathbf{d}(k)$, $k = 0, \ldots, K_P - 1$, the optimal control problem is solved delivering the optimal control trajectory (translated into optimal on-ramp outflows) and the corresponding optimal state trajectory. These trajectories are forwarded as set values to the decentralized Direct Control Layer, that has the task of realizing the suggested policy.

For each on-ramp o with merging segment $(\mu, 1)$ (Fig. 15.4) a local regulator is applied with control sample time $T_c = z_c T$, $z_c \in \mathbb{N}$, in order to calculate the on-ramp outflow $q_o^r(k_c)$, where $k_c = z_c \cdot k$. We define the average quantities $\bar{\rho}^*_{\mu,1}(k_c) = \sum_{z=k}^{k+z_c} \rho^*_{\mu,1}(z)/z_c$ and $\bar{q}^*_{\mu,1}(k_c) = \sum_{z=k}^{k+z_c} q^*_{\mu,1}(z)/z_c$, where the $*$-index denotes optimal values resulting from AMOC.

We distinguish two cases for later comparison. In the first case, the optimal control trajectories are directly applied to the traffic process, i.e.

$$q_o^r(k_c) = \bar{q}^*_{\mu,1}(k_c). \tag{15.14}$$

This is followed by the queue control override (15.11), (15.12).

In the second case, the Direct Control Layer is actually introduced. More specifically, the regulators ALINEA and flow-based ALINEA ([12], [15]) are employed as local regulators, while the optimal state trajectory is used to determine the set-points for each particular on-ramp.

The flows $\bar{q}^*_{\mu,1}$ are preferable as set-points for local regulation because they are directly measurable without the uncertainty caused by modelling. However, flows do not uniquely characterize the traffic state, as the same flow may be encountered under non-congested or congested traffic conditions. Hence a flow set-point $\tilde{q}_{\mu,1} = \bar{q}^*_{\mu,1}(k_c)$ is used (in conjunction with flow-based ALINEA), only if $\bar{\rho}^*_{\mu,1}(k_c) \leq \rho_{\mu,cr}$ and $\bar{q}^*_{\mu,1}(k_c) \leq 0.9 q_{\mu,cap}$, i.e. only if the optimal flows are well below the critical traffic conditions. If $\bar{\rho}^*_{\mu,1}(k_c) \geq \rho_{cr,\mu}$ then ALINEA is applied during the period k_c with set-point $\tilde{\rho}_{\mu,1} = \bar{\rho}^*_{\mu,1}(k_c)$. In any other case, ALINEA is applied with $\tilde{\rho}_{\mu,1} = \rho_{\mu,cr}$ so as to guarantee maximum flow even in presence of various mismatches.

The update period or application horizon of the model-predictive control is $K_A \leq K_P$, after which the optimal control problem is solved again with updated state estimation and the disturbance predictions, thereby closing the control loop of AMOC as in model-predictive control. The control actions will be generally more efficient with increasing K_P and decreasing K_A.

15.4 Simulation Results

15.4.1 The Amsterdam network

For the purposes of our study, the counter-clockwise direction of the A10 freeway, which is about 32 km long, is considered. There are 21 on-ramps on this freeway, including the junctions with the A8, A4, A2, and A1 freeways, and 20 off-ramps, including the connections with A4, A2, A1, and A8. The topological network

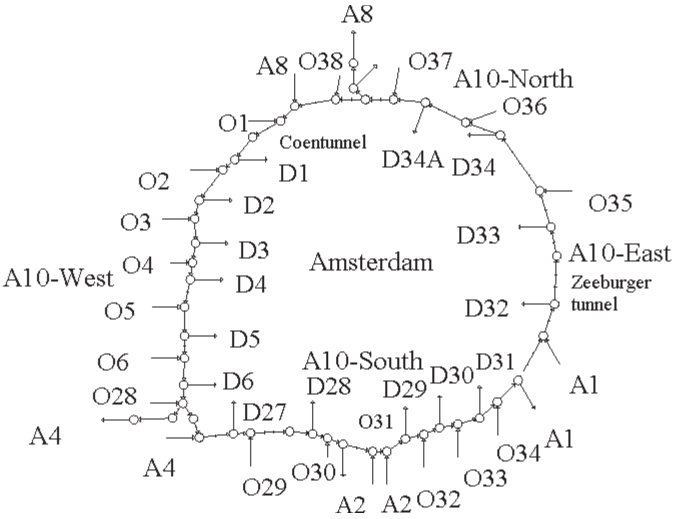


Fig. 15.6. The Amsterdam ring-road model.

model may be seen in Fig. 15.6. It is assumed that ramp metering may be performed at all on-ramps. The model parameters for this network were determined from validation of the network traffic flow model against real data [5].

The ring-road was divided in 76 segments with average length 421m. This means that the state vector is 173-dimensional (including the 21 on-ramp queues). With ramp metering applied to all on-ramps, the control vector is 21-dimensional, while the disturbance vector is 41-dimensional.

The network traffic model described in section 15.3.1 is available as a macroscopic simulator METANET [8] to be used for simulation purposes. This means that the same model is used here for both the optimal control AMOC and the simulator METANET, albeit under some

mismatch conditions detailed later.

15.4.2 The no-control case

Using real (measured) time-dependent demand and turning rate trajectories as input to METANET for the evening peak period 16:00-20:00 p.m. without any control measures, heavy congestion appears in the freeway and large queues are built in the on-ramps. The density evolution profile is displayed in Fig. 15.7 and the corresponding queue evolution profile in Fig. 15.8. The excessive demand coupled with the uncontrolled entrance of drivers into the mainstream causes congestion (Fig. 15.7). This congestion originates at the junction of A1 with A10 and propagates upstream blocking the A4 and a large part of the A10-West. As a result many vehicles are accumulated in the ftf on-ramp of A4 (i.e. we have a spillback of the congestion onto the A4 freeway) and in the surrounding on-ramps (Fig. 15.8). The TTS for this sce-

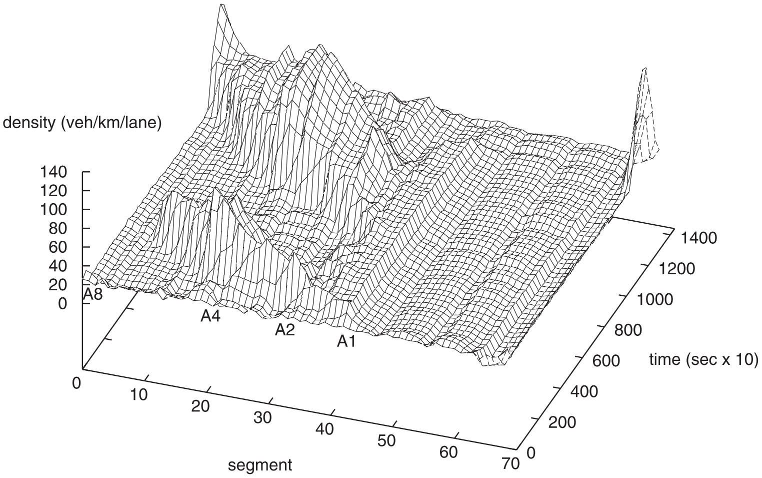


Fig. 15.7. No-control scenario: Density evolution profile.

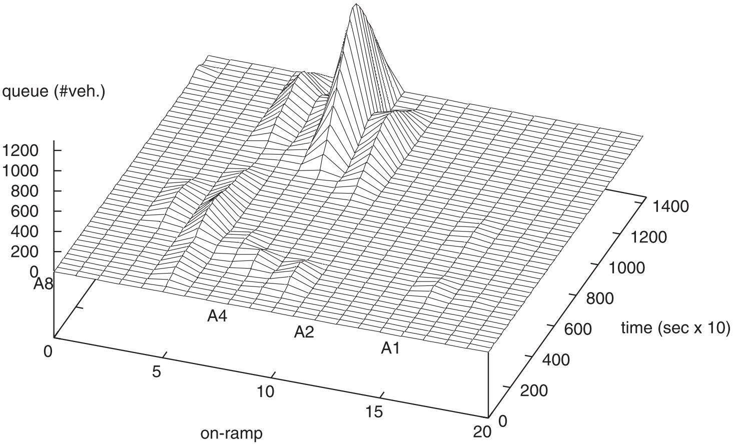


Fig. 15.8. No-control scenario: Ramp queue evolution profile.

nario is equal to 14,167 veh·h. The described simulated traffic conditions correspond pretty accurately to the real uncontrolled traffic conditions in this network during the evening peak period.

15.4.3 Application of ALINEA

In this section the application of the ALINEA strategy (15.9) to all on-ramps is examined. ALINEA is used as a stand-alone strategy for each on-ramp without any kind of coordination. The set-point for each on-ramp *o* is set equal to the critical

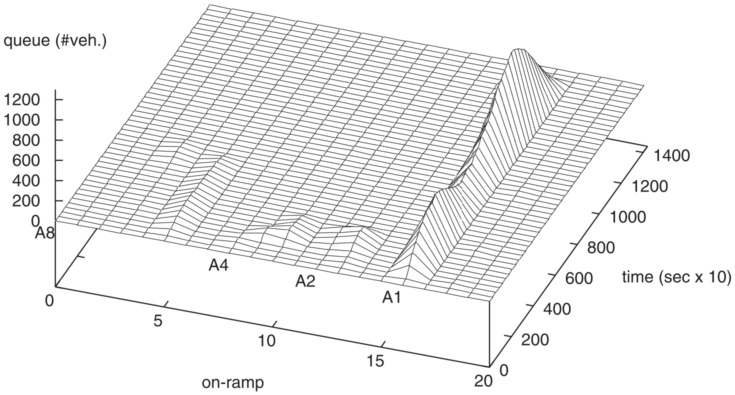


Fig. 15.9. ALINEA control: Ramp queue profile without queue constraints.

density of the corresponding link μ, i.e. $\tilde{\rho}_\mu = \rho_{cr,\mu}$, so as to maximize the local freeway throughput. Two cases are considered with respect to the presence or not of the maximum queue constraint in the sense of (15.12). In case the maximum queue constraints are active, we will assume that the maximum queue length for the urban on-ramps is 100 veh and for the ftf ramps 200 veh. Furthermore, we assume that there is no re-routing of the drivers towards the surrounding urban network when they are confronted with large queues at the on-ramps.

The application of ALINEA without queue constraints leads to a significant amelioration of the traffic conditions and the TTS is reduced to 7,924 veh·h, which is an improvement of 44% compared to the no-control case. The critical point, however, is in the queue evolution profile, where it may be seen (Fig. 15.9) that a huge queue is formed at the A1 ftf ramp, that actually prevents A1's demand from triggering the congestion at the junction of A1 with A10. Clearly, the large A1 ramp queue is not acceptable because it incures excessive delays to the corresponding ramp users (albeit to the strong benefit of the rest of the driver population).

When maximum queue constraints are considered in the sense of (15.12), the application of ALINEA becomes less efficient and the resulting TTS equals 10,478 veh·h, a 26% improvement over the no-control case. The reduction of the strategy's efficiency is due to the fact that the creation of the large queue in the A1 ftf ramp is not allowed any more, hence a congestion is created there, is propagating unpstream and triggers ALINEA action in further upstream ramps (Fig. 15.10).

15.4.4 Application of hierarchical control

First the optimal open-loop solution under the assumption of perfect information with respect to the future disturbances for the entire simulation time is considered. This solution serves as an "upper bound" for the efficiency of the control strategy as it relies on ideal conditions. The TTS in this case becomes 6,974 veh·h, which is a 50.8% improvement over the no-control case.

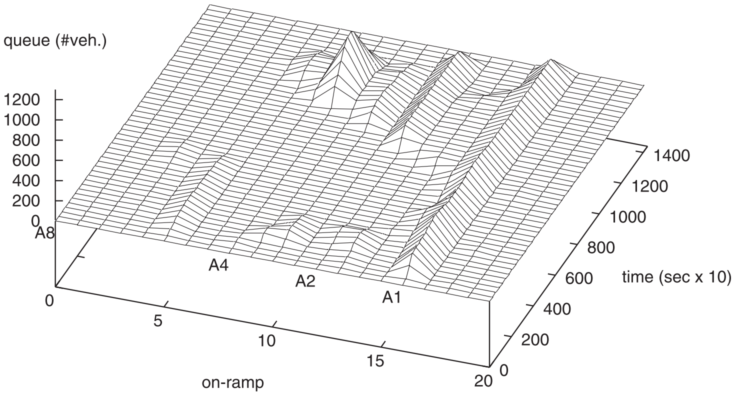


Fig. 15.10. ALINEA control: Ramp queue profile with queue constraints.

As mentioned in section 15.3.4, however, the results obtained by the optimal open-loop control are not realistic because the assumption of perfect knowledge of the future disturbances cannot hold in practice. The hierarchical control proposed is able to cope with this problem by employing the rolling horizon technique. For its application we will use $K_P = 360$ (1 hour) and $K_A = 60$ (10 min). For the purposes of this control scenario, it is assumed that the state of the system is known exactly when AMOC is applied every 10 minutes, which is a fairly realistic assumption.

With respect to the on-ramp demands, we assume that a fairly good predictor is available. Fig. 15.11 depicts an example of the actual and predicted demand, for the ftf on-ramp A8. The actual demand is input to the simulator METANET while the predicted trajectory is input to AMOC. With respect to the prediction of the turning rates, it is possible, based on historical data, to find a mean value for every turning rate for the considered time period. Thus, while METANET considers the real time-dependent turning rates, AMOC uses the average turning rates. Finally, we assume that there is no mismatch between the model parameters used by METANET and the corresponding parameters used by AMOC and that there are no incidents in the network.

As mentioned in section 15.3.4, there are two cases for the application of AMOC results. In the first case, the optimal ramp flows calculated by AMOC are directly applied to the traffic flow process (with ramp queue override when necessary). In the second case, the ALINEA and flow-based ALINEA strategies are employed. In the first case the TTS becomes equal to 8,267 veh·h, which is a 41.6% improvement over the no-control case and 18.5% worsening compared to the optimal open-loop control. When ALINEA is used at the direct control layer, the TTS becomes equal to 8,086 veh·h, which is a 42.9% improvement over the no-control case and 15.9% larger than the TTS of the optimal open-loop control. The density and queue evolution profiles of the second case, are depicted in Figs. 15.12 and 15.13, respectively. The TTS difference between the direct application of AMOC results and the ALINEA employment as a Direct Control Layer is minor in this example; on-going investigations indicate

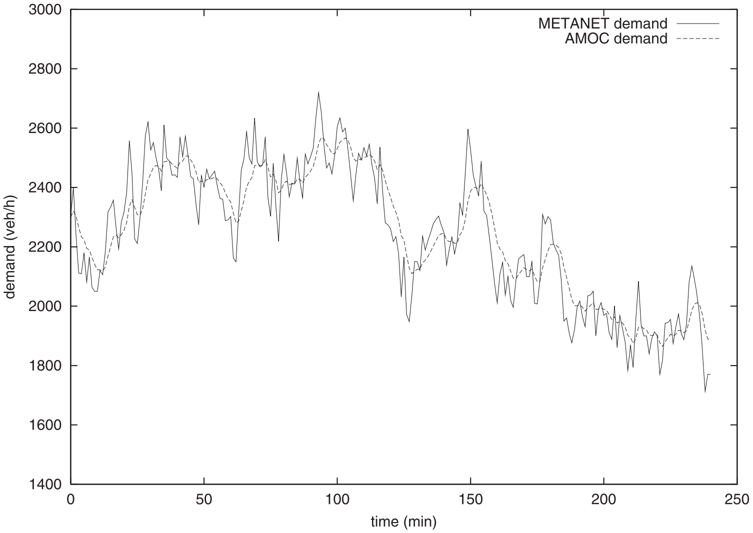


Fig. 15.11. Real and predicted demand at ftf ramp A8.

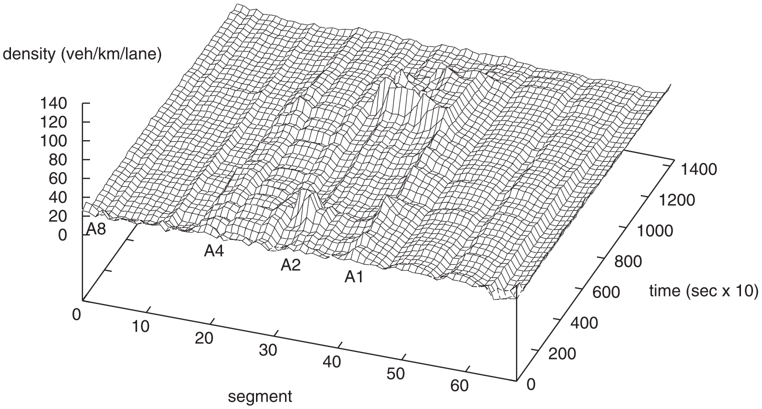


Fig. 15.12. Hierarchical control with ALINEA: Density profile.

large improvements in case of more significant mismatch between AMOC and the METANET simulator.

Comparing the on-ramp queue evolution profile of Fig. 15.13 with the corresponding profile in the case of ALINEA with queue control (Fig. 15.10), the difference between both control strategies becomes apparent. In the ALINEA case, queues are built in the second half of the simulation horizon, in *reaction* to the congestion that has been formed. In the hierarchical control case the queues are built early in the

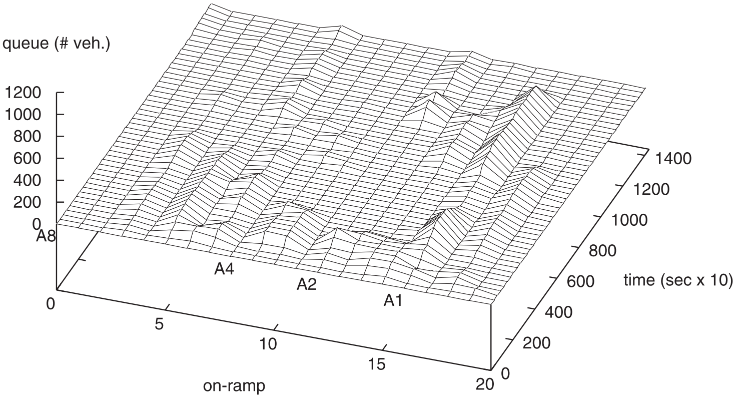


Fig. 15.13. Hierarchical control with ALINEA: Ramp queue profile.

simulation time in *anticipation* of the future congestion. Furthermore, this is done in such a manner that the maximum queue constraints are taken into consideration without serious degradation of the strategy's efficiency.

15.4.5 Equity

The maximum queue constraints may also be used to implicitly address the problem of equity [3]. Fig. 15.14 depicts the average time spent by a vehicle in the ramp queue plus traveling 6.5 km downstream on the freeway, for the no-control case and the three control scenarios considered. It can be seen that in the no-control case the mean travel time is large at the A10-West ramps as a direct consequence of the created congestion. Without queue control ALINEA reduces the mean time for all on-ramps but for A1, where a large peak appears due to the extended delays in the on-ramp queue (Fig. 15.9). The introduction of the queue constraints for ALINEA reduces the mean travel time at A1 but leads to significant travel time increases in other upstream on-ramps of A10-South and A10-West due to mainstream congestion. Clearly this is not a fair distribution of the ramp delays required for the amelioration of the traffic conditions. In the case of the hierarchical control strategy, the travel times for virtually all on-ramps are significantly lower than for no-control or ALINEA with queue constraints. The high peaks in A1 and A2 are not present anymore at the expense of a relatively low increase of the travel times of the on-ramps upstream of A1 compared to the case of ALINEA without queue constraints. The hierarchical controller's distribution of the delays is performed in a more balanced way which is more equitable for the drivers, especially those of A1 and A2. Thus, the proposed hierarchical control leads to a substantial amelioration of the TTS of the whole driver population (efficiency) by improving the travel times of drivers for virtually every individual on-ramp (equity) compared to the no-control case. Since

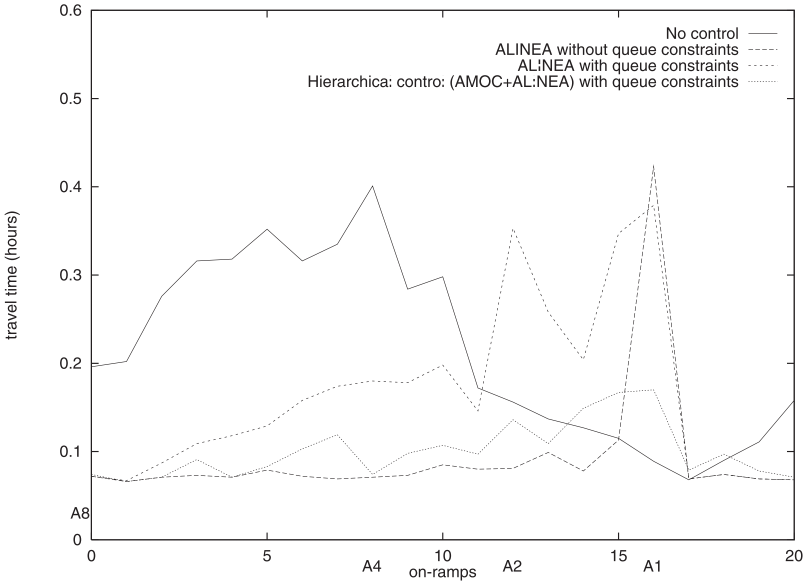


Fig. 15.14. Average travel time for queuing and traveling 6.5 km downstream for every on-ramp.

travel times for all ramps are reduced, the control scheme corresponds to a win-win situation and every driver should be happy with it.

15.4.6 Computation time

In order for the hierarchical control to be applied in the field, the computation time needed for the numerical solution of the associated optimal control problem at each application must be sufficiently low for the real-time application of this approach to be feasible. The required CPU-time varies from application to application, but generally the algorithm converges very fast to an optimal solution within a few CPU-seconds (1MHz P3 processor with Linux), which proves that the real-time application of the control strategy in the field is feasible even for application periods much shorter than the 10 min employed here.

15.5 Conclusions

Modern freeway network capacity is daily underutilized, particularly during rush hours and at the occurrence of incidents, i.e. when it is most urgently needed, due to:

- reduced congestion outflow (see section 15.2.2)
- reduced off-ramp flow (see section 15.2.3)

- uncontrolled flow distribution in the overall network (see section 15.2.4).

The introduction of ramp metering at some particular ramps or particular freeway stretches within the overall network can help to reduce some local traffic problems and to improve the local traffic conditions. However, the significant amelioration of the global traffic conditions in the overall traffic network calls for comprehensive control of all or most of the ramps, including the freeway-to-freeway links, in the aim of optimal utilisation of the available infrastructure. The limitations of partial (rather than comprehensive) ramp metering are:

1. The potential benefits of partial ramp metering (according to Figs. 15.2, 15.3) may be counterbalanced to some extent by a modified route choice behaviour of drivers who attempt the minimisation of their individual travel times under the new conditions.
2. Individual on-ramps have a limited storage capacity for waiting vehicles; if the on-ramp queue reaches back to the surface street junction, ramp metering control is typically released in order to avoid interference with surface street traffic and mainstream congestion cannot be avoided.
3. The freeway network is a common resource for many driver groups with different origins and destinations. Partial ramp metering, by its nature, does not address the strategic problem of optimal utilisation of the overall infrastructure, nor does it guarantee a fair and orderly capacity allocation among the ramps.

Comprehensive ramp metering, on the other hand, does not suffer from these shortcomings, first because of complete control of the network traffic flow and its spatial and temporal distribution, and second because of sufficient available storage capacity. In fact, one or a few particular ramps located at a critical bottleneck area may not have sufficient storage capacity to completely avoid the building up of a congestion. However, in case of comprehensive optimal ramp metering in the sense of section 15.3.3, the total available storage space in all ramps and freeway intersections is usually sufficient to effectively and ultimately combat freeway congestion.

It should be emphasized that the implementation and operation cost of a comprehensive ramp metering system is estimated to be rather low as compared to the corresponding infrastructure cost and to the expected benefits in terms of dramatically reduced delays, increased traffic safety, and reduced environmental pollution. It should also be noted that the advanced methodological tools required for efficient operation of such a comprehensive ramp metering system are currently available, see section 15.3. The major problem to overcome today, is the inertia of political decision-makers which, on its turn, is mainly due to the lack of understanding of the huge potential of comprehensive ramp metering systems.

We believe that freeway networks will have to be operated as completely controllable systems in the near future, similar to the urban traffic networks, because this is the smartest way to avoid further degradation and even fatal gridlocks. The sooner this is realized by the road authorities, the better for the road users who will be the major beneficiaries of this evolution. We would like to acknowledge the valuable research [1] and further manifold actions of Pravin Varaiya towards this end.

Acknowledgments

This work was partly funded by the European Commission in the framework of the project EURAMP (IST-2002-23110). The content of this chapter is under the sole responsibility of the authors and in no way represents the views of the European Commission. The authors would like to thank Mr. F. Middelham from AVV-Rijkswaterstaat, The Netherlands, for providing the necessary data for the Amsterdam ring-road.

References

[1] C. Chen, Z. Jia, and P. Varaiya, Causes and cures of highway congestion, *IEEE Control Systems Magazine*, **21**:26–33, December, 2001.

[2] A. Kotsialos and M. Papageorgiou, Optimal coordinated ramp metering with Advanced Motorway Optimal Control, *Transportation Research Record*, **1748**:55–71, 2001.

[3] A. Kotsialos and M. Papageorgiou, Efficiency and equity properties of freeway network-wide ramp metering with AMOC, *Transportation Research C*, **12** (6):401–420, 2004.

[4] A. Kotsialos and M. Papageorgiou, Nonlinear optimal control applied to coordinated ramp metering, *IEEE Trans. on Control Systems Technology*, **12** (6):920–933, 2004.

[5] A. Kotsialos, M. Papageorgiou, C. Diakaki, Y. Pavlis, and F. Middleham, Traffic flow modeling of large-scale motorway networks using the macroscopic modeling tool METANET, *IEEE Transactions on Intelligent Transportation Systems*, **3** (4):282–292, 2002.

[6] A. Kotsialos, M. Papageorgiou, M. Mangeas, and H. Haj-Salem, Coordinated and integrated control of motorway networks via nonlinear optimal control, *Transportation Research C*, **10** (1):65–84, 2002.

[7] T. McLean, C. Brader, S. Hangleiter, M. Tsavachidis, C. Damas, B. Maxwell, and P. Barber, Urban integrated traffic control evaluation results, In *Deliverable D8.3*, Eur. Transport Telematics Project TABASCO, Brussels, Belgium, 1998.

[8] A. Messmer and M. Papageorgiou, METANET: A macroscopic simulation program for motorway networks, *Traffic Engineering and Control*, **31** (8/9):466–470; 549, 1990.

[9] M. Papageorgiou, *Application of Automatic Control Concepts in Traffic Flow Modelling and Control*, Springer-Verlag, NY, USA, 1983.

[10] M. Papageorgiou, Multilayer control system design applied to freeway traffic, *IEEE Transactions on Automatic Control*, **29** (6):482–490, 1984.

[11] M. Papageorgiou, J.M. Blossville, and H. Hadj-Salem, *La fluidification des rocades de l' Ile de France: Un projet d' importance*, Dynamic Systems and Simulation Laboratory, Internal Report 1998-17, Technical University of Crete, Chania, Greece, 1998.

[12] M. Papageorgiou, H. Haj-Salem, and J.M. Blosseville, ALINEA: A local feedback control law for on-ramp metering, *Transportation Research Record*, **1320**:58–64, 1991.

[13] M. Papageorgiou, H. Haj-Salem, and F. Middelham, ALINEA local ramp metering: Summary of field results, *Transportation Research Record*, **1603**:90–98, 1998.

[14] M. Papageorgiou and M. Marinaki, A Feasible Direction Algorithm for the Numerical Solution of Optimal Control Problems, Dynamic Systems and Simulation Laboratory, Technical University of Crete, Internal Report 1995-4, Chania, Greece, 1995.

[15] E. Smaragdis and M. Papageorgiou, A series of new local ramp metering strategies, *Transportation Research Record*, **1856**:74–86, 2003.

[16] E. Smaragdis, M. Papageorgiou, and E. Kosmatopoulos, A flow-maximizing adaptive local ramp metering strategy, *Transportation Research B*, **38**:251–270, 2004.

[17] Y. Wang, M. Papageorgiou, and A. Messmer, RENAISSANCE: A real-time motorway network traffic surveillance tool, In *Preprints of 10th IFAC Symposium on Control in Transportation Systems*, 235–240, Tokyo, Japan, 2003.

16

Transportation System Intelligence: Performance Measurement and Real-Time Traffic Estimation and Prediction in a Day-to-Day Learning Framework

Hani S. Mahmassani and Xuesong Zhou

Dept. of Civil and Environmental Engineering
Maryland Transportation Initiative
University of Maryland
College Park, MD 20742 U.S.A
masmah@umd.edu, xszhou@wam.umd.edu

16.1 Introduction

Intelligent Transportation Systems (ITS) represent a natural convergence of many of the technologies, concepts, and problem domains in which Varaiya has made seminal contributions over his distinguished career to date. The overarching rationale of ITS is that developments in sensing, location, information and communication technologies can be put to effective use in improving the performance of transportation systems and facilities. Inherently, transportation and communication systems bear many similarities: both are complex dynamic spatial systems, organized around hierarchical network structures, built to deliver services that meet critical human needs. Both carry flows that vary dynamically, with varying degrees of predictability, from origins to destinations, and require control architectures and operational rules to avoid conflict and enable flows to reach their destinations in a manner that maximizes efficient utilization of resources. The main differences lie in the underlying physics: in transportation systems, an added source of complexity is that human beings are the primary agents determining the behavior of the individual particles (vehicles) interacting nonlinearly through the network components. Traffic scientists and physicists have long recognized that this interaction produces collective effects that present both predictable patterns as well as sometimes volatile properties, which greatly affect the resulting performance of these systems, and their ability to meet users' needs and expectations for safe and reliable travel [18].

The interplay between the individual behavior of particles and the collective properties of the system has played a central role in the development of the field of traffic science, and its applications to engineering practice. While advances in traffic science over the past half century have resulted in some well understood and reasonably predictable phenomena, many vexing questions remain, especially with regard to the behavior and properties of systems under high levels of demand relative to the available service resources (infrastructure). The resulting congestion is accompanied by a high degree of inefficiency, as the nature of human interactions in dense environments results in considerable service degradation and reduction in service rates (throughput) of the impacted facilities.

At the root of this inefficiency are the cognitive and behavioral limitations of human drivers. Microscopic characteristics at the individual driver level, such as perception time lags, reaction times, and a natural tendency towards over-reaction under stressful situations or perceived risk, result in volatility, congestion, instability, frustrating stop-and-go patterns, capacity loss, and other component and system level macroscopic phenomena. Eliminate or reduce individual human error, and the system will operate more efficiently. Monitor the state of the system at all times, and it would be possible to intervene and apply control actions in real-time to best utilize available resources. These two realizations have motivated the two main development directions for Intelligent Transportation Systems. Varaiya's substantial contributions to the intellectual, theoretical, methodological and, increasingly, professional practice dimensions of ITS development address both of these opportunity targets.

In the first area, Varaiya realized that by eliminating or minimizing the active role of the human being in the driving process, through automated control systems that rely on precise measurements of neighbor vehicle properties as well as accurate representation of the surrounding environment (e.g. critical roadway design features), one would considerably increase efficiency and reliability. Such systems have been called different things at different times, including automatic vehicle control systems (AVCS) as well as simply automated highway systems (AHS). Varaiya's contributions to this problem form the core scientific underpinnings for the operational analysis and system design of AHS. They provided a much needed access ramp for engineers and researchers from various disciplines, such as control theory and traffic science, to address these problems. In a seminal sole-authored paper [32] (Varaiya, 1993), as well as in a collaborative paper with Hedrick and Tomizuka [17], Varaiya contributed field-defining works that articulated the principal control issues of AHS (see also [19]). Varaiya and his students also produced several seminal contributions to traffic flow modeling under AHS operational rules, as well as to the formulation of rules and protocols for insuring safe maneuvers in a mixed traffic environment, i.e. one in which automated vehicles share the right of way with human-controlled vehicles [13], [29], [4].

In the second area, Varaiya acted on the dual realization, one technical and the other professional/institutional, that (1) sensing prevailing conditions in traffic networks plays a central role as a basis for "intelligence" in transportation systems, especially freeways; and (2) existing sensor systems already deployed are only delivering a small fraction of their potential value to the owner agencies because of absent or arcane decision support tools that could enable traffic managers to query the rich database of accumulating traffic information. This gave rise to PeMS, a general Performance Measurement System, initially applied to California Freeways [11],[33], which has become a model showcase of how to effectively leverage the massive amount of sensor data collected on the transportation system. Built around an elegant software design that retains considerable simplicity and ease of use (via simple web browser), yet enables efficient data query and analysis, PeMS is a framework that allows the ITS community to leverage the investment in sensors and sensor data to support a wide range of operational and planning uses by both practitioners and researchers. It also makes all data available online, providing an excellent resource for researchers.

Applications of the PeMS data bases are numerous, and the potential applications cover a range of operations and planning issues; some examples include travel time reliability assessment, systematic identification of freeway bottlenecks, and exploration of the causes and cures of traffic congestion [10], [8], [9]. Additional work on sensors and traffic sensor data interpretation [23] is continuing, and is now targeting the development of relatively low-cost wireless sensor networks for widespread deployment of traffic sensing capabilities.

While the emergence of modern ITS ideas can be traced back to the late 1980's, early ideas dating back to the 1960's had been articulated by control theorists and traffic systems engineers. As is often the case with visionary technology applications, the technologies themselves (e.g. in wireless communications, wireless-assisted GPS location, mapping and geographic information systems) have already advanced well beyond the initial vision, though the deployment and institutional adoption of the overall systems have remained far short of the original designs. What is remarkable about Varaiya's contributions to ITS is that they span the whole realm of underlying enabling technologies, to specific control structures and rules for the application of these technologies, as well as, more recently, decision support platforms for the delivery of mission-critical applications to user agencies that allow them to exploit the potential of their investment in sensing and monitoring infrastructure. With the data comes knowledge and with knowledge comes power—these investments in data analysis tools are already beginning to generate important payoff in terns of knowledge about fundamental properties of the systems under observation.

An important use of the PeMS data is the role it can play in the development, calibration, validation and operational deployment of advanced traffic network analysis tools, such as network assignment-simulation models for real-time estimation and prediction of network states to support development of online routing and control strategies. The rest of the chapter describes such an application conducted by the authors at the University of Maryland, for which PeMS data for Orange County in California played an important role. The chapter focuses specifically on the framework devised to estimate and predict dynamic origin-destination (OD) trip demand patterns, and update these patterns from one period (one day) to the next. The predicted OD demand serves as input to a network simulation capability that incorporates users' responses to supplied traffic information and control actions.

The rest of the chapter is organized as follows. Following the motivation in the next section, a structural state space model for real-time OD estimation and prediction is presented, within a rolling horizon execution framework in connection with real-time dynamic traffic assignment simulators. By considering demand deviations from the a priori estimate of the regular pattern as a time-varying process with smooth trend, a polynomial trend filter is developed as the core model to capture possible structural deviations in real-time demand. In Section 16.4, a Kalman filter formulation and the corresponding optimal updating algorithms are presented to keep track of the up-to-date regular demand pattern using real-time information. Section 16.5 describes the application results of the proposed models and algorithms using real-world PeMS data for the Irvine, Orange County network.

16.2 Motivation and Background

The premise of ITS is the ability to sense prevailing conditions and rapidly devise actions to optimize system performance in real-time. Because the dynamics of traffic systems are complex, as they depend on the interaction of many independent agents (drivers) acting non-cooperatively in a spatially connected network, many situations call for strategies that anticipate unfolding conditions instead of adopting a purely reactive approach. Real-time simulation of the traffic network forms the basis of a state prediction capability that fuses historical data with sensor information, and uses a description of how traffic behaves in networks to predict future conditions, and accordingly develop control measures. Because these actions are predicated on network conditions, which in turn depend on the users' decisions, network states have to be determined simultaneously with the tripmaker choices, generally in an iterative scheme. The estimated state of the network and predicted future states, in terms of flows, travel times and other time-varying performance characteristics on the various components of the network, are used in the on-line generation and real-time evaluation of a wide range of measures, including information supply to users. The core of the descriptive DTA capability is a traffic simulation model, intended to capture the dynamics of traffic flow movement in the network [24], [20] [26],[25].

The two capabilities above (descriptive and normative), along with their support functions, are integrated in the DYNASMART-X DTA System, to provide, in real-time: (1) estimates of network traffic conditions, (2) predictions of network flow patterns over the near and medium terms in response to various contemplated traffic control measures and information dissemination strategies, and (3) routing information to guide trip-makers in their travel. The system includes several functional modules (for OD estimation, OD prediction, real-time network state simulation, consistency checking, updating and resetting functions, and network state prediction), integrated through a flexible distributed design that uses CORBA (Common Object Request Broker Architecture) standards, for real-time operation in a rolling horizon framework with multiple asynchronous horizons for the various modules.

Dynamic origin destination (OD) demand estimation and prediction is an important capability in its own right, and an essential support function for real-time dynamic traffic assignment (DTA) model systems for ITS applications. The dynamic OD demand estimation and prediction problem seeks to estimate time-dependent OD trip demand patterns at the current stage, and predict demand volumes over the near and medium terms in a general network, given historical demand information and real-world traffic measurements from various surveillance devices (e.g. occupancy and volume observations from loop detectors on specific links).

Substantial research efforts have been devoted to dynamic demand estimation and prediction problems over the past 20 years. Existing models can be grouped into two classes: DTA based vs. non-DTA based, depending on whether a DTA component is incorporated into the estimation process [22],[6],[28]. In this chapter, existing models are categorized according to the underlying assumptions in representing dynamic demand processes. Assuming that the deviations of flow (demand) from historical averages define a stationary time series, the first group applies auto-regressive (AR) models to the recursive estimation and prediction process. In the Kalman filtering formulation proposed by Okutani and Stephanedes [27], the original data is first detrended from historical observations, then an AR model is used to estimate and forecast time-varying traffic flows on a single link. Along the same line, Ashok and Ben-Akiva [2],[3] formulated deviations of OD demand from historical averages as AR processes, and further developed a Kalman filter for real-time OD demand estimation and prediction, in which a 4^{th}-order AR model is adopted based on several data sets. In general, an autoregressive model is suitable to describe a stationary random process with constant mean and variance. On the other hand, if the prevailing OD demand is structurally different from the regular demand pattern, demand deviations will not satisfy the fundamental stationarity assumption for AR processes, and such non-stationarity could seriously degrade the overall prediction performance. In addition, an AR type model with high-order terms requires extensive off-line calibration effort for the autocorrelation coefficients, and the corresponding augmented state space also dramatically increases the on-line computational burden, especially for large-scale network applications.

Alternatively, without requiring prior demand information, a simple random walk model can be relatively easily built for short-term demand prediction, corresponding to an AR(1) model with autocorrelation coefficient of 1. Cremer and Keller [14],[15], as well as Chang and Wu [7] applied the random walk model to predict dynamic OD flow split parameters, by directly extending the latest estimates as the future forecasts. Although this model is effective for a slowly changing process, it might not be rich enough to capture non-linear trends in time-varying OD flows, especially for medium term prediction. In order to describe the non-linearity in dynamic OD demand, Kang [22] and Mahmassani et al. [26] proposed a polynomial trend filter to estimate time-dependent OD flows on a general network, using historical information to calibrate demand evolution processes.

In a closely related problem area, approaches for off-line time-varying OD demand estimation have also been proposed in the past decade, mostly for operational planning applications. Using a simplified assignment model, Cascetta et al. [5] presented a generalized least squares framework for estimating time-varying demand in a network. A bi-level DTA-based time-varying demand estimation formulation was introduced by Tavana and Mahmassani [31] and further extended by Zhou et al. [35] to utilize multi-day link counts. In contrast, little attention has been given to procedures for effectively and systematically updating the historical demand information for on-line estimation and prediction purposes. Ashok [1] suggested several heuristic approaches to update the historical demand estimate with recent estimates obtained in real-time, but no optimal updating formulation was given.

In general, regular OD trip desires can be viewed as a repeated process with similar within-day dynamic patterns. By utilizing knowledge from household interview surveys and off-line estimation results on multiple days, historical demand data represents the *a priori* estimate of the regular OD demand pattern. In particular, in the context of long-range demand prediction, reliable historical data can serve as an informative source under normal conditions. On the other hand, it is necessary to recognize the possible existence of structural deviations of real-time OD demand from the average pattern; these might be caused by severe weather conditions, special events, as well as the responses of travelers to information and/or other system management measures. The first two factors have been well recognized as critical determinants in the effectiveness of travel demand management systems. With increasing availability, traveler information, particularly, pre-trip information, is expected to play a more active role in gradually changing day-to-day trip-making decisions and the resulting temporal distributions of OD demand. In addition, random fluctuations would still account for the effect of other unobserved factors and the inherent stochastic nature of daily time-varying demand.

In the early deployment of real-time OD estimation and prediction, a common issue is that only unreliable historical demand data with significant uncertainty is available, often consisting of out-of-date survey data and limited surveillance data. In this case, as the prior estimate cannot adequately describe the average conditions, the real-time estimate becomes more informative in the sense that it captures the prevailing demand pattern and encapsulates up-to-date demand information.

To provide accurate and robust demand estimation and prediction for real-time dynamic traffic assignment in operational settings, the following primary functional requirements need to be satisfied: (1) incorporate regular demand information into the real-time demand prediction process; (2) recognize and capture possible structural changes in demand patterns under various conditions; and (3) optimally update the *a priori* estimate of the regular pattern using new real-time estimation results and traffic observations.

In this work, actual dynamic OD demand is decomposed to three meaningful components in a structural state space model, namely,

true demand = regular pattern + structural deviations + random fluctuations.

The next section first describes a rolling horizon execution framework for real-time OD estimation and prediction in connection with real-time DTA simulators, followed by the introduction of a structural state space model for real-time OD estimation and prediction. By considering demand deviations from the *a priori* estimate of the regular pattern as a time-varying process with smooth trend, a polynomial trend filter is developed as the core model to capture possible structural deviations in real-time demand.

16.3 Structural Model for Real-Time OD Estimation and Prediction

The rolling horizon framework in this chapter follows the system design of a real-time dynamic traffic assignment system [25],[31]. The scheme entails sequential execution of the OD estimator and predictor, in conjunction with real-time DTA simulators. As shown in Figure 16.1, the prediction (or planning) horizon represents the time length for which forecasted OD demand should be available for the DTA simulator. The prediction horizon starts at the end of a roll period, which is the time shift between the respective beginning of consecutive prediction horizons. Predictions for a given period are based on the estimation results obtained during the roll period, using observations streaming in real-time over a certain observation period.

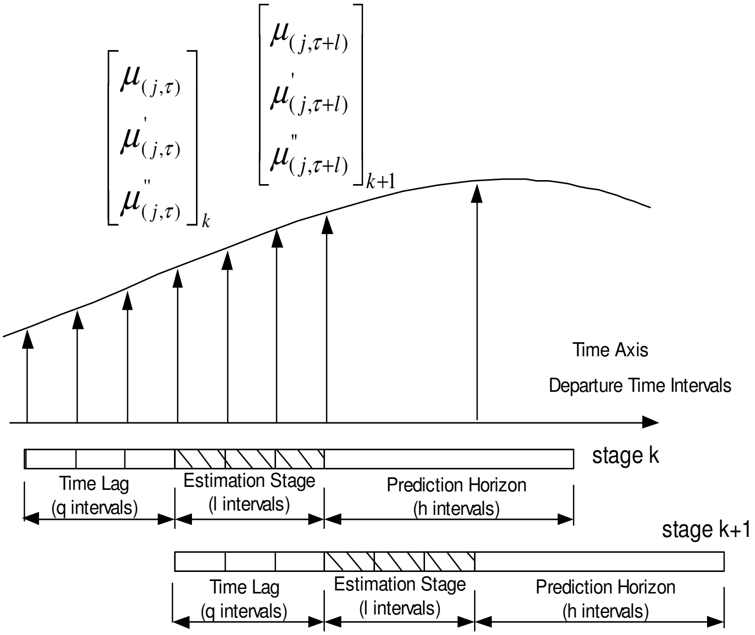

$\begin{bmatrix}\mu_{(j,\tau)}\\ \mu'_{(j,\tau)}\\ \mu''_{(j,\tau)}\end{bmatrix}_k$
$\begin{bmatrix}\mu_{(j,\tau+l)}\\ \mu'_{(j,\tau+l)}\\ \mu''_{(j,\tau+l)}\end{bmatrix}_{k+1}$
Time Axis
Departure Time Intervals
stage k
Time Lag (q intervals)
Estimation Stage (l intervals)
Prediction Horizon (h intervals)
stage k+1
Time Lag (q intervals)
Estimation Stage (l intervals)
Prediction Horizon (h intervals)

Fig. 16.1. Illustration of rolling horizon implementation

In the approach above, a thorny modeling issue in OD estimation is how to handle lagged OD demand on current link observations. This issue arises because each traveler takes a certain time to complete his/her trip in a large city network, and the resulting travel time can be very long depending on trip length and prevailing traffic conditions. Failure to recognize the existence of lagged demand would attribute all current flows to demands departing during the current estimation stage, potentially leading to serious bias in estimation results. One possible solution is to extend the dimension of the state variable vector so as to include all the lagged OD demand variables in the current estimation stage [27], but the resulting expanded state space could significantly increase the computational complexity. The proposed polynomial trend model offers a compact representation of lagged demands, as described in a later section.

The rolling horizon implementation of real-time OD estimation and prediction is stated as Algorithm 1. This approach integrates real-time OD estimation and prediction with other on-line DTA components; specifically, the DTA simulator is relied upon to generate link proportions for the OD estimation module at the current stage, and OD prediction provides future OD demands for the assignment and simulation in the next stage.

16.3.1 Algorithm 1. Rolling horizon implementation for real-time OD estimation and prediction

Step 1: Receive real-time traffic measurements from surveillance system.
Step 2: Fetch link proportion data for the current estimation stage from the DTA simulator.
Step 3: (OD estimation) Estimate time-varying OD demand matrices involved in the current estimation stage using the Kalman filtering method.
Step 4: (OD prediction) Predict OD demand over next future horizon.
Step 5: Advance roll period forward, and then go back to Step 1.

For convenient reference, the notation used in the real-time OD estimation and prediction model is first presented, as follows:

i = index for links with traffic measurements, $i = 1, \ldots, N_{obs}$.

j = index for origin-destination pairs, $j = 1, \ldots, N_{od}$

τ = index for aggregated departure time intervals, $\tau = 1, 2, \ldots$

t = index for observation time interval, i.e. sampling time interval, $t = 1, 2, \ldots$

k = index for stage period, $k = 1, 2, 3, \ldots$

n = number of observation intervals per departure time interval

l = number of departure time intervals per roll period

h = prediction horizon in numbers of departure time intervals

q = maximum lag length in numbers of departure time intervals, i.e. the traffic flow at the current departure time interval τ can include traffic demand departing from interval τ, $\tau - 1$, $\tau - 2$, $\ldots$, $\tau - q$

$c_{(i,t)}$ = number of vehicles measured on link i, during observation interval t

$D_{(j,\tau)}$ = demand volume from origin-destination pair j during departure time interval τ

$LP_{(i,t),(j,\tau)}$ = link proportions, that is the proportion of vehicles on link i at observation time t (coming from OD pair j at departure time τ) to the total demand of OD pair j at departure time τ

$D^r_{(j,\tau)}$ = demand volume in regular demand pattern for origin-destination pair j during departure time interval τ

$\tilde{D}^r_{(j,\tau)}$ = *a priori* estimate of regular demand volume for origin-destination pair j during departure time interval τ

$\mu_{(j,\tau)}$ = structural demand deviation of from *a priori* estimate

$\tilde{D}^{r}_{(j,\tau)}$ for OD pair j with departure time τ

$\varepsilon_{(j,\tau)}$ = error term in approximating true demand for OD pair j with departure time τ

$\mu'_{(j,\tau)}$, $\mu''_{(j,\tau)}$, $\mu'''_{(j,\tau)}$ = first, second and third-order derivatives of demand deviation $\mu_{(j,\tau)}$, respectively

p = order index of a polynomial model

$\mu^{(p)}_{(j,\tau)}$ = p^{th}-order derivative of demand deviation $\mu_{(j,\tau)}$

m = maximum order of a polynomial model

$w^{(p)}_{(j,\tau)}$ = evolution noise for p^{th}-order derivative of demand deviation $\mu_{(j,\tau)}$

$u_{(i,t)}$ = combined error term in the estimation of link observation $c_{(i,t)}$ due to inconsistencies in assumptions about traffic assignment, traffic control and flow propagation, as well as measurement noise

$v_{(i,t)}$ = combined error term due to $u_{(i,t)}$ and $\varepsilon_{(j,\tau)}$ for link observation $c_{(i,t)}$

$\hat{D}_{(j,\tau)}$ = estimated mean value of$D_{(j,\tau)}$

$\hat{\mu}^{(p)}_{(j,\tau)}$ = estimated mean value of$\mu^{(p)}_{(j,\tau)}$

Z_k = state variable vector at stage k

Y_k = measurement vector at stage k

H_k = measurement matrix, relating measurement Y_k and state Z_k

w_k = process noise at stage k

v_k = measurement noise at stage k

$\hat{Z}_{k,k-1}$ = prediction of Z_k using observations up to stage $k-1$,
i.e., $E\left(Z_k\left|Y_1, Y_2, \cdots, Y_{k-1}\right.\right)$

$\hat{Z}_{k,k}$ = estimation of Z_k using observations up to stage k,
i.e., $E\left(Z_k\left|Y_1, Y_2, \cdots, Y_k\right.\right)$

$P_{k,k-1}$ = predicted state covariance matrix of Z_k at stage $k-1$,
i.e., $Var(Z_k - \hat{Z}_{k,k-1})$

$P_{k,k}$ = estimated state covariance matrix of $Z_k$at stage k, i.e., $Var(Z_k - \hat{Z}_{k,k})$

Transition Equation.

The objective of the dynamic OD demand estimation and prediction problem is to find the time-dependent demand $D_{(j,\tau)}$for origin-destination pair j at departure time interval τ. As discussed previously, the true demand $D_{(j,\tau)}$ can be partitioned into three components, namely, the regular pattern, structural deviations and random fluctuations. Theoretically, only the *a priori* estimate $\tilde{D}^{r}_{(j,\tau)}$ of the regular demand, reflecting prior survey data and surveillance information up to the previous day, is available before performing real-time estimation on the current day. For this reason, the true demand $D_{(j,\tau)}$ in the following study is modeled as a linear combination of the a priori estimate, structural deviation and random disturbance:

$$D_{(j,\tau)} = \tilde{D}^{r}_{(j,\tau)} + \mu_{(j,\tau)} + \varepsilon_{(j,\tau)}, \tag{16.1}$$

where the random disturbance term is assumed to follow a Normal distribution with zero mean. Moreover, a polynomial trend model is introduced to describe the structural deviations based on the following assumption:

Assumption 1. (Polynomial trend) Deviation at time τ + ζ can be adequately represented *locally* by an m^{th}-order polynomial function as Equation (16.2) near time τ for a small value of ζ, while derivatives of higher orders are assumed to be zero: $\mu^{(p)}_{(j,\tau)}$ = 0 for $p > m$.

$$\mu_{(j,\tau+\zeta)} = b_0 + b_1\zeta + b_2\zeta^2 + \cdots + b_p\zeta^p + \cdots + b_m\zeta^m. \tag{16.2}$$

From Taylor's theorem, the smooth function of $\mu_{(j,\tau+\zeta)}$ can be expanded about the point $\mu_{(j,\tau)}$ as

$$\mu_{(j,\tau+\zeta)} = \mu_{(j,\tau)} + \zeta\mu'_{(j,\tau)} + \frac{\zeta^2}{2!}\mu''_{(j,\tau)} + \cdots + \frac{\zeta^p}{(p)!}\mu^{(p)}_{(j,\tau)} + \cdots + \frac{\zeta^m}{(m)!}\mu^{(m)}_{(j,\tau)}. \tag{16.3}$$

A comparison of Equations (16.2) and (16.3) indicates that the polynomial coefficients in the original functional form can be obtained directly from

$$b_p = \frac{\mu^{(p)}_{(j,\tau)}}{p!}. \tag{16.4}$$

A more compact form for the p^{th}-order derivative of a polynomial can be generalized as

$$\mu^{(p)}_{(j,\tau+\zeta)} = \sum_{s=p}^{m} \frac{\zeta^{(s-p)}}{(s-p)!}\mu^{(s)}_{(j,\tau)}. \tag{16.5}$$

The corresponding matrix representation for a third-order polynomial model can be expressed as

$$\begin{pmatrix} \mu_{(j,\tau+\zeta)} \\ \mu'_{(j,\tau+\zeta)} \\ \mu''_{(j,\tau+\zeta)} \\ \mu'''_{(j,\tau+\zeta)} \end{pmatrix} = \begin{bmatrix} 1 & \zeta & \zeta^2/2! & \zeta^3/3! \\ & 1 & \zeta & \zeta^2/2! \\ & & 1 & \zeta \\ & & & 1 \end{bmatrix} \begin{pmatrix} \mu_{(j,\tau)} \\ \mu'_{(j,\tau)} \\ \mu''_{(j,\tau)} \\ \mu'''_{(j,\tau)} \end{pmatrix}. \tag{16.6}$$

The next assumption is required to allow time-varying trends to evolve stochastically between time stages.

Assumption 2. (Evolution process) From stage k to stage $k + 1$, the change of derivative $\mu^{(p)}_{(j,\tau)}$ can be described as

$$\mu^{(p)}_{(j,\tau+l)} = \sum_{s=p}^{m} \frac{l^{(s-p)}}{(s-p)!}\mu^{(s)}_{(j,\tau)} + w^{(p)}_{(j,\tau)} \tag{16.7}$$

where departure time index $\tau = kl$, and $w^{(p)}_{(j,\tau)} \sim N[0, W^{(p)}_{(j,\tau)}]$.

Taking a third-order polynomial trend model for OD pair j as an example, the corresponding transition equation in the Kalman filtering formulation is given in (16.8). More precisely, the state vector consists of the zeroth to mth-order derivatives of demand structural deviations from the *a priori* regular demand pattern estimate for OD pair j. Note that the transition matrix is independent of the current stage k and related departure time interval τ.

$$\begin{pmatrix} \mu_{(j,\tau+l)} \\ \mu'_{(j,\tau+l)} \\ \mu''_{(j,\tau+l)} \\ \mu'''_{(j,\tau+l)} \end{pmatrix} = \begin{bmatrix} 1 & l & l^2/2! & l^3/3! \\ & 1 & l & l^2/2! \\ & & 1 & l \\ & & & 1 \end{bmatrix} \begin{pmatrix} \mu_{(j,\tau)} \\ \mu'_{(j,\tau)} \\ \mu''_{(j,\tau)} \\ \mu'''_{(j,\tau)} \end{pmatrix} + \begin{pmatrix} w_{(j,\tau)} \\ w'_{(j,\tau)} \\ w''_{(j,\tau)} \\ w'''_{(j,\tau)} \end{pmatrix}. \tag{16.8}$$

Consequently, the single OD-pair model above can be easily extended to consider all the OD pairs in a network. Considering a third-order polynomial filter with departure time τ=*kl* at stage k, we can define the state vector as

$$\begin{aligned} Z_k = (&\mu_{(1,\tau)}, \mu'_{(1,\tau)}, \mu''_{(1,\tau)}, \mu'''_{(1,\tau)}, \mu_{(2,\tau)}, \mu'_{(2,\tau)}, \mu''_{(2,\tau)}, \mu'''_{(2,\tau)}, \ldots, \ldots, \\ &\mu_{(N_{od},\tau)}, \mu'_{(N_{od},\tau)}, \mu''_{(N_{od},\tau)}, \mu'''_{(N_{od},\tau)})^T \end{aligned} \tag{16.9}$$

and the transition matrix as

$$A_k = Diag(A_k^1, A_k^2, \ldots, A_k^j, \ldots, A_k^{N_{od}}) \tag{16.10}$$

where

$$A_k^j = \begin{bmatrix} 1 & l & l^2/2! & l^3/3! \\ & 1 & l & l^2/2! \\ & & 1 & l \\ & & & 1 \end{bmatrix} \tag{16.11}$$

for $j = 1, 2, \ldots, N_{od}$.

By assuming the evolution noise w_k as

$$\begin{aligned} w_k = (&w_{(1,\tau)}, w'_{(1,\tau)}, w''_{(1,\tau)}, w'''_{(1,\tau)}, w_{(2,\tau)}, w'_{(2,\tau)}, w''_{(2,\tau)}, w'''_{(2,\tau)}, \ldots, \\ &w_{(N_{od},\tau)}, w'_{(N_{od},\tau)}, w''_{(N_{od},\tau)}, w'''_{(N_{od},\tau)})^T, \end{aligned} \tag{16.12}$$

the complete transition equation in real-time OD estimation and prediction can be written as

$$Z_{k+1} = A_k Z_k + w_k. \tag{16.13}$$

To obtain the future demand level with prediction horizon h, we need to first predict the demand deviation at time τ + h based on estimated derivatives at the current stage, and then substitute the predicted demand deviation and the *a priori* estimate of the regular demand pattern into Equation (16.1). Thus,

$$\begin{aligned} E\left[D_{(j,\tau+h)} \left| \mu_{(j,\tau)} \right.\right] &= \tilde{D}^r_{(j,\tau+h)} + E\left[\mu_{(j,\tau+h)} \left| \mu_{(j,\tau)} \right.\right] \\ &= \tilde{D}^r_{(j,\tau+h)} + \sum_{s=0}^{m} \frac{h^s}{s!} \hat{\mu}^{(s)}_{(j,\tau)} \end{aligned} \tag{16.14}$$

where τ=*kl*.

By incorporating the *a priori* estimate of the regular demand pattern in the proposed structural model, one computational advantage is the reduction in the dimension of the state variable vector. To show this, suppose the original demand can be adequately fitted by an sth-order polynomial model as

$$D_{(j,\tau+\zeta)} = a_0 + a_1\zeta + a_2\zeta^2 + \cdots + a_m\zeta^m + \cdots + a_s\zeta^s + \varepsilon_{(j,\tau+\zeta)}. \qquad (16.15)$$

If $\tilde{D}^r_{(j,\tau+\zeta)}$ is a good approximation to$D_{(j,\tau+\zeta)}$, it can be further assumed that $\tilde{D}^r_{(j,\tau+\zeta)}$ also corresponds to an sth-order polynomial model satisfying $(a_p - \tilde{a}^r_p) = 0$ for $p > m$, as

$$\tilde{D}^r_{(j,\tau+\zeta)} = \tilde{a}^r_0 + \tilde{a}^r_1\zeta + \tilde{a}^r_2\zeta^2 + \cdots + \tilde{a}^r_m\zeta^m + \cdots + \tilde{a}^r_s\zeta^s. \qquad (16.16)$$

Then, ignoring the higher-order terms from m+1 to s, we have

$$\begin{aligned} \mu_{(j,\tau+\zeta)} &= D_{(j,\tau+\zeta)} - \tilde{D}^r_{(j,\tau+\zeta)} - \varepsilon_{(j,\tau+\zeta)} \\ &= (a_0 - \tilde{a}^r_0) + (a_1 - \tilde{a}^r_1)\zeta + \cdots + (a_m - \tilde{a}^r_m)\zeta^m + \cdots + (a_s - \tilde{a}^r_s)\zeta^s \\ &= (a_0 - \tilde{a}^r_0) + (a_1 - \tilde{a}^r_1)\zeta + \cdots + (a_m - \tilde{a}^r_m)\zeta^m \end{aligned} \qquad (16.17)$$

and the resulting order of the polynomial model will be reduced from s to m. Since the computational complexity of the Kalman filter is on the order of O(N^3), where $N = N_{od} \times m$ in our case, the reduction of the model order from s to m dramatically decreases the size of the state vector, and therefore improves computational time efficiency.

In addition, incorporating a reliable estimate of the regular demand pattern is always beneficial for improving estimation and prediction quality. From the linear regression standpoint, the regular daily pattern can be viewed as a good explanatory regressor that absorbs a considerable amount of variation in the independent variable (i.e. true dynamic demand). Thus, compared to a pure polynomial model, the proposed structural model with the regular pattern component leads to smaller regression residual errors, that is, smaller estimation and prediction errors.

Measurement Equations. In general, the measurement equation connects the link observations and OD demands through a link proportion matrix, as in Equation (16.18). Specifically, the link proportions map all the lagged and prevailing demands at the current stage k to n*l measurements for each link with available observations.

$$c_{(i,t)} = \sum_{\zeta=-q}^{l-1} \sum_{j=1}^{N_{od}} \left(LP_{(i,t),(j,\tau+\zeta)} * D_{(j,\tau+\zeta)} \right) + u_{(i,t)} \qquad (16.18)$$

where $\tau = kl$, i=1, 2, ..., N_{obs} and $knl \le t \le (k+1)nl - 1$.

To relate link measurements to the state variables constructed previously, substituting Equations (16.1) and (16.3), the equation above becomes

$$c_{(i,t)} = \sum_{\zeta=-q}^{l-1} \sum_{j=1}^{N_{od}} \left[LP_{(i,t),(j,\tau+\zeta)} * (\sum_{s=0}^{m} \frac{\zeta^s}{s!} \mu^{(s)}_{(j,\tau)} + \tilde{D}^r_{(j,\tau+\zeta)} + \varepsilon_{(j,\tau+\zeta)}) \right] + u_{(i,t)}, \qquad (16.19)$$

and it can be further transformed to

$$\begin{aligned}
&c_{(i,t)} - \sum_{\zeta=-q}^{l-1}\sum_{j=1}^{N_{od}}\Big(LP_{(i,t),(j,\tau+\zeta)} * \tilde{D}^{r}_{(j,\tau+\zeta)}\Big) \\
&= \sum_{\zeta=-q}^{l-1}\sum_{j=1}^{N_{od}}\left(LP_{(i,t),(j,\tau+\zeta)} * \sum_{s=0}^{m}\frac{\zeta^{s}}{s!}\mu^{(s)}_{(j,\tau)}\right) \\
&+ \sum_{\zeta=-q}^{l-1}\sum_{j=1}^{N_{od}}\big(LP_{(i,t),(j,\tau+\zeta)} * \varepsilon_{(i,t+\zeta)}\big) + u_{(i,t)}.
\end{aligned} \tag{16.20}$$

Consequently, we can define the observation vector and measurement error in the Kalman formulation as follows:

$$Y_k = H_k Z_k + v_k, \tag{16.21}$$

$$\begin{aligned}
Y_k = \big(&y_{(1,knl)}, y_{(1,knl+1)}, \ldots, y_{(1,(k+1)nl-1)}, \ldots, \\
&y_{(N_{obs},knl)}, y_{(N_{obs},knl+1)}, \ldots, y_{(N_{obs},(k+1)nl-1)}\big)^{T}
\end{aligned} \tag{16.22}$$

where

$$y_{(i,t)} = c_{(i,t)} - \sum_{\zeta=-q}^{l-1}\sum_{j=1}^{N_{od}}\left(LP_{(i,t),(j,\tau+\zeta)} * \tilde{D}^{r}_{(j,\tau+\zeta)}\right), \tag{16.23}$$

$$\begin{aligned}
v_k = \big(&v_{(1,knl)}, v_{(1,knl+1)}, \ldots, v_{(1,(k+1)nl-1)}, \ldots, \\
&v_{(N_{obs},knl)}, v_{(N_{obs},knl+1)}, \ldots, v_{(N_{obs},(k+1)nl-1)}\big)^{T}
\end{aligned} \tag{16.24}$$

where

$$v_{(i,t)} = \sum_{\zeta=-q}^{l-1}\sum_{j=1}^{N_{od}}\big(LP_{(i,t),(j,\tau+\zeta)} * \varepsilon_{(j,\tau+\zeta)}\big) + u_{(i,t)}. \tag{16.25}$$

The final measurement error term in the transition equation combines the random noise in OD demand, other errors associated with link proportions, as well as the sensor errors in traffic measurements.

The dimension of measurement matrix H$_k$ is $(N_{obs}$*nl,N_{od}*$m)$, and its $(i,t)^{th}$, $(j,p)^{th}$ element is

$$H_{(i,t),(j,p)} = \sum_{\zeta=-q}^{l-1}\left(LP_{(i,t),(j,\tau+\zeta)} * \frac{\zeta^{p}}{p!}\right) \tag{16.26}$$

where $\tau = kl$.

By applying a polynomial approximation for OD demands during departure time intervals from *kl*–*q* to $(k{+}1)l$-1, the polynomial trend filter neatly incorporates the lagged demands into the estimation procedure for the current stage, leading to an efficient state space representation, desirable for large-scale network applications.

Assumption 3. w_k and v_k are white noise terms uncorrelated with the initial state Z_0 and with each other, where $w_k \sim N[0, W_k]$ and v_k ~N(0, V_k).

From Assumption 3, the Kalman filtering algorithm is ready to be integrated into the following recursive estimation and prediction algorithm.

Algorithm 2. *Real-time dynamic demand OD estimation and prediction*
Step 1: (Initialization) Set up initial estimates $P_{0,0} = Var(Z_0)$ and $\hat{Z}_{0,0} = E(Z_0)$. Let $k = 1$.
Step 2: (Prediction) Propagate the mean and covariance estimates from $k - 1$ to k.

$$\hat{Z}_{k,k-1} = A_k \hat{Z}_{k-1,k-1} \quad (16.27)$$
$$P_{k,k-1} = A_k P_{k-1,k-1} A_k' + W_k \quad (16.28)$$

Step 3: (Estimation of state variable) After receiving new link proportions and link observations, calculate the weighting matrix as

$$K_k = P_{k,k-1} H_k^T (H_k P_{k,k-1} H_k^T + V_k)^{-1}, \quad (16.29)$$

and then update the a posteriori mean and covariance estimates.

$$\hat{Z}_{k,k} = \hat{Z}_{k,k-1} + K_k (Y_k - H_k \hat{Z}_{k,k-1}) \quad (16.30)$$
$$P_{k,k} = (I - K_k H_k) P_{k,k-1} \quad (16.31)$$

Step 4: (Estimation of real-time demand) Calculate the estimation of real-time demand using new estimates $\hat{\mu}_{(j,\tau)}$.

$$\hat{D}_{(j,\tau)} = E\left(\tilde{D}^r_{(j,\tau)} + \mu_{(j,\tau)} + \varepsilon_{(j,\tau)}\right) = \tilde{D}^r_{(j,\tau)} + \hat{\mu}_{(j,\tau)} \quad (16.32)$$

where τ=*kl,kl+1,. . . , (k*+1)$l-1$.
Step 5: Advance roll period forward from k to k+1, and then go back to Step 2.
Further, if independence of measurement errors is assumed, that is,

$$v_k \sim N(0, diag[V_{(1,knl)}, V_{(1,knl+1)}, \ldots, \ldots, V_{(N_{obs},(k+1)nl-1)}]), \quad (16.33)$$

we can apply the scalar updating scheme described in Ashok [1] in order to avoid complicated matrix inversion in a real-time setting.

In the context of short-term economic forecasting, the zeroth, first and second-order polynomial models can be viewed as the local level model, local linear growth model and local quadratic growth model, respectively. Regarding connections between the polynomial trend models and other time-series ARIMA models, West and Harrison [34] demonstrated that, if restrictions are imposed on the autocorrelation structure, the limiting case of an (m+1)th polynomial trend model is equivalent to an ARIMA(0, m, m) model. According to the generalized state-space architecture proposed by Harvey [16], an auto-regression (AR) term can be also incorporated into the state variable vector to model the autocorrelation structure in the random disturbance. It is worth remarking that, even if the underlying trends for demand structural deviations are negligible, it is advisable to embed a polynomial trend component in the space state representation so as to monitor and identify possible changes in the process structure.

16.4 Adaptive Day-To-Day Updating of Regular Demand Pattern Information

As discussed earlier, the initial estimate for the regular demand pattern could be unreliable due to limited sample size, and the normal daily pattern could evolve smoothly due to day-to-day demand dynamics. Hence, it is necessary to update the *a priori* estimate using the new demand estimate and new observations. A desirable updating formulation should be able to adaptively recognize and capture the systematic day-to-day evolution, and also maintain robustness under disruptions due to special events. An updating formulation based on a Kalman filter framework is proposed.

The notation used in the real-time OD estimation and prediction model is extended to the day-to-day context as follows:

d = index for day

D_d^r = state variable vector of regular OD demand pattern on day d,consisting of elements $D_{(j,\tau)}^r$

ξ_d = day-to-day evolution variance on day d

$\hat{D}_d$ = vector of the real-time demand estimate on day d, consisting of estimates $\hat{D}_{(j,\tau)}$

η_d = measurement variance matrix on day d

$\hat{D}_{d,d-1}^r$ = predicted state variable vector D_d^r using observations up to day $d-1$, consisting of elements $\tilde{D}_{(j,\tau)}^r$

$\hat{D}_{d,d}^r$ = estimated state variable vector D_d^r using observations up to day d

$\Sigma_{d,d-1}$ = predicted state covariance matrix for the regular demand pattern on day d

$\Sigma_{d,d}$ = estimated state covariance matrix for the regular demand pattern on day d

K_d = Kalman gain matrix for using real-time demand estimates on day d

$\hat{\mathrm{M}}_d$ = vector of estimated demand deviations on day d, with elements $\hat{\mu}_{(j,\tau)}$

K_d' = Kalman gain matrix for using real-time observations on day d

C_d = vector of traffic observations on day d, consisting of elements $c_{(i,t)}$

LP_d = link proportion matrix on day d, consisting of elements $LP_{(i,t),(j,\tau)}$

The transition and measurement equations for the day-to-day demand evolution can be written as

Transition Equation:
$$D_{d+1}^r = D_d^r + \xi_d. \tag{16.34}$$

Measurement Equation:
$$\hat{D}_d = D_d^r + \eta_d. \tag{16.35}$$

Assumption 4. ξ_d and η_d are white noise terms uncorrelated with the initial state D_0^r and with each other, where ξ_d ~N(0, Q_d) and η_d ~N(0, R_d)

According to transition equation (16.34), the regular demand pattern can evolve smoothly from day to day, where stochastic day-to-day evolution is captured by the evolution random term ξ_d with zero mean. In the measurement equation, since the true demand state cannot be directly observed, the new real-time demand estimate $\hat{D}_d$ is considered as “measurement” incoming every day. Following the standard Kalman filtering algorithm, the updating procedure can be summarized as follows.

Algorithm 3. *Day-to-day updating for regular demand pattern estimate*
Step 1: (Initialization) Set up $\hat{D}^{r}_{0,0}$ and $\Sigma_{0,0}$ as the initial estimated mean and covariance of the regular demand. Let d=1.
Step 2: (Computation of *a priori* estimate) The *a posteriori* estimate $\hat{D}^{r}_{d-1,d-1}$ on previous day d-1 is used as the *a priori* estimate for current day d. The corresponding covariance matrix is updated by taking evolution noise into account:

$$\hat{D}^{r}_{d,d-1} = \hat{D}^{r}_{d-1,d-1}, \tag{16.36}$$
$$\Sigma_{d,d-1} = \Sigma_{d-1,d-1} + Q_d. \tag{16.37}$$

Step 3: (Real-time OD estimation and prediction) Run the real-time OD estimation and prediction module in conjunction with real-time DTA simulators, to obtain new estimates $\hat{M}_d$ and$\hat{D}_d$ for day d.
Step 4: (Update of gain matrix) Compute the gain matrix using predicted state covariance matrix and measurement variance matrix:

$$K_d = \Sigma_{d,d-1}(\Sigma_{d,d-1} + R_d)^{-1}. \tag{16.38}$$

Step 5: (Update of mean and covariance) Update the estimated mean and covariance matrix for the regular demand state vector:

$$\begin{aligned}\hat{D}^{r}_{d,d} &= \hat{D}^{r}_{d,d-1} + K_d(\hat{D}_d - \hat{D}^{r}_{d,d-1}) \\ &= \hat{D}^{r}_{d,d-1} + K_d(\hat{D}_d - \tilde{D}^{r}_{d}) \\ &= \hat{D}^{r}_{d,d-1} + K_d\hat{M}_d, \end{aligned} \tag{16.39}$$
$$\Sigma_{d,d} = (I - K_d)\Sigma_{d,d-1}. \tag{16.40}$$

Step 6: Move to the next day, $d = d$+1, and then go back to Step 1.

An important point is that $\hat{D}_d - \hat{D}^{r}_{d,d-1}$ in the day-to-day updating algorithm above is equivalent to $\hat{\mathrm{M}}_d$, which consists of the demand deviation estimate $\hat{\mu}_{(j,\tau)}$ generated from the real-time estimation and prediction algorithm on day d. It can be shown that $\Sigma_{d,d} \leq \Sigma_{d,d-1}$, that is, the conditional demand estimate contains less uncertainty than the corresponding *a priori* estimate for the regular demand pattern, after incorporating additional information from the new real-time estimation result. More importantly, the recursive updating algorithm above naturally integrates with the proposed structural model in the previous section, and it is able to accumulate the information from the real-time estimator on a daily basis.

In order to make this recursive algorithm operational, the next question is how to specify the values of evolution variance Q_d and measurement varianceR_d. By using the multi-day OD estimation method proposed by Zhou et al. [35], the variance of the measurement noise in Equation (16.35) can be obtained by evaluating the variance of estimated OD demands across several days. Determining the day-to-day variance is generally more difficult, since we cannot directly observe the day-to-day demand evolution process. Recognizing that the proposed day-to-day evolution process can be described as a random walk plus noise model, existing time series techniques are applied to choose appropriate values of process variance. A common approach is to first assume a constant signal to noise ratio$\lambda = \frac{Q_d}{R_d}$, indicating the ratio of inherent system variance with respect to observational variance. Based on calibrated R_d, $Q_d = \lambda R_d$, so we can select an appropriate signal to noise ratio so as to minimize average prediction errors in the training data sets. The reader is referred to West and Harrison [34] for a comprehensive treatment. In early stages of applying this updating mechanism, considerable uncertainty in the predicted state covariance $\Sigma_{d,d-1}$ results in a high gain factor, implying that the new real-time estimates receive relatively large weighting. After a certain number of iterations, the gain factor becomes stable as it is gradually reaching a steady state. If constant Q and R are assumed, the following limiting behavior for the Kalman gain matrix can be derived [34]:

$$\lim_{d\to\infty} K_d = \frac{\lambda}{2}\left(\sqrt{1+{}^4\!/\!_\lambda}-1\right). \tag{16.41}$$

A typical value of λ can be 0.05, leading to $\lim\limits_{d\to\infty} K_d = 0.2$, so the most recent real-time estimate receives relatively small weighting eventually. If λ=0.5, corresponding to a limiting gain factor of 0.5, then the *a priori* estimate and the new real-time estimate share equal importance in determining a priori demand information for the next day.

Another approach to update the historical demand estimate is to directly utilize link observations instead of real-time demand estimates. The modified transition and measurement equations are given in Equations (16.42) and (16.43):

Transition Equation:

$$D^r_{d+1} = D^r_d + \xi_d. \tag{16.42}$$

Measurement Equation:

$$C_d = LP_d D^r_d + \eta'_d. \tag{16.43}$$

Assumption 5. ξ_d and η'_d are white noise terms uncorrelated with the initial state D^r_0 and with each other, where ξ_d ~N(0, Q_d) and η'_d ~N(0, R'_d).

Note that the transition equation above is identical to the one using real-time estimates. As the measurement equation uses link proportions to link real-world measurements and the regular demand pattern, the measurement variance R'_d should be recalibrated accordingly. The gain matrix, mean and covariance updating formulation in Steps 4 and 5 of Algorithm 3 should also be changed to the following:

$$K'_d = \Sigma_{d,d-1} LP_d^T (LP_d \Sigma_{d,d-1} LP_d^T + R'_d)^{-1}, \tag{16.44}$$

$$D^r_{d,d} = D^r_{d,d-1} + K'_d (C_d - LP_d D^r_{d,d-1}), \tag{16.45}$$

$$\Sigma_{d,d} = (I - K'_d)\Sigma_{d,d-1}. \tag{16.46}$$

This Kalman filtering formulation provides a least-squares unbiased estimator for the regular demand pattern, with the optimal weights on the *a priori* estimate and new information. It is important to note that this recursive prediction-correction algorithm only requires *a priori* mean and covariance statistics at each iteration instead of the entire historical data series, resulting in efficient storage implementation for on-line applications. Practically, this updating method can be viewed as a moving average method with adaptive weights, depending on the respective reliability of the *a priori* and real-time information sources.

16.5 Application to Irvine Network Using PeMS Data

Numerical experiments are presented to illustrate the application of the proposed model and algorithms to the Irvine, CA test bed traffic network, which consists of three freeway corridors (I-5, I-405, Highway 133) and other main arterials. As shown in Figure 16.2, this network includes 61 OD zones, 326 nodes and 626 links, where traffic counts are measured at 30-second intervals on 19 freeway links and at 5-minute intervals on 28 arterial links. In addition, the *a priori* estimate of the regular demand pattern is constructed by the off-line OD estimation method [30],[35] using the first day data. Real-world observations on the second day are used to calibrate the system and measurement variances in the real-time OD estimation and prediction model. The third day data are used to validate the proposed real-time OD estimation and prediction algorithm. The time of interest in the following experiments is the morning peak period (4:00 AM – 10:00 AM), while the demand departure time interval and roll period are 5 and 15 minutes, respectively.

First, a first-order polynomial trend model (i.e. local linear trend model) is applied to estimate the demand deviations from the *a priori* estimate of the regular demand pattern. In Figure 16.3, the *a priori* regular pattern estimate and the corresponding demand deviations are displayed for the OD pair from zone 53 to zone 40, which carries the highest trips for all among all the OD pairs in the study network. On average, the *a priori* demand data underestimates the real-time demand on the third day for this selected OD pair, but the prior information still shows similar time-varying dynamic patterns. As expected, the demand deviations exhibit a much slower changing pattern than the corresponding real-time demands over the same time. Essentially, the estimated structural demand deviations are caused by the day-to-day dynamics, but the deviations shown in this case can be also due to the estimation noise in the *a priori* demand data, which only utilizes one-day observations.

Considering the smooth trend for demand deviations in Figure 16.3, it is desirable to further reduce the first-order polynomial model to the zeroth-order polynomial model. To assess and compare the estimation performance of alternative models, we define root mean square (RMS) error in density as

$$RMS_t = \sqrt{\frac{\sum_i (c_{i,t} - \hat{c}_{i,t})}{N_{obs}}} \tag{16.47}$$

where $c_{i,t}$= density measured on link i, during observation interval t, and $\hat{c}_{i,t}$ = simulated density from the real-time DTA estimator on link i, during observation interval t.

The RSE errors at every 5 minutes are plotted in Figure 16.4, for the zeroth and first-order polynomial models, respectively. The average RMS error of the zeroth-order model during the study horizon (10.2608) is marginally greater than the average RMS error of the first-order model (10.8588) by 1.8%. Basically, these two models perform better in the early morning (from 4:00 AM to 6:00 AM), compared to the peak hour period (from 7:00 AM to 9:00 AM). Such time-dependent RMS measures can be explained by the increasing variability in the peak hour demands

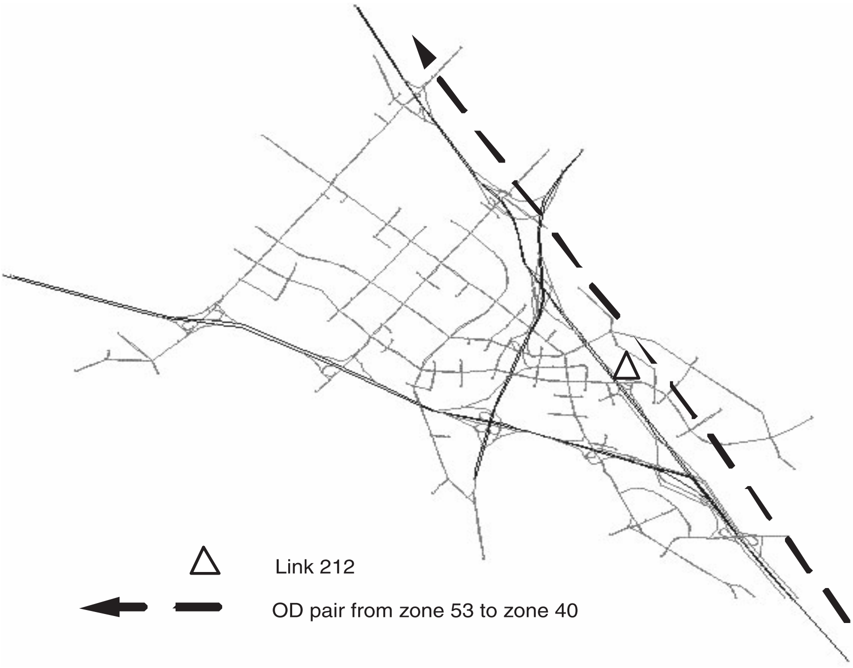


Fig. 16.2. Irvine network.

and the high dynamics in the related traffic flow propagation processes. Based on experiment results from the third day data, the zeroth order polynomial model seems to be more attractive than the first-order model, since it offers acceptable accuracy with considerable enhanced computation efficiency. On the other hand, if real-time response constraints can be satisfied, keeping a high order polynomial model is always preferable, because it can capture nonlinear patterns in the demand structure changes.

Figure 16.5 plots simulated density, predicted density and the observed density on link 212, using a 20-minute prediction horizon. Specifically, the simulated density and the predicted density are generated from the DTA network state estimation module and the DTA state prediction module, respectively. Link 212 is a freeway link going northbound, and its location is marked in Figure 16.2. The DTA network state estimator is able to capture the time-varying trends of real-world traffic, while the DTA network state predictor can forecast dynamic flow propagation with acceptable quality. The results above further validate the effectiveness of the proposed real-time OD estimation and prediction framework, and illustrate the role of the PeMS database in enabling and improving this process.

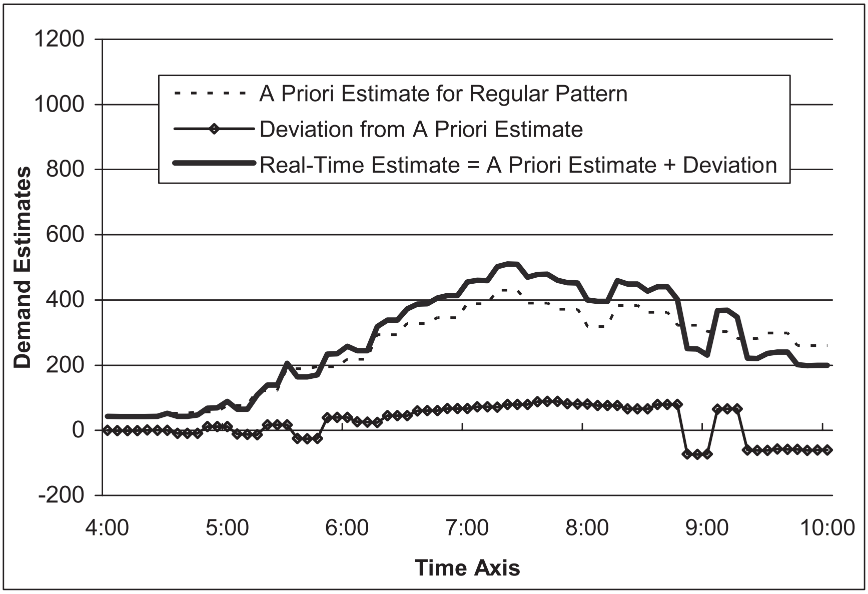


Fig. 16.3. Dynamic demand estimates for OD pair from zone 53 to zone 40.

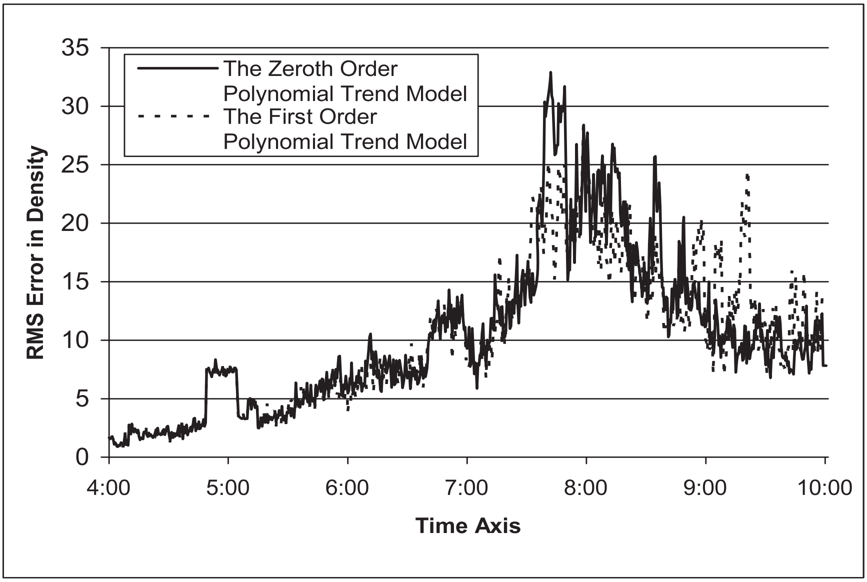


Fig. 16.4. RMS errors in density for polynomial models.

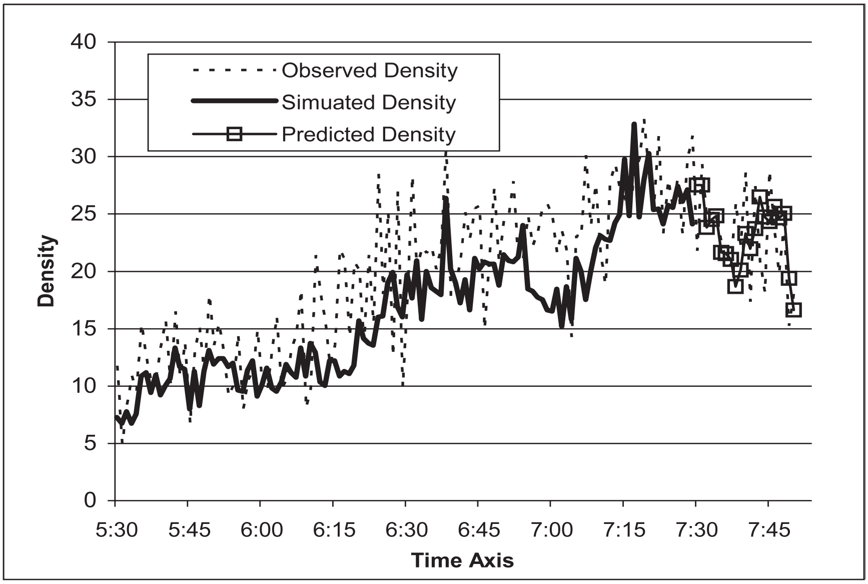


Fig. 16.5. Observed density vs. simulated density and predicted density on link 212.

16.6 Concluding Comments

Real-time OD estimation and prediction is an important component in real-time dynamic traffic assignment for ATMS/ATIS network applications. This chapter exploits the potential of using a structural state space model to systematically incorporate regular demand pattern information, structural deviations and random fluctuations. The contributions of this study include the following. First, a polynomial trend filter is developed to estimate and predict demand deviations from the *a priori* estimate of the regular demand pattern, so as to utilize valuable historical information and adaptively respond to possible structural deviations in demands. Second, based on a Kalman filtering framework, an optimal recursive procedure is proposed for updating the regular demand pattern estimate with new real-time estimates and observations obtained every day. These models can be naturally integrated into the real-time DTA system and provide an effective and efficient approach to utilize the real-time traffic data continuously in the operational settings. Third, the application to the Irvine network provides an illustration of the usefulness of the PeMS data to calibrate and test advanced network modeling tools intended for ITS planning and operation.

One particularly attractive opportunity would be to integrate our dynamic network modeling capability explicitly with PeMS into the decision support system. The day-to-day updating framework would then be used to regularly update the modeling tools, keeping an up-to-the-minute version ready for application to analyze scenarios and contemplated actions for unfolding conditions. An important capability in this process would be the identification, through the extensive PeMS data warehouse, of most comparable days or patterns in order to provide a starting point, to be updated using the methods presented in this work.

Through the efforts of Varaiya, his students and collaborators, the theory and practice of traffic systems engineering is making strides towards greater levels of intelligence in terms of information on prevailing conditions as well as dynamic management strategies.

Acknowledgments. This chapter is based in part on research funded through the US Department of Transportation. The work has benefited from the contributions of several current and former graduate students, including Xiao Qin, Hossein Tavana, Ying Kang and Hayssam Sbayti. The authors are of course responsible for all results and opinions expressed in this chapter.

References

[1] K. Ashok, Estimation and Prediction of Time-Dependent Origin-Destination Flows, Ph.D. Dissertation, MIT, 1996.

[2] K. Ashok and M. Ben-Akiva, Dynamic origin-destination matrix estimation and prediction for real-time traffic management systems, *Proceedings of the 12th International Symposium on Transportation*, C. Daganzo (ed), Elsevier, 1993.

[3] K. Ashok and M. Ben-Akiva, Alternative approaches for real-time estimation and prediction of time- dependent origin-destination flows, *Transportation Science*, **34** (1):21–36, 2000.

[4] M. Broucke and P. Varaiya, A theory of traffic flow for automated highway systems, *Transportation Research*, Part C, **4** (4):181–210, 1996.

[5] E. Cascetta, D. Inaudi and G. Marquis, Dynamic estimators of origin-destination matrices using traffic counts, *Transportation Science*, **27** (4):363–373, 1993.

[6] G.L. Chang and X.D. Tao, An integrated model for estimating time-varying network origin-destination distributions, *Transportation Research*, **33A** (5):381–399, 1999.

[7] G.L. Chang and J. Wu, Recursive estimation of time-varying origin-destination flows from traffic counts in freeway corridors, *Transportation Research*, **28B** (2):141–160, 1994.

[8] C. Chen, A. Skabardonis and P. Varaiya, Travel-time reliability as a measure of service, *Transportation Research Record*, **1855**:74–79, 2003.

[9] C. Chen, A. Skabardonis and P. Varaiya, Systematic identification of freeway bottlenecks, *Transportation Research Record*, **1867**:46–52, 2004.

[10] C. Chen, Z. Jia and P. Varaiya, Causes and cures of highway congestion, *IEEE Control Systems Magazine*, **21** (6):26–32, 2002.

[11] C. Chen, K. Petty, A. Skabardonis, P. Varaiya and Z. Jia, Freeway performance measurement system: mining loop detector data, *Transportation Research Record*, **1748**:96–102, 2001.

[12] C. Chui and G. Chen, *Kalman Filtering with Real-Time Applications*, Springer-Verlag, 1991.

[13] D.N. Godbole, F. Eskafi, E. Singh and P. Varaiya, Design of entry and exit maneuvers for IVHS, *Proceedings of the American Control Conference*, **5**:3576–3580, 1995.

[14] M. Cremer and H. Keller, Dynamic identification of flows from traffic counts at complex intersections, In: E. Hauer and V. Hurdle, Eds., *Proceedings of the 8th International Symposium on Transportation and Traffic Theory*, 1981.

[15] M. Cremer and H. Keller, A new class of dynamic methods for the identification of origin-destination flows, *Transportation Research*, **21B** (2):117–132, 1987.

[16] A.C. Harvey, *Forecasting Structural Time Series Models and the Kalman Filter*, Cambridge University Press, 1989.

[17] J.K. Hedrick, M. Tomizuka and P. Varaiya, Control issues in automated highway systems, *IEEE Control Systems Magazine*, **14** (6):21–32, 1994.

[18] R. Herman, Technology, human interaction, and complexity: reflections on vehicular traffic science, *Operations Research*, **40**:199–212, 1992.

[19] R. Horowitz and P. Varaiya, Control design of an automated highway system, *Proceedings of the IEEE*, **88** (7), 2000.

[20] R. Jayakrishnan, H.S. Mahmassani and T-Y. Hu, An evaluation tool for advanced traffic information and management systems in urban networks, *Transportation Research*, **2C** (3):129–147, 1994.

[21] Z. Jia, C. Chen, B. Coifman and P. Varaiya, The PeMS algorithms for accurate, real-time estimates of g-factors and speeds from single-loop detectors, *Proceedings 2001 IEEE Intelligent Transportation Systems*, 536–541, 2001.

[22] Y. Kang, Estimation and Prediction of Dynamic Origin-Destination (O-D) Demand and System Consistency Control for Real-Time Dynamic Traffic Assignment Operation, Ph.D. Dissertation, The University of Texas at Austin, 1999.

[23] J. Kwon, C. Chen and P. Varaiya, Statistical methods for detecting spatial configuration errors in traffic surveillance sensors, *Transportation Research Record*, **1870**:124–132, 2004.

[24] H.S. Mahmassani and R. Jayakrishnan, Dynamic simulation-assignment methodology to evaluate in-vehicle information strategies in urban traffic networks, *Proceedings of Winter Simulation Conference*, 763–769, New Orleans, LA, 1990.

[25] H.S. Mahmassani, Dynamic network traffic assignment and simulation methodology for advanced system management applications, *Networks and Spatial Economics*, **1-2** (3,4):267–292, 2001.

[26] H.S. Mahmassani, Y. Hawas, K. Abdelghani, A. Abdelfatah, Y.-C. Chiu and Y. Kang, DYNASMART-X; Volume II: Analytical and Algorithmic Aspects, Technical Report ST067-85-Volume II, Center for Transportation Research, The University of Texas at Austin, 1998.

[27] I. Okutani and Y. Stephanedes, Dynamic prediction of traffic volume through Kalman filtering theory, *Transportation Research*, **18B** (1):1–11, 1984.

[28] S. Peeta, and A. Ziliaskopoulos, Foundations of dynamic traffic assignment: the past, the present and the future, Networks and Spatial Economics, **1-2** (3,4):233–266, 2001.

[29] A. Puri, and P. Varaiya, Driving safely in smart cars, *Proceedings of the American Control Conference*, **5**:3597–3599, 1995.

[30] H. Tavana, Internally-Consistent Estimation of Dynamic Network Origin-Destination Flows from Intelligent Transportation Systems Data Using Bi-Level Optimization, Ph.D. Dissertation, The University of Texas at Austin, 2001.

[31] H. Tavana and H.S. Mahmassani, Estimation of dynamic origin destination flows from sensor data using bi-level optimization method, *Transportation Research Board CD-ROM Paper Preprints* (TRB Paper No. 01-3241), National Research Council, Washington D.C., 2001.

[32] P. Varaiya, Smart cars on smart roads: Problems of control, *IEEE Transactions on Automatic Control*, **38** (2):195–207, 1993.

[33] P. Varaiya, California's performance measurement system: Improving freeway efficiency through transportation intelligence, TR News, **218**:18–24, 2002.

[34] M. West and B.J. Harrison, *Bayesian Forecasting and Dynamic Models* (2nd Edition), Springer-Verlag, 1997.

[35] X. Zhou, X. Qin and H.S. Mahmassani, Dynamic origin-destination demand estimation using multi-day link traffic counts for planning applications, *Transportation Research Record*, **1831**:30–38, 2003.

17

Modeling, Estimation, and Control of Freeway Traffic

Roberto Horowitz, Laura Muñoz, and Xiaotian Sun

Department of Mechanical Engineering, University of California at Berkeley
{horowitz,lmunoz,sunx}@me.berkeley.edu

17.1 Introduction

Freeway traffic congestion is a major problem in today's metropolitan areas. It occurs regularly during commute hours. In addition, non-recurrent congestion often takes place as a result of incidents, road work, or public events. Congestion leads to a variety of detrimental effects such as inefficient operation of freeways, wasted resources, increased pollution, and intensified driver fatigue. *The 2004 Urban Mobility Report* [1] finds: "Congestion has grown everywhere in areas of all sizes. Congestion occurs during longer portions of the day and delays more travelers and goods than ever before." It was estimated in this report that, in 2002, congestion cost Americans 3.5 billion hours of delay and 5.7 billion gallons of wasted fuel, with an equivalent monetary cost of U.S. $63.2 billion. On-ramp metering has been widely used as an effective strategy to increase freeway operation efficiency. It has been recommended to the U.S. Federal Highway Administration as the No. 1 tool to address the congestion problem, other than adding more capacity to transportation infrastructures [2]. It has been reported that ramp metering was able to reduce delay by 101 million person-hours in 2002, approximately 5% of the congestion delay on freeways where ramp-metering was in effect [1].

Accurate freeway traffic models are extremely valuable tools for the design and evaluation of on-ramp metering strategies. However, the development and calibration of these models, particularly those that are microscopic, is often laborious and time-consuming. To help fulfill the goal of providing an accurate, computationally efficient and easy to calibrate model for the development and analysis of on-ramp metering algorithms, a piecewise-linearized version of Daganzo's macroscopic Cell Transmission Model (CTM) [3, 4], called the Switching-Mode Model (SMM), has been developed [5], and will be discussed in Section 17.2.2. Its linear structure lends the advantage of simplifying control analysis, design, and data-estimation methods. The CTM, which is briefly described in Section 17.2.1, has many favorable features, particularly its simplicity, ease of calibration, and its ability to reproduce important traffic phenomena such as shock wave propagation. These properties are also inherited by the SMM. Both the CTM and SMM have been shown to perform well in describing traffic behavior when tested with data from a 2-mile portion of a 14-mile test section of Interstate 210 Westbound (I-210W) in southern California [5]. Furthermore, the observability and controllability properties of the SMM, which are of fundamental importance in the design of traffic data estimators and freeway on-ramp control systems, can be derived using standard linear systems techniques, as discussed in Section 17.2.3. Moreover, in order to facilitate the calibration process of these models, a semi-automated calibration methodology for estimating the CTM and SMM parameters has been developed [6, 7], as described in Section 17.3. The semi-automated calibration procedure was also tested with data from the same 14-mile segment of I-210W, which typically endures heavy congestion during the weekday morning commute period, and the calibrated CTM was shown to reproduce observed bottleneck locations and the qualitative evolution of traffic congestion, yielding an average of about 2% error in the predicted partial total travel time [6, 7].

To effectively control freeway traffic, it is desirable that traffic data, such as flow and density, be available continuously across time and space. However, because of the high cost and difficulty of installing and maintaining loop detectors, oftentimes traffic data are not available at all desired locations at all times and, as a consequence, missing data must be estimated using available data from other locations. On the other hand, the traffic congestion mode, i.e., whether the traffic is flowing freely or is congested, cannot be measured directly and has to be inferred from other quantities. For these purposes, a traffic state estimator has been developed based on the SMM [8, 9], using a modified mixture Kalman filter (MKF) algorithm [10], as described in Section 17.4. This estimator is able to estimate the vehicle densities at unmeasured locations, as well as determine the traffic congestion mode in a freeway section. It was tested on a 2-mile section of freeway and its performance was evaluated using the measured data. It was shown that on average, a mean percentage error of $\sim$10% was achieved for vehicle density estimation at unmeasured locations. The MKF-based traffic state estimator was implemented on our entire selected 14-mi I-210W test site, as well as interfaced with both a calibrated CTM simulator [6] and a calibrated VISSIM microscopic simulator [11].

Section 17.5 discusses on-ramp metering control strategies. Traffic-responsive and decentralized on-ramp metering control schemes, such as the ALINEA algorithm of Papageorgiou [12], in which each on-ramp controller only utilizes traffic measurements from the freeway mainline near the on-ramp merge point, have been shown to be effective and in many instances perform more robustly than coordinated metering schemes. However, because traffic dynamics behave differently under free-flow or congested conditions, leading to a freeway segment having different controllability and observability properties depending on its congestion state [5], it may be advantageous to change the structure of the localized controller to suit the current controllability and observability properties of the segment. Based on this observation, a traffic responsive and decentralized switching ramp-metering controller has been developed [13, 14], which employs a different feedback structure depending on whether the mainline segment at the on-ramp merge junction is in a *free-flow* or *congested* mode, and is briefly described in Section 17.5.1.

A requirement that is often imposed on ramp-metering schemes is that queues must not exceed the storage capacity of the on-ramp, in order to prevent the queue from spilling over and inducing congestion in surface streets. It has been observed [15, 16] that the "queue-override" scheme that is most-frequently used in the U.S. leads to oscillatory behavior and under-utilization of on-ramp storage capacities. To address this problem, [16] proposed the use of a proportional on-ramp queue controller that requires knowledge of the on-ramp vehicle demand, which may not be available in real-time in the field. To avoid the use of on-ramp vehicle demand measurements, a proportional and integral queue length regulator has been proposed [13, 14], which estimates the length of the queue by measuring vehicle speeds as they approach the queue, and is discussed in Sections 17.5.2 and 17.5.3.

Section 17.5.4 discusses an overall localized and traffic-responsive on-ramp control strategy that reduces the spatial and temporal span of the congestion while maintaining on-ramp queues within on-ramp storage capacities. This control strategy switches between the mainline density ramp-metering controller in Section 17.5.1 and the queue length regulator in Section 17.5.2, by choosing the controller that produces the less restrictive metering strategy at every sampling instance. Test results of the use of this control strategy on a calibrated microscopic traffic simulator are presented and compared with results obtained when ALINEA [12] is used instead of the switching ramp-metering controller of Section 17.5.1.

Conclusions and final remarks are presented in Section 17.6.

17.2 Macroscopic Modeling and Analysis of Freeway Traffic Dynamics

Two fundamentally different approaches have typically been applied in order to model traffic dynamics. The microscopic approach seeks to reproduce the behavior of the individual driver/vehicle unit, as it responds to its environment by adjusting its speed. The macroscopic approach, on the other hand, ignores the dynamics of the individual driver and instead attempts to replicate the aggregate response of a large number of vehicles. In this section we first briefly review the cell transmission model (CTM) introduced by Daganzo [3, 4], which is a finite difference approximation of the well-known Lighthill, Whitham and Richards (LWR) model [17, 18] and is based on the intuitive concepts of sending and receiving flows. Subsequently, we describe a piecewise-linearized version of the CTM, called the switching-mode model (SMM), where the traffic dynamics in each segment of the freeway is modeled by a hybrid system that switches among several sets of linear difference equations, depending on the congestion status of the cells in the segment and the boundary conditions of the segment. The SMM forms the basis for most of the real-time estimation and on-ramp metering control algorithms that are discussed in this chapter.

17.2.1 Cell Transmission Model (CTM)

In Daganzo's CTM, a freeway is partitioned into a series of cells. An example is shown in Fig. 17.1(a), where it is assumed that nonuniform cell lengths are allowed.

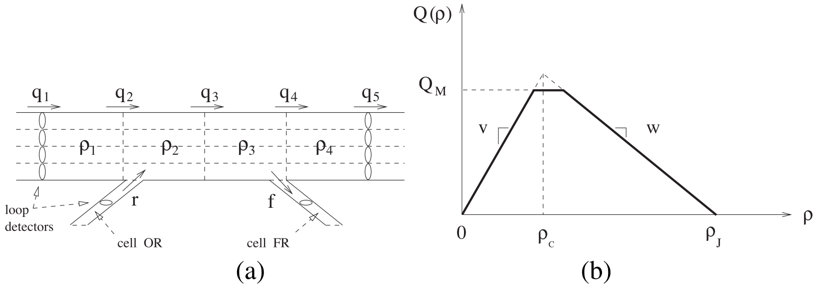


Fig. 17.1. (a) Freeway segment partitioned into cells: 1–4 on mainline, cell OR on on-ramp, and cell FR on off-ramp; (b) Trapezoidal fundamental diagram.

The traffic density in cell i evolves according to conservation of vehicles:

$$\rho_i(k+1) = \rho_i(k) + \tfrac{T_s}{l_i}\big(q_{i,in}(k) - q_{i,out}(k)\big) \tag{17.1}$$

where $q_{i,in}(k)$ and $q_{i,out}(k)$ are, respectively, the total flows, in vehicles per unit time, entering and leaving cell i during the k^{th} time interval, $T_s\ [k,\ k+1)$, including flows along the mainline and the on- and off-ramps. k is the time index, T_s is the discrete time interval, l_i is the length of cell i, and $\rho_i(k)$ is the density, in vehicles per unit length of freeway, in cell i at time $k\,T_s$. The model parameters include v, w, Q_M, and ρ_J, which are depicted in the trapezoidal fundamental diagram of Fig. 17.1(b). The parameters can be uniform over all cells or allowed to vary from cell to cell. For reference, the parameters are defined as:

v ... free-flow speed (mph)
w ... backward congestion wave speed (mph)
Q_M ... maximum allowable flow (veh/hr, i.e., vph)
ρ_J ... jam density (veh/mi, i.e., vpm)
ρ_c ... critical density (vpm)

The left part of the fundamental diagram of Fig. 17.1(b), where $Q(\rho) = v\rho$, is an approximation of the typical behavior of *free-flow* traffic, whereas the right side ($Q(\rho) = w(\rho_J - \rho)$) is associated with *congested* traffic.

Three different types of intercell connection are allowed: simple connection, merge, and diverge. In a simple connection, two cells are connected to one another without any intervening on-ramps or off-ramps (for example, cells 2 and 3 in Fig. 17.1(a)) . Let $i-1$ be the upstream cell and i be the downstream cell in the pair. As described in [4], $q_i(k)$, the flow entering cell i from the mainline, is determined by taking the minimum of two quantities:

$$q_i(k) = \min\Big(S_{i-1}(k), R_i(k)\Big), \tag{17.2}$$

$$S_{i-1}(k) = \min\Big(v_{i-1}\rho_{i-1}(k), Q_{M,i-1}\Big), \tag{17.3}$$

$$R_i(k) = \min\Big(Q_{M,i}, w_i\big(\rho_{J,i} - \rho_i(k)\big)\Big), \tag{17.4}$$

where $S_{i-1}(k)$ is the maximum flow that can be *supplied* by cell $i-1$ under free-flow conditions, over the k^{th} time interval, and $R_i(k)$ is the maximum flow that can be *received* by cell i under congested conditions, over the same time interval.

In the form presented here, the CTM also uses density-based versions of the merge and diverge laws of [4] to incorporate on-ramp and off-ramp flows; a complete statement of these merge and diverge laws can be found in [7]. A merge and diverge are shown within the context of a freeway segment in Fig. 17.1(a), where q_2 and r are the flows merging into cell 2 from cells 1 and OR, and q_4 and f are the flows diverging into cells 4 and FR from cell 3. The diverging flows are defined as $q_4(k) = (1-\beta(k))q_{3,out}(k)$, and $f(k) = \beta(k)q_{3,out}(k)$, where $\beta(k)$ is the *split ratio* for the diverge junction, i.e., the fraction of vehicles leaving cell 3 which exits to cell FR during the k^{th} time interval. It is assumed here that the split ratios can be determined externally to the model as functions of time.

The CTM is subject to the same intercell connectivity restrictions as those described in [4], such as limiting the maximum number of separate flow streams entering any cell to 2. Another requirement of the CTM is that the cell lengths must be longer than the free-flow travel distance, i.e.,

$$v_i T_s \leq l_i, \tag{17.5}$$

for cell i. The necessity of this condition for convergence of CTM solutions to LWR model solutions is explained in [19]. The CTM consists of flow conservation, Eq. (17.1), for each cell, along with the flow relations, Eqs. (17.2)–(17.4) and the merge and diverge laws. The state vector is $\boldsymbol{\rho} = [\rho_1 \dots \rho_N]^T$ for a freeway partitioned into N cells.

17.2.2 Switching-Mode Model

In order to gain additional insight into freeway traffic behavior, and to simplify the control analysis, control design, and data-estimation design methods, a piecewise-linearized version of the CTM, called the switching-mode model (SMM), has been designed [5]. Since the SMM is composed of several linear models, straightforward linear techniques for model analysis and control design can be applied to the individual linear subsystems.

The SMM is a hybrid system that switches among five sets of linear difference equations, depending on the congestion status of the cells and the values of the mainline boundary data. Assuming the state vector is composed of the cell densities, $\boldsymbol{\rho} = [\rho_1 \dots \rho_N]^T$, the key difference between the CTM and the SMM is that, with respect to density, the former is nonlinear, whereas each mode of the latter is linear. The SMM can be extracted from the CTM by writing each inter-cellular flow as either an explicit function of cell density, or as a constant. For example, in the case of a segment without merges or diverges, each q_i would be replaced with $v\rho_{i-1}(k)$, $w(\rho_J - \rho_i(k))$, or Q_M. This explicit density dependence is achieved by supplying a set of logical rules that determine the congestion status of each cell, at every time step, based on measurements at the segment boundaries.

For simplicity, the following assumptions are made:

1. The densities and flows at the upstream and downstream segment boundaries, as well as flows on all the on-ramps, are measured.
2. There is at most one status transition (or wave front) in the freeway section. If both the upstream and downstream mainline boundaries are of the same status, i.e., both *free-flow* or both *congested*, it is assumed that all the mainline cells, 1 through N, have the same status, while if the two boundaries are of different status, there exists a single wave front in the segment, upstream of which all the cells have congested (free-flow) status, and downstream of which all cells have free-flow (congested) status.

The single-wave front assumption is an approximation that is expected to be acceptable for short freeway segments with only one on-ramp and off-ramp, such as the example later in this section. To more accurately deal with longer sections with many on- and off-ramps, the switching logic can be modified to allow multiple wave fronts within a segment.

Since an SMM-modeled section contains at most one congestion wave front, the modes of the SMM can be distinguished by the congestion status of the cells upstream and downstream of the wave front. If there is no wave front in the section, a repeated label, e.g., "Free-flow–Free-flow", can be used to indicate the absence of any status transition. The five modes are denoted: (1) "Free-flow–Free-flow" (FF), (2) "Congestion–Congestion" (CC), (3) "Congestion–Free-flow" (CF), (4) "Free-flow–Congestion 1" (FC1), and (5) "Free-flow–Congestion 2" (FC2). The two modes of "Free-flow–Congestion" are determined by the relative magnitudes of the *supplied* flow of the last uncongested cell upstream of the wave front and the *received* flow of the first congested cell downstream of the wave front. If the former is smaller, the SMM is in FC1, while if the latter is smaller, it is in FC2. Respectively, these two cases are distinguished by whether the congestion wave is traveling forward or backward within the segment.

Consider the freeway segment in Fig. 17.1 (a), which contains 4 mainline cells. The on- and off-ramps will not be modeled as cells in this case. The measured aggregate flows and densities at the upstream and downstream mainline detectors are denoted by q_u, ρ_u, and q_d, ρ_d. All five modes of the SMM can be summarized as follows:

$$\boldsymbol{\rho}(k+1) = A_s\, \boldsymbol{\rho}(k) + B_s\, \boldsymbol{u}(k) + B_{J,s}\, \boldsymbol{\rho}_J + B_{Q,s}\, \boldsymbol{q}_M, \tag{17.6}$$

where $s = 1, 2, 3, 4, 5$ indicates the mode (1: FF, 2: CC, 3: CF, 4: FC1, 5: FC2), $\boldsymbol{\rho} = [\rho_1 \ldots \rho_4]^T$ is the state, and $\boldsymbol{u} = [q_u \;\; r_2 \;\; \rho_d]^T$ are the flow and density inputs; specifically, r_2 is the measured on-ramp flow entering the section, subscripted according to its cell of entry. $\boldsymbol{\rho}_J = [\rho_{J1}\ \rho_{J2}\ \rho_{J3}\ \rho_{J4}\ \rho_{J5}]^T$ is the vector of jam densities, and $\boldsymbol{q}_M = [Q_{M1}\ Q_{M2}\ Q_{M3}\ Q_{M4}]^T$ is the vector of maximum flow rates. Eq. (17.6) can alternatively be written with the downstream flow, q_d, as an input instead of ρ_d, but the stated form has the advantage of eliminating the $B_{J,s}\ \boldsymbol{\rho}_J$ term for modes with downstream congestion, in the case where w_i and $\rho_{J,i}$ are the same for each cell.

In FF mode, the flow across each cell boundary is dictated by *upstream* conditions; specifically, each cell releases traffic at the free-flow rate according to the first term in Eq. (17.3). That is, the total flow exiting cell i is given by $v_i\rho_i$. The flow across the upstream boundary of cell 1 is q_u. In the case of time-varying split ratios, the FF-mode state matrices (for the segment of Fig. 17.1(a)) are

$$A_1(k) = \begin{bmatrix} 1 - \frac{v_1 T_s}{l_1} & 0 & 0 & 0 \\ \frac{v_1 T_s}{l_2} & 1 - \frac{v_2 T_s}{l_2} & 0 & 0 \\ 0 & \frac{v_2 T_s}{l_3} & 1 - \frac{v_3 T_s}{l_3} & 0 \\ 0 & 0 & (1-\beta(k))\frac{v_3 T_s}{l_4} & 1 - \frac{v_4 T_s}{l_4} \end{bmatrix}, \quad B_1 = \begin{bmatrix} \frac{T_s}{l_1} & 0 & 0 \\ 0 & \frac{T_s}{l_2} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$
$$B_{J,1} = 0_{4\times 5}, \quad B_{Q,1} = 0_{4\times 4}. \tag{17.7}$$

For the CC mode, the flow across each cell boundary is dictated by *downstream* conditions; specifically, each cell absorbs flow according to the second term in Eq. (17.4). That is, the total flow entering cell i is given by $w_i(\rho_{J,i} - \rho_i)$. The flow released by cell 4 is determined by the downstream density ρ_d. The split-ratio-dependent matrices are

$$A_2(k) = \begin{bmatrix} 1 - \frac{w_1 T_s}{l_1} & \frac{w_2 T_s}{l_1} & 0 & 0 \\ 0 & 1 - \frac{w_2 T_s}{l_2} & \frac{w_3 T_s}{l_2} & 0 \\ 0 & 0 & 1 - \frac{w_3 T_s}{l_3} & \frac{1}{1-\beta(k)}\frac{w_4 T_s}{l_3} \\ 0 & 0 & 0 & 1 - \frac{w_4 T_s}{l_4} \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 & \frac{T_s}{l_1} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{w_5 T_s}{l_4} \end{bmatrix},$$
$$B_{J,2}(k) = \begin{bmatrix} \frac{w_1 T_s}{l_1} & -\frac{w_2 T_s}{l_1} & 0 & 0 & 0 \\ 0 & \frac{w_2 T_s}{l_2} & -\frac{w_3 T_s}{l_2} & 0 & 0 \\ 0 & 0 & \frac{w_3 T_s}{l_3} & -\frac{1}{1-\beta(k)}\frac{w_4 T_s}{l_3} & 0 \\ 0 & 0 & 0 & \frac{w_4 T_s}{l_4} & -\frac{w_5 T_s}{l_4} \end{bmatrix},$$
$$B_{Q,2} = 0_{4\times 4}. \tag{17.8}$$

Since the FF and CC modes are of primary importance in the estimator and controller designs of Sections 17.4 and 17.5, a discussion of the mixed modes (CF, FC1, and FC2) is omitted here. For a complete description of the mixed modes, please see [5, 7]. A set of switching rules were developed [5] to determine the mode of the system at each time step, based on the measured mainline boundary data and the congestion status of the cells in the section. If both ρ_u and ρ_d have free-flow status (for a triangular fundamental diagram, this means they are below the critical density ρ_c for that region), the FF mode is selected, and if both of these densities are congested, the CC mode is selected. If ρ_u and ρ_d are of opposite status, then the SMM performs a search over the ρ_i to determine whether there is a status transition inside the section. This wave front search consists of searching through the cells, in order, looking for the first status transition between adjacent cells. The CTM and SMM have been validated for an approximately 2-mi section of I-210W, where they were shown to yield estimation errors of about 13% for density and 4% for flow [5, 7].

17.2.3 Observability and controllability of the SMM

Observability and controllability properties have been derived for the SMM modes, using standard techniques for time-varying linear systems. The middle column of Table 17.1 summarizes the observability for each SMM mode, assuming that mainline detectors are only available at the upstream and downstream segment boundaries, as in Fig. 17.1(a). On the left side, “upst. cells” and “downst. cells” give the status of cells both upstream and downstream of the congestion wave front. If there is no such wave front, both sets of cells have the same status. The middle column indicates which of the two mainline boundary measurements, if either, can be used to make the mode observable. These results can be obtained by computing the observability grammians for the $A_s(k)$ with the output matrices $C_u = [1\ 0\ 0\ 0]$ and $C_d = [0\ 0\ 0\ 1]$.

From the table, it can be seen, as a general result, that if all cells have free-flow status, the densities are observable using a downstream measurement, while in congested mode, they are observable using an upstream measurement. This is related to the wave (information) propagation directions on a freeway in different congestion modes. When a freeway section is in free-flow mode, the information propagates downstream at speed v, which is the vehicle traveling speed. Therefore, in order to be able to estimate the cell densities, the downstream density measurement is needed. When the freeway is in congestion, the information propagates upstream at speed w, which is the backward congestion wave traveling speed, and an upstream measurement is needed to estimate densities.

Table 17.1. Observability and controllability for different SMM modes (“OR” indicates on-ramp).

Upst. Cells	Downst. Cells	Observable with	Controllable using
Free-flow	Free-flow	Downst. Measurement	Upst. On-Ramp
Congested	Congested	Upst. Measurement	Downst. On-Ramp
Congested	Free-flow	Upst. & Downst. Meas.	Not Controllable
Free-flow	Congested 1	Unobservable	Upst. & Downst. OR
Free-flow	Congested 2	Unobservable	Upst. & Downst. OR

Controllability can be analyzed analogously, and the results are summarized in the last column of Table 17.1. Generally, a section in free-flow mode is controllable from an on-ramp at its upstream end, whereas a congested section can be controlled from an on-ramp at its downstream end. The observability and controllability of the mixed modes (CF, FC1, and FC2), along with derivations of the results for all modes, are discussed in more detail in [7].

17.3 Calibration of the Cell Transmission and Switching Mode Model Parameters to the I-210 Testbed

In this section, a methodology for tuning the CTM and SMM parameters to reproduce observed freeway traffic behavior is described. The calibration method has been tested on a 14-mile stretch of Interstate 210 Westbound (I-210W) in Pasadena, California, which is shown in the map in Fig. 17.2, and typically endures heavy congestion during the weekday morning commute period.

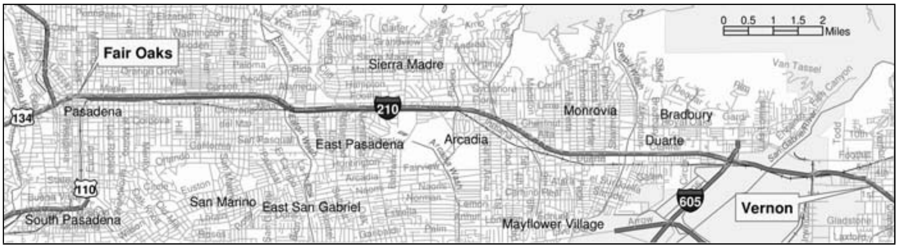


Fig. 17.2. A map of the I-210W testbed. Composed using the U.S. Census Bureau 2004 TIGER/Line®data.

17.3.1 Freeway representation

The 14-mile I-210W test segment has been divided into 41 cells, as shown in Fig. 17.3. This partition was adapted from a 40-cell partition that was designed by Gabriel Gomes for use in the optimization work of [20, 21]. The traffic flow direction is in order of increasing cell index, i.e., left to right, starting at the top of the figure. The cell index is located in the center of each cell. The uppermost row of numbers above the cells is the cell length (in feet). The second row of numbers gives the number of mixed-flow lanes (4 to 6) in each cell. Vertical gray bars mark the locations of the mainline loop detectors, and the postmile of the detector (e.g., 39.159) is listed above the detector marker. On- and off-ramps are depicted as numbered arrows. Associated street names are given for each set of ramps. A single high-occupancy vehicle (HOV) lane runs parallel to the leftmost mainline lane on this segment of I-210W. Each of the six HOV-lane gates is indicated by a horizontal gray bar, and an additional fictitious on-ramp (no. 22) was used to approximate the flow of vehicles entering the mixed-flow lanes from the Lake Ave. HOV gate.

In the default partitioning method, cell boundaries are placed on the mainline immediately upstream of on-ramps and immediately downstream of off-ramps. However, for the chosen model time step of 10 sec., and a typical free flow speed of 63 mph, three of the cells were found to be shorter than the minimum allowed cell length of 924 ft; to satisfy this constraint, several adjustments (described in [6, 7]) were made to the asterisk-marked cells.

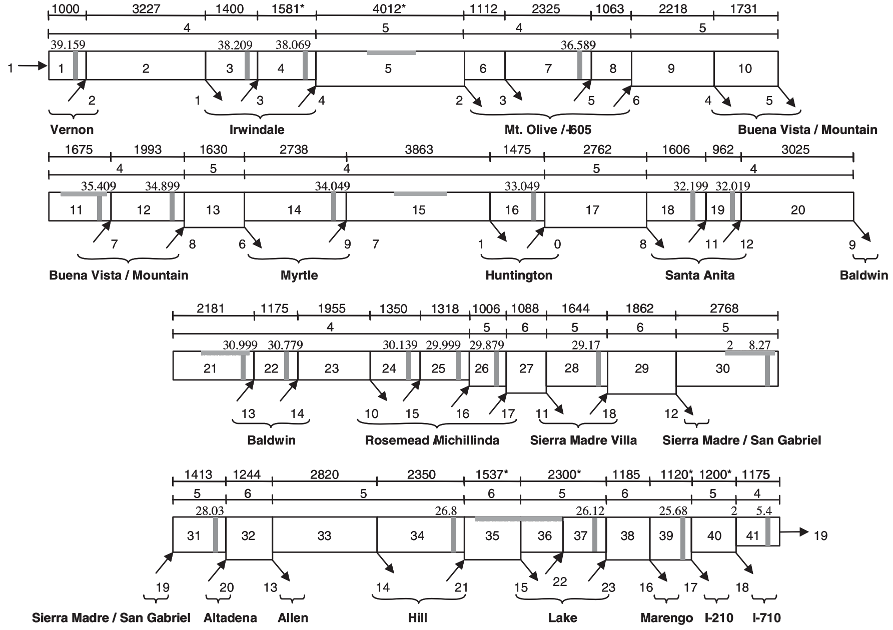


Fig. 17.3. 41-cell partition of I-210W testbed. Adapted from a 40-cell partition developed by Gabriel Gomes for use in optimization studies related to [20, 21].

Traffic data used in the model calibration was mostly obtained from the PeMS website [22], developed by Varaiya and his associates. In some few instances, where PeMS data was incomplete, demands were reconstructed using a set of manually-counted I-210W ramp flows provided by Caltrans. The reader is referred to [6, 7] for details on data processing, demand reconstruction, split ratio estimation, HOV modeling, and applicable simplified cases of the merge and diverge laws.

17.3.2 Calibration methodology

The main steps of the calibration procedure are as follows:

1. Free-flow Parameter Calibration: The free-flow traffic velocities, v_i, are determined by performing a least-squares fit on the flow versus density data over the period 5:00–6:00AM. For the I-210 section, traffic typically flows freely during this period. For the j^{th} detector, v_j is the solution, in the least-squares sense, to the equation $\Phi_j v_j = Y_j$, where Φ_j and Y_j are column vectors that respectively contain densities and flows measured over the specified time interval. The free-flow speed v_j is assigned to the cell containing detector j, and free-flow speeds are computed for non-detector cells by linear interpolation.

2. Bottleneck Identification: Bottleneck locations are identified by examining contour plots of the measured traffic densities and/or speeds, and determining the locations of fixed spatial boundaries which divide the freeway into an upstream congested region and a downstream free-flow region. For example, in the top plot of Fig. 17.4, a bottleneck was observed to form between the detectors at 33.049 and 32.199 during the 6:00 time slice.

3. Non-Bottleneck Capacity Selection: A set of nominal $Q_{M,i}$ are assigned to the cells that are not located at bottlenecks. It is not advisable to set $Q_{M,j}$ equal to the maximum observed flow at each detector-equipped cell, since this will most likely result in underestimating the true capacity of the freeway. Typically, the nominal $Q_{M,i}$ must be chosen to be larger than the maximum observed flows (usually $\geq$ 2000 veh/hr per lane (vphpl)) in each region of the freeway.

4. Bottleneck Capacity Determination: Consider a freeway portion divided into two consecutive cells, 1 and 2, where an active bottleneck exists between the two cells, hence the upstream cell is congested, while the downstream cell remains in free-flow status. Further assume that an on-ramp (with merging flow r_2 entering cell 2) exists between the two cells. It can be shown that the bottleneck capacity in this situation is represented by $Q_{M,2} = q_2 + r_2$, where q_2 is the flow entering cell 2 from the mainline [6, 7]. Since both q_2 and r_2 are measurable, these quantities are used to estimate the bottleneck flow rate, with the default method (assuming no faulty or missing data) being $\hat{Q}_{M,2} = \text{mean}_{k \in K_M}(q_2(k) + r_2(k))$. K_M corresponds to the half-hour time interval ending at $\arg\max(q_2(k) + r_2(k))$.

5. Congestion Parameter Calibration: First, the critical density is estimated for each detector; $\hat{\rho}_{c,j} = \max_k(q_{d_j}(k))/v_j$, where q_{d_j} is the flow measured at

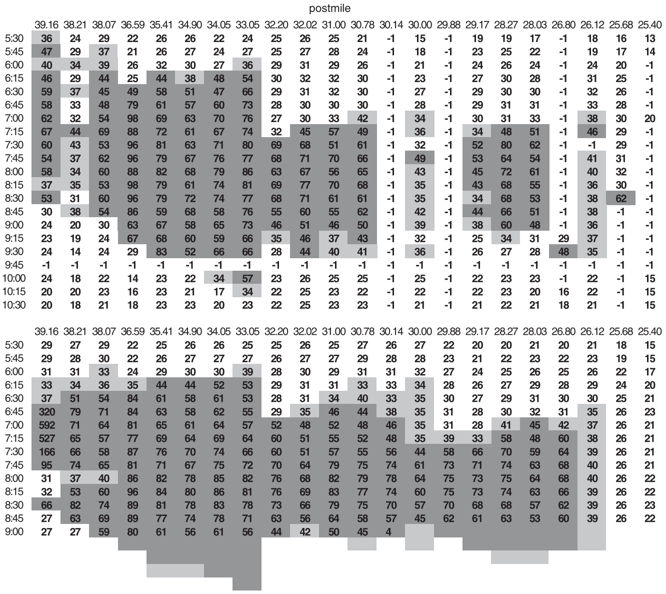
detector j. Then, the flow and density measurements are sorted so that only congested pairs are used in the parameter estimation. The congested-mode equations can be rewritten so they are linear in $[w_j \ w_j \rho_{J,j}]^T$. By substituting in the congested $(\rho_{d_j}(k), q_{d_j}(k))$ measurements, and applying the constraint $Q_{M,j} \leq \frac{v_j \ w_j \ \rho_{J,j}}{v_j + w_j}$ to preserve the maximum flow rates determined in previous steps, the estimated parameters are derived from the $[w_j \ w_j \rho_{J,j}]^T$ that solves the constrained least-squares problem. Additional details on constraints and interpolation methods are provided in [6, 7].

6. Time-Varying Parameter Adjustments: If necessary, temporary parameter changes (e.g., reduction of $Q_{M,i}$ in a region) can be applied to reproduce the effect of an incident. Also, by reducing w_i in the mid-morning time range, when the traffic is still congested but beginning its recovery back to the free-flow mode, the effect of flow-density hysteresis can be approximated.

17.3.3 Results and discussion

Fig. 17.4 shows contour plots for the measured (top) and simulated CTM (bottom) densities for a particular day (Wednesday, Nov. 28, 2001) in the I-210 testbed.

The numbers inside the shaded cells are traffic densities, in vehicles per mile per lane (vpmpl). Free-flow densities (0–33 vpmpl) are shown as white. Mid-range congestion (33–43 vpmpl) is medium gray. Dark gray indicates heavy congestion (43 vpmpl or greater). Traffic is flowing from left to right in these plots, and the



postmile

	39.16	38.21	38.07	36.59	35.41	34.90	34.05	33.05	32.20	32.02	31.00	30.78	30.14	30.00	29.88	29.17	28.27	28.03	26.80	26.12	25.68	25.40
5:30	36	24	29	22	26	26	22	24	25	26	25	21	-1	15	-1	19	19	17	-1	18	16	13
5:45	47	29	37	21	26	27	24	27	25	27	28	24	-1	18	-1	23	25	22	-1	19	17	14
6:00	40	34	39	26	32	30	27	36	29	31	29	26	-1	21	-1	24	26	24	-1	24	20	-1
6:15	46	29	44	25	44	38	48	54	30	32	32	30	-1	23	-1	27	30	28	-1	31	25	-1
6:30	59	37	45	49	58	51	47	66	29	31	32	30	-1	27	-1	29	30	30	-1	32	26	-1
6:45	58	33	48	79	61	57	60	73	28	30	30	30	-1	28	-1	29	31	31	-1	33	28	-1
7:00	62	32	54	98	69	63	70	76	27	30	33	42	-1	34	-1	30	31	33	-1	38	30	20
7:15	67	44	69	88	72	61	67	74	32	45	57	49	-1	36	-1	34	48	51	-1	46	29	-1
7:30	60	43	53	96	81	63	71	80	69	68	51	61	-1	32	-1	52	80	62	-1	-1	29	-1
7:45	54	37	62	96	79	67	76	77	68	71	70	66	-1	49	-1	53	64	54	-1	41	31	-1
8:00	58	34	60	88	82	68	79	86	63	67	56	65	-1	43	-1	45	72	61	-1	40	32	-1
8:15	37	35	53	98	79	61	74	81	69	77	70	68	-1	35	-1	43	68	55	-1	36	30	-1
8:30	53	31	60	96	79	72	74	77	68	71	61	61	-1	35	-1	34	68	53	-1	38	62	-1
8:45	30	38	54	86	59	68	58	76	55	60	55	62	-1	42	-1	44	66	51	-1	38	-1	-1
9:00	24	20	30	63	67	58	65	73	46	51	46	50	-1	39	-1	38	60	48	-1	36	-1	-1
9:15	23	19	24	67	68	60	59	66	35	46	37	43	-1	32	-1	25	34	31	29	37	-1	-1
9:30	24	14	24	29	83	52	66	66	28	44	40	41	-1	36	-1	26	27	28	48	35	-1	-1
9:45	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
10:00	24	18	22	14	23	22	34	57	23	26	25	25	-1	25	-1	22	23	23	-1	22	-1	15
10:15	20	20	23	16	23	21	17	34	22	25	23	22	-1	22	-1	22	23	20	16	22	-1	15
10:30	20	18	21	18	23	23	20	23	22	25	23	23	-1	21	-1	21	22	21	18	21	-1	15

	39.16	38.21	38.07	36.59	35.41	34.90	34.05	33.05	32.20	32.02	31.00	30.78	30.14	30.00	29.88	29.17	28.27	28.03	26.80	26.12	25.68	25.40
5:30	29	27	29	22	25	26	26	25	25	26	25	27	26	27	22	20	20	21	20	21	18	15
5:45	29	28	30	22	26	27	27	27	26	27	27	29	28	28	23	21	22	23	22	23	19	15
6:00	31	31	33	24	29	30	30	39	28	30	29	31	31	32	27	24	25	26	25	26	22	17
6:15	33	34	36	35	44	44	52	53	29	31	31	33	33	34	28	26	27	29	28	29	24	20
6:30	37	51	54	84	61	58	61	53	28	31	34	40	33	35	30	27	29	31	30	30	25	21
6:45	320	79	71	84	63	58	62	55	29	35	46	44	38	35	31	28	30	32	31	35	26	23
7:00	592	71	64	81	65	61	64	57	52	48	52	48	46	35	31	28	41	45	42	37	26	21
7:15	527	65	57	77	69	64	69	64	60	51	55	52	48	35	39	33	58	48	60	38	26	21
7:30	166	66	58	87	76	70	74	66	60	51	57	55	56	44	58	66	70	59	64	39	26	21
7:45	95	74	65	81	71	67	75	72	70	64	79	75	74	61	73	71	74	63	68	40	26	21
8:00	31	37	40	86	82	78	85	82	76	68	82	79	78	64	75	73	75	64	68	40	26	22
8:15	32	53	60	96	84	80	86	81	76	69	83	77	74	60	75	73	74	63	66	39	26	22
8:30	66	82	74	89	81	78	83	78	73	66	79	75	70	57	70	68	68	57	62	39	26	23
8:45	27	63	69	89	77	74	78	71	63	56	64	58	57	45	62	61	63	53	60	39	26	22
9:00	27	27	59	80	61	56	61	56	44	42	50	45	4									

Fig. 17.4. Contour plots of 15-minute average measured densities (top) and simulated CTM densities (bottom), vpmpl, for Nov. 28, 2001.

time, in 15-minute intervals, is given in the leftmost column. The time range is 5:30–10:30AM. Loop detector outages are indicated by crossed-out boxes in the measured-data contour plot. Loop detectors which were suspected to be faulty for the whole day have their postmile labels surrounded by a dashed box at the top of the measured-data plot. If a detector was classified as faulty due to outages in some, but not all, of the lanes,the corresponding "measured" density displayed in the contour plot is a scaling-reconstructed estimate. Details regarding the values and consistency of the estimated parameters obtained with this method can be found in [6, 7].

To evaluate the performance of the simulation, we define a partial Total Travel Time: $TTT = T_s \sum_{k=k_{5:00}}^{k_{11:45}} \sum_{i \in C_d} l_i \rho_i(k)$. Here, C_d is the set of cells which had problem-free mainline detectors over each of the examined days. C_d excludes detectors at postmiles 38.209, 38.069, 34.049, 30.779, 30.139, 29.999, 29.879, 28.030, 26.800, and 25.400. Although it functions properly, the detector at 39.159 is also excluded, since the CTM boundary condition prevents the model from accurately reproducing congestion that (in the real system) spills upstream outside of the simulated region. Results for TTT are summarized in Table 17.2, along with the spatial mean of the mean percentage error (MMPE) at the non-excluded detectors, defined as $E_{MMPE} = 100 \times \frac{1}{N_{C_d}} \sum_{i \in C_d} \frac{1}{M} \sum_{k=1}^{M} \left| \frac{\rho_i(k) - \hat{\rho}_i(k)}{\rho_i(k)} \right|$, where N_{C_d} is the number of non-excluded detectors (11 out of 22 in this case). The resulting values of MMPE are not surprising, since they are similar to the MPEs in the short-segment tests of [5].

Table 17.2. Total Travel Time (veh-hr) and mean MPE for three different days.

Date	Meas.	Sim.	% TTT Err.	MMPE
Jan. 10, 2002	3778	3766	-0.32	14.2
Nov. 28, 2001	4273	4278	0.01	14.2
Nov. 13, 2001	4163	3961	-4.85	16.1
mean	4071	4002	-1.72	14.8
std. dev.	260	258	2.71	1.1

From Table 17.2, it can be seen that the CTM reproduces the observed bottlenecks and the approximate duration and spatial extent of the congestion upstream of each bottleneck, and predicts the total travel time with approximately 2% mean error over three days. Simulation tests documented in [7] indicate that TTT is more sensitive to Q_M than other model parameters.

17.4 Traffic State Estimation

In order to effectively control the on-ramp flows, traffic state information, such as vehicle density and the presence or absence of nearby congestion, has to be made available to the ramp metering controller. However, cost and other limitations prevent sensory devices being installed and maintained at all desired locations. Therefore, these traffic states must be estimated using the available data. In this section we describe a traffic state estimator based on the switching-mode model (SMM), described in Section 17.2.2, and the mixture Kalman filter (MKF) [10], that is capable of estimating the vehicle densities at unmeasured locations, as well as determining the traffic congestion mode in a freeway section.

17.4.1 Improved mixture Kalman filter

In the switching-mode model (SMM) described in Section 17.2.2, there were five possible congestion modes. In the remaining sections of this chapter, we further simplify this model by considering only two of the five modes, i.e., purely *free-flow* and fully *congested*, that are most important, while neglecting all other mixed cases, such as the mode wherein half the cells in a section are in free-flow and half are in congestion. This selection of modes is motivated by the observation that short freeway sections tend to spend most of their time in either a free-flow or congested condition, with mixed modes being transient.

The mode (*free-flow* or *congested*) is determined by the flow condition in the section. However, there is no direct measurement or observation of the current traffic congestion mode in a freeway section. The congestion mode can only be inferred from measured quantities, such as traffic speed. The general practice in traffic engineering is to set an upper threshold and a lower threshold for the speed. When the speed in a section is above the upper threshold, the section is considered to be in free flow; when the speed is below the lower threshold, the section is in congestion; when the speed is between the two thresholds, the section is considered somewhat likely to be in congestion. The problem with this kind of method is two-fold: 1) The selection of the thresholds is based on experience and, to a certain degree, is arbitrary, and 2) When the speed falls between the two thresholds, the mode of the section cannot be determined.

Therefore, we assume that we do not have direct observation of the mode and that the mode jumps between possible values following a discrete-time Markov chain with a certain transition probability. Under these assumptions, the switching-mode model falls into a special class, called *Markov jump linear systems (MJLS)*. If only the FF and CC modes of Section 17.2.2 are considered, the previous four-cell example has a continuous state $x = [\rho_1\ \rho_2\ \rho_3\ \rho_4]^T$ and possible discrete modes 1 (*free-flow*) and 2 (*congested*).

It is known that it is difficult to estimate the states and the mode when the mode itself is not observed. The difficulty lies in the fact that the sample space S^t of the mode sequence grows exponentially as time t increases, where S is the set of possible discrete modes.

The *mixture Kalman filter (MKF)* [10] approximately solves this difficult probability inference problem by employing a Monte Carlo approach that approximates the exponentially growing sample space by a fixed finite number, M, of mode sample sequences $\boldsymbol{s}_t^{(m)}$, where $m = 1, \ldots, M$. A weight is associated with each of the sample sequences to represent the *a posteriori* probability of that sample sequence,

$$\xi_t^{(m)} := \frac{p\left(\boldsymbol{s}_t^{(m)} \mid \boldsymbol{y}_t, \boldsymbol{u}_t\right)}{\sum_{m=1}^{M} p\left(\boldsymbol{s}_t^{(m)} \mid \boldsymbol{y}_t, \boldsymbol{u}_t\right)}, \tag{17.9}$$

where a symbol in boldface, for example, $\boldsymbol{s}_t$, represents a sequence from time 0 to time t and y_t and u_t respectively denote the output measurement and control input. After the new measurement y_{t+1} is available, these weights are updated by

$$\xi_{t+1}^{(m)} = \frac{\xi_t^{(m)} \zeta_{t+1}^{(m)}}{\sum_{m=1}^{M} \xi_t^{(m)} \zeta_{t+1}^{(m)}}, \tag{17.10}$$

where the incremental weight

$$\zeta_{t+1}^{(m)} := p\left(y_{t+1} \mid \boldsymbol{y}_t, \boldsymbol{u}_t, \boldsymbol{s}_t^{(m)}\right), \tag{17.11}$$

represents the likelihood of the new measurement for a given mode sample sequence.

On each of these mode sample sequences, a (time-varying) Kalman filter is implemented to estimate the continuous states. The state estimates on all mode sample sequences are then "mixed" (averaged) by the weights, and this weighted average approximates the *a posteriori* estimate of the continuous states, i.e.,

$$\hat{x}_{t|t} = \sum_{m=1}^{M} \xi_t^{(m)} \hat{x}_{t|t}^{(m)}, \tag{17.12}$$

where $\hat{x}_{t|t}^{(m)}$ is the *a posteriori* state estimate from the m^{th} Kalman filter.

The accuracy of this Monte Carlo method is improved by a predictive sampling technique, in which the current mode is sampled according to a predictive probability

$$\mu_{t+1}^{(m)}(s) \propto p\left(s_{t+1}^{(m)} = s, y_{t+1} \mid \boldsymbol{s}_t^{(m)}, \boldsymbol{y}_t, \boldsymbol{u}_{t+1}\right), \tag{17.13}$$

which favors the mode with higher likelihood given the current measurements and the previously sampled modes.

The weight update procedure is recursive. The entire history from time 0 influences the current weight. It is often found in implementation that most of these weights approach 0 while only a few remain of modest magnitudes. This phenomenon reduces the effective number of sample sequences and introduces an underflow risk for the weights when implemented on a machine with finite floating point precision. Therefore, a forgetting and weight underflow prevention scheme [8] has been introduced in our implementation. In this scheme, the weights of the sample sequences are bounded from below,

$$\xi_{t+1}^{(m)} = \max\left\{\xi_{t+1}^{(m)}, \underline{\xi}\right\}, \tag{17.14}$$

and are re-normalized after the bounding step. This simple procedure not only prevents the underflow, but in effect limits the influence of the early history on the weights and makes the weights recover more quickly when their corresponding sample sequences are favored by the current measurements.

In addition to the mixture estimate of the continuous states, the mixture Kalman filter also provides an approximate maximum *a posteriori* (MAP) estimation of the congestion mode:

$$\hat{s}_{t,\text{MAP}} = \arg\max_s p\left(s_t = s \mid \boldsymbol{y}_t, \boldsymbol{u}_t\right), \tag{17.15}$$

where

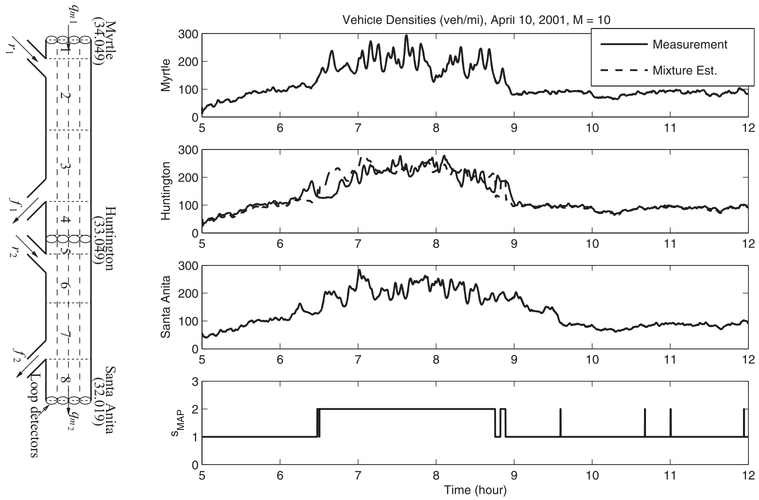
$$p\left(s_t = s \mid \boldsymbol{y}_t, \boldsymbol{u}_t\right) \approx \sum_{m=1}^{M} \xi_t^{(m)} 1_s\left(s_t^{(m)}\right), \tag{17.16}$$

and $1_s(s_t)$ is the indicator function. This is particularly important to our application because different control schemes will be used in different congestion modes (*free-flow* or *congested*), based on the freeway density controllability properties [5].

17.4.2 Results and discussion

The mixture Kalman filter based congestion mode and vehicle density estimation algorithm was first tested on a two-mile long section of I-210W, from Myrtle to Santa Anita, as shown in Fig. 17.5(a). The flow and density measurements at the Myrtle and Santa Anita stations were used as feedback to the estimator, while the measurements at the Huntington station were assumed to be unavailable and were used to evaluate the estimation accuracy. Fig. 17.5(b) shows the estimation results using data from

April 10, 2001, with the number of sample sequences $M = 10$. The first three plots show the vehicle density estimation results. In these plots, the solid lines are the measured vehicle densities, while the dashed lines are the mixture estimates. The last plot shows the MAP congestion mode estimates, where 1 indicates *free-flow* mode and 2 indicates *congested* mode. It can be seen from the plots that the estimator was able to accurately estimate the density at the Huntington station, which was not available to the estimator. The mean percentage errors (MPEs), as defined by $E_{\mathrm{MPE}} = \frac{1}{T+1}\sum_{t=0}^{T}\left|\frac{\hat{x}_{t|t}-x_t}{x_t}\right|$, were 9.48% for the mixture estimate in this particular experiment. A complete list of estimation errors with different estimator settings and using data from different days can be found in [8]. On average, this estimator achieves an MPE of $\sim$10%.



(a) Test site.

(b) Estimation results.

Fig. 17.5. Test results of the congestion mode and vehicle density estimator.

This estimator was also implemented for the entire 14-mile long test segment [9]. The program was written in the C language for reasons of efficiency and portability. Not only can the estimator run on collected traffic data sets, but it has also been successfully interfaced with a calibrated VISSIM microscopic traffic simulator [11], through the VISSIM DDE (Direct Data Exchange) interface, and with a calibrated macroscopic cell transmission model [6]. The estimator runs synchronously with these traffic simulators. The running time of the estimator with 10 mode sample sequences for a 7-hour (from 5AM to 12 noon) time period and the full 14-mile segment is less than one minute on a 1.4 GHz Pentium M computer.

The traffic data was extracted from the PeMS [22] database in 2002 and 2003. The traffic flows and vehicle occupancies are available every 30 seconds, while the speeds and g-factors (estimated average vehicle lengths) are available every 5 minutes. We interpolated the 30-second flow and density data into 2-second intervals and passed the interpolated data through a low-pass filter to reduce the amount of noise in the original 30-second traffic data. The estimator produced the vehicle density for each cell, as well as the congestion mode for each section, using this 2-second interpolated and filtered data.

Fig. 17.6(b) shows an example of the MAP congestion mode estimation from the estimator. In this example, data from January 10, 2002 was used. In the plot, light gray indicates *free-flow* mode, while dark gray indicates *congested* mode.

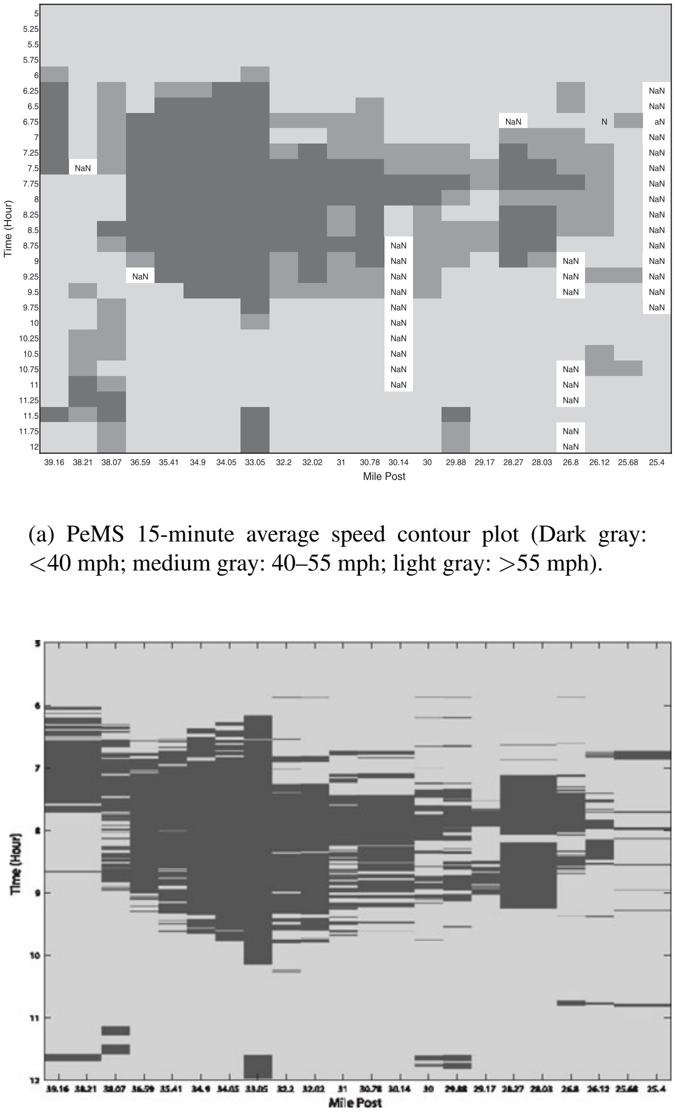
For comparison, a contour plot of the PeMS-derived 15-minute average speeds for that day is given in Fig. 17.6(a). In this plot, light gray indicates an average speed of 55 miles per hour and above, which is generally considered to indicate free-flow conditions in traffic engineering. Dark gray indicates an average speed of 40 miles per hour and below, in which case the traffic is considered to be in congestion. Medium gray indicates an average speed between 40 and 55 miles per hour; in this range, the traffic is somewhat likely to be in congestion. In the plot, white indicates unavailable data.

It can be seen from the plot that in general, the congestion mode estimation by the MKF-based estimator agrees with the speed contour plot. However, the MKF-based congestion mode estimation is preferable for the following reasons.

1. As mentioned earlier, the thresholds for the speed are determined empirically and can vary from location to location.
2. It is not clear whether a section is in congestion or not when the speed is between the upper and lower thresholds.
3. The speed data usually are not available as frequently as the density and flow data when the data are collected using single loop detectors, which is usually the case.
4. More importantly, the MKF-based estimation provides the statistically most probable mode that directly corresponds to one of the possible dynamic models, while the speed-based estimate itself does not have this direct correspondence.
5. The MKF-based estimator also provides vehicle density estimation for all the cells where no measurements are available.

17.5 Ramp-Metering Control

The goal of an on-ramp control system is to improve the efficiency of a freeway by regulating the number of vehicles that are allowed to enter through the on-ramps, in order to delay the onset of congestion and minimize its duration, and consequently maximize off-ramp exit flows, while preventing the on-ramp queues from spilling over and producing congestion in arterial routes. In this section, we briefly review a localized and traffic responsive ramp-metering strategy that we have developed [13, 14], which chooses the less restrictive ramp metering rate among the following two control systems: a control system that monitors and regulates the freeway mainline traffic density in the neighborhood of a ramp merge, and a control



(a) PeMS 15-minute average speed contour plot (Dark gray: <40 mph; medium gray: 40–55 mph; light gray: >55 mph).

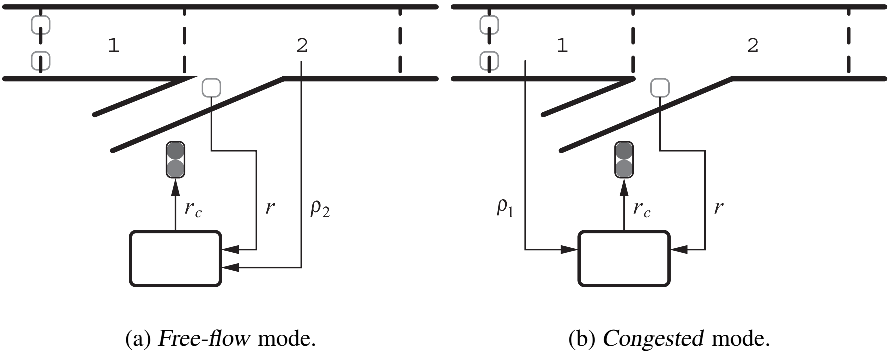
(b) MKF maximum *a posteriori* estimation (Dark gray: *congested*; light gray: *free-flow*).

Fig. 17.6. Congestion mode estimation for the test segment of Interstate 210 Westbound in Pasadena, California (January 10, 2002).

system that estimates and regulates the on-ramp queue length, preventing it from spilling over into the arterial traffic. This structure was first proposed by Smaragdis and Papageorgiou [16].

17.5.1 Switching ramp-metering mainline density control

As discussed in Sections 17.2.2 and 17.4.1, the traffic dynamics in a freeway section are different under different congestion conditions—free-flow or congested. Under free-flow conditions, unmeasured mainline densities in a section can only be estimated using a downstream measurement, while on-ramp flows affect downstream mainline traffic densities. On the other hand, under congested conditions, unmeasured mainline densities in a section can only be estimated using an upstream measurement, while on-ramp flows affect upstream mainline traffic densities. It is therefore advantageous to change the structure of the localized controller to suit the current controllability and observability properties of the mainline segment near the on-ramp merge, as illustrated in Fig. 17.7. Moreover, since most California freeways only have mainline loop detectors located upstream of the on-ramp, as depicted in Fig. 17.7, mainline densities in sections that are downstream of the on-ramp, such as the mainline density ρ_2 in Fig. 17.7(a), must be estimated. The mixture Kalman filter (MKF) based traffic state estimator that we have developed [8, 9] is used to estimate, in real time, the most probable congestion mode and the cell vehicle densities in a freeway section. The estimated congestion mode is used to determine the appropriate control structure, and the estimated vehicle densities are used as feedback.



(a) *Free-flow* mode. (b) *Congested* mode.

Fig. 17.7. Different control structures for different congestion modes.

To compensate for disturbances and to accommodate the difference between the model sampling time and the metering-rate update interval, a multirate linear quadratic control with integral action (multirate LQI) approach [13] was used to synthesize the ramp-metering controller for both of the congestion modes.

In either mode, the desired metering rate is first calculated using

$$r_c(t) = r(t-1) - K(t) \begin{bmatrix} \tilde{\rho} \\ z \end{bmatrix} (t), \tag{17.17}$$

and then saturated using

$$r_c(t) = \min\{r_{\max}, \max\{r_{\min}, r_c(t)\}\}, \tag{17.18}$$

where $r(t)$ is the actual ramp flow measured by the entrance loop-detector, as shown in Fig. 17.8, $\tilde{\rho}(t)$ is the mainline density error ($\rho - \bar{\rho}$ for a desired density $\bar{\rho}(t)$), $z(t) = \tilde{\rho}(t) - \tilde{\rho}(t-1)$, and $r_{\max}$ and $r_{\min}$ are the established maximum and minimum metering rates. There is an anti-windup scheme in (17.17) that is similar to what is used in ALINEA [12] to address the metering-rate saturation problem.

In (17.17),

$$K(t) = \begin{cases} K_p, & \text{when } t = np \text{ for some } n \in \mathbb{Z}, \\ 0, & \text{when } t \neq np \text{ for any } n \in \mathbb{Z}, \end{cases} \tag{17.19}$$

where p is the ratio between the metering-rate update interval and the model sampling time, and K_p is determined by solving a periodic Riccatti equation. See [13] for details.

17.5.2 Queue length regulation

A typical configuration of loop detectors and signals on an on-ramp on a California freeway is shown in Fig. 17.8. To prevent the on-ramp queue from spilling over into surface streets and interfering with the street traffic, the queue length must be regulated. The "queue-override" scheme currently used on California freeways steadily increases the metering rate (e.g., 120 vehicles per hour per lane every 30 seconds) whenever the end of the queue reaches the queue detector, until the metering rate saturates to the maximum value. After the queue dissipates and recedes behind the queue detector location, the metering rate is reset to the value determined by the mainline traffic responsive metering controller. This scheme is equivalent to an integral control with a saturated integrating rate and resetting. It can be easily shown that the resulting closed loop dynamics is not asymptotically stable, given that the open loop queue length dynamics is that of a simple integrator. It has been noted [15, 16] that this queue-override scheme leads to oscillatory behavior and under-utilization of on-ramp storage capacities. In [16], Smaragdis and Papageorgiou proposed a proportional controller that relies on the on-ramp vehicle demands. However, real-time demand measurements are generally not available in the field, and such a control scheme would have to instead rely on historical demands.

If the queue length $l(t)$ could be measured, an asymptotically stable PI-controller

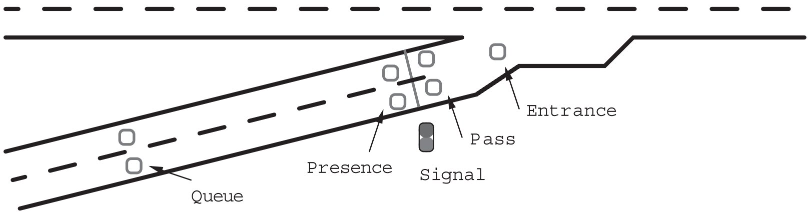


Fig. 17.8. A typical configuration of loop detectors and signals on an on-ramp.

$$r_c(z) = \left(k_P + \frac{k_I}{z-1}\right)\tilde{l}(z) \tag{17.20}$$

would be able to regulate the queue length precisely at a specified value. This controller can be designed by choosing proper gains k_P and k_I, using the root-locus method on the closed-loop sensitivity function from the disturbance to the error, which is given by

$$\frac{\tilde{l}(z)}{d(z)} = \frac{T_s(z-1)}{(z-1)^2 - k_P T_s z + (k_I - k_P)T_s}, \tag{17.21}$$

where $\tilde{l}(t)$ is the queue length error, and $d(t)$ is the vehicle arrival rate (the demand), which is regarded as a disturbance.

The anti-windup and saturating mechanisms in (17.17) and (17.18) also need to be implemented in this queue length regulator.

17.5.3 On-ramp queue length estimation

Though it has a more stable response than the queue-override scheme, the PI regulator described in Section 17.5.2 needs the current queue length as its feedback, which unfortunately is not presently available in the field. A suitable estimator has to be designed using available information, such as the speed of the vehicles entering the on-ramp, as measured by the queue detector [14].

We assume the following simplified driving behavior model for a vehicle approaching the end of the queue: the vehicle decelerates at a constant rate, $-a$, from its cruising speed to a target speed v_0, which is achieved at the position where the distance from the end of the queue is s. We also assume a uniform effective vehicle length g. Let l_0 be the number of vehicle spaces from the stop line to the queue detector and $v(t)$ be the vehicle speed measured by the queue detector, as depicted in Fig. 17.9.

A straightforward kinematic calculation yields

$$g\big(l_0 - l(t)\big) - s = \frac{v(t)^2 - v_0^2}{2a}, \tag{17.22}$$

where $l(t)$ is the current queue length, in number of vehicles. From (17.22), we obtain

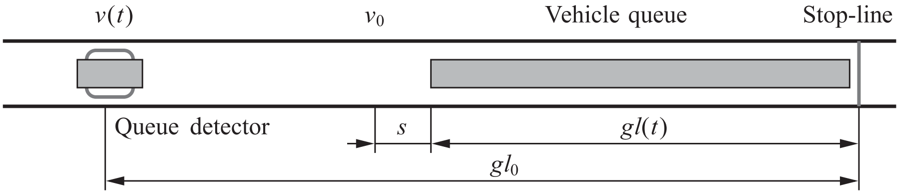


Fig. 17.9. A schematic for on-ramp queue length estimation.

$$gl(t) = gl_0 - s + \frac{v_0^2}{2a} - \frac{v(t)^2}{2a} = c_0 - c_2 v(t)^2. \tag{17.23}$$

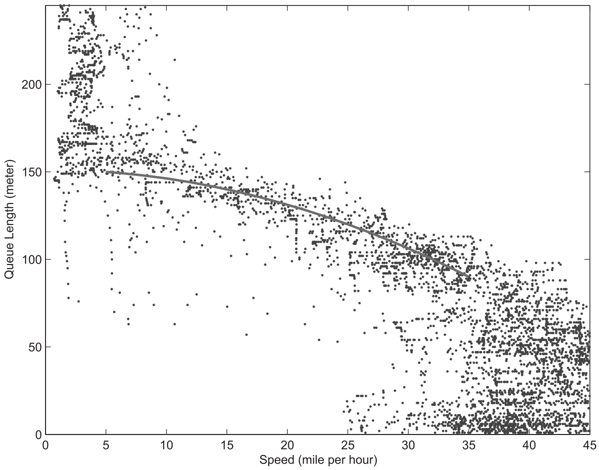


Fig. 17.10. A scatter plot of queue lengths vs. queue detector speeds and the least median-of-squares curve fit for one of the on-ramps.

To determine the coefficients c_0 and c_2 in (17.23), a curve fitting was performed on the $gl(t)$ and $v(t)$ data collected using the VISSIM [23] microscopic traffic simulator. Fig. 17.10 shows a typical scatter plot of queue lengths versus speeds. A few points need to be noted:

1. When the queue is shorter than a certain length, the approaching vehicles pass the queue detector at the drivers' desired cruising speeds, which are independent of the queue length. This phenomenon corresponds to the data points at the lower-right corner of the scatter plot.
2. When the queue is longer than gl_0, i.e., the queue has extended beyond the queue detector, the measured speed is also a constant, which is related to the queue discharging rate and the vehicle lengths, and is also independent of the queue length. This phenomenon corresponds to the data points at the upper-left corner of the scatter plot.

3. There are many outliers among the data points. Therefore, the usual least-squares curve fitting method, which is biased toward outliers, is not suitable.

For these reasons, we neglected the data points whose speeds are below $v_{\min}$ or above $v_{\max}$ and those whose queue lengths are below $l_{\min}$ or above $l_{\max}$ in the curve fitting. These values were determined by visual inspection of the scatter plots.

To increase robustness to outliers, we used the least median-of-squares [24] curve fitting method, instead of the usual least (sum-of-)squares. The fitted curve is also shown in Fig. 17.10.

After the l–v curve is fitted for each on-ramp, the difference between the actual and desired queue length, which is used as the feedback to the regulator (17.20), is estimated as

$$\tilde{l}(t) = \begin{cases} \big(c_0 - gl_0 - c_2 v(t)^2\big)/g, & \text{if } v(t) \geq v_{\min}, \\ -kc_2\big(v(t)^2 - v_{\min}^2\big)/g, & \text{if } v(t) < v_{\min}, \end{cases} \tag{17.24}$$

where k is a tuning parameter.

When $v(t) < v_{\min}$, the end of the queue is very close to or beyond the queue detector, and the speed $v(t)$ measured by the queue detector is a constant, which is roughly gr_c. Therefore, (17.24) can be thought of as a method for saturating $\tilde{l}$ to $-kc_2\big((gr_c)^2 - v_{\min}^2\big)/g$, which is larger when the metering rate r_c is lower. This has a desirable effect on the regulator: The metering rate r_c will be increased more aggressively when there is more room for this increase, and more slowly when r_c is close to its maximum value. In addition, this saturation value can be further tuned by changing the value of k.

It is also worth mentioning that the coefficients c_0 and c_2 identified by the least median-of-squares fitting are very close to the nominal values predicted by using the actual distance between the stop-line and the queue detector and a nominal vehicle deceleration of $2.5\ \mathrm{m/s^2}$. Therefore, when the queue length measurements are unavailable through any means to perform a curve fitting, these nominal values can be used in the queue length estimation.

17.5.4 A localized and traffic responsive on-ramp metering control strategy

Localized traffic responsive strategies are desired for reasons including reduced algorithmic complexity, lower computational requirements, and higher robustness to changing traffic conditions such as unpredicted demands. We have proposed a localized metering strategy and tested it on a calibrated macroscopic traffic model [13]. It is described as follows:

1. The set-point for the switching mainline traffic responsive ramp-metering controller is chosen to be the critical density, i.e., the density at which congestion is about to form. This is adopted to slow down congestion shock waves propagating in the upstream direction and to speed up congestion shock waves moving downstream.

2. The set-point of the queue length regulator is the maximum allowed queue length. This value is chosen to fully utilize the available storage capacity on the ramp and to deter short-trip travelers from using the freeway, thus making the freeway capacity available to longer-distance travelers.
3. The *higher* of the two rates determined by the mainline traffic responsive metering controller and the queue length regulator is chosen to be the actual metering rate that is sent to the signal control box. This rule, which was first proposed by Smaragdis and Papageorgiou [16], is designed to properly resolve the conflict between the objectives of these two controllers.

17.5.5 Performance measures

Before presenting results, we first define some performance measures for quantitative evaluation of a given freeway segment. All the quantities are defined for the time period T and the freeway segment L.

$D_{\mathrm{V,tot}}$ Total Vehicle Distance, which is defined as the sum of the distances traveled by all the vehicles in L within T.

$T_{\mathrm{V,tot}}$ Total Vehicle Time, which is the sum of the time that is spent by all vehicles in L within T. It includes the time spent by vehicles waiting in the on-ramp queues.

$DL_{\mathrm{V,tot}}$ Total Vehicle Delay (also known as Congestion Delay), which is the difference between the Total Vehicle Time and the time that would be spent by all the vehicles if there were no congestion. $DL_{\mathrm{V,tot}} = T_{\mathrm{V,tot}} - D_{\mathrm{V,tot}}/v_0$, where v_0 is the nominal free-flow speed.

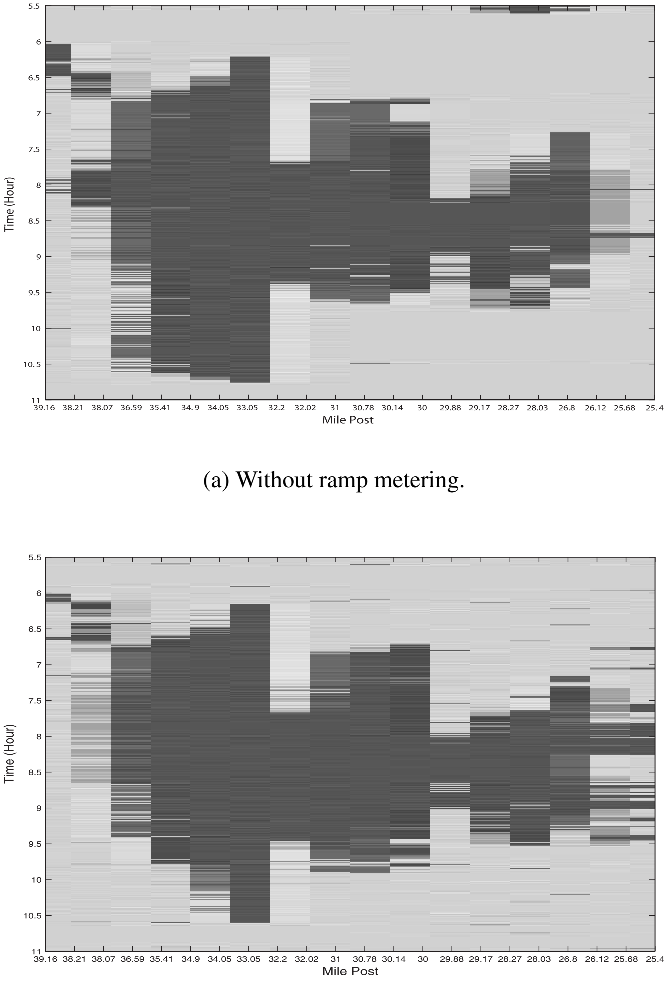
$\bar{v}_{\mathrm{V,tot}}$ Average Total Vehicle Speed $\bar{v}_{\mathrm{V,tot}} = D_{\mathrm{V,tot}}/T_{\mathrm{V,tot}}$.

$\bar{v}_{\mathrm{V,ml}}$ Average Mainline Vehicle Speed, which is similar to $\bar{v}_{\mathrm{V,tot}}$ but calculated using only data from the mainline portion of the freeway.

Another set of passenger-weighted performance measures can be defined by first computing the traffic quantities separately for the low- or high-occupancy vehicle classes, and then, during the performance-measure calculation, weighting these quantities by the average passenger number in each vehicle class. This set of passenger-weighted performance measures includes Total Passenger Distance $D_{\mathrm{P,tot}}$, Total Passenger Time $T_{\mathrm{P,tot}}$, Total Passenger Delay $DL_{\mathrm{P,tot}}$, Average Total Passenger Speed $\bar{v}_{\mathrm{P,tot}}$, and Average Mainline Passenger Speed $\bar{v}_{\mathrm{P,ml}}$.

17.5.6 Results

The switching mainline traffic responsive metering controller and the queue length regulator were implemented and interfaced with the VISSIM microscopic traffic simulation model that has been calibrated to the I-210W test segment in [11]. The localized control strategy described in Section 17.5.4 was used. Fig. 17.11 shows the congestion patterns, as determined by the MKF traffic state estimator [8, 9], before and after ramp metering. In the plots, dark gray indicates *congested* mode and light gray *free-flow*. The vertical axis is time, from 5:30 to 11:00 in the morning. The horizontal axis is the mile post along the freeway, and the traffic travels from left to right. It can be seen that the localized ramp metering strategy was able to reduce the congestion, in terms of both the spatial span and the time duration.



(a) Without ramp metering.

(b) With ramp metering.

Fig. 17.11. Congestion modes for the I-210W test segment under different metering scenarios, light gray: *free-flow*, and dark gray: *congested*.

For these tests, the parameters in the multirate LQI design were set as follows: $Q = I$, $R = 5$ for the *free-flow* mode, and $Q = I$, $R = 20$ for the *congested* mode, with gains $k_P = k_I = 120$ in the queue length regulator.

We also implemented a modified version of ALINEA [12] and combined it with the queue length estimator and regulator that we have presented in Sections 17.5.3 and 17.5.2. In this modified ALINEA, we used the occupancy data measured upstream of the on-ramps, instead of using occupancies measured downstream of the on-ramps (as originally recommended for ALINEA), in order to better reflect the loop-detector configuration typically available on California freeways. In [25] it has been shown, using the calibrated I-210W VISSIM model, that this modified ALINEA can achieve comparable and sometimes even better performance, when compared to the original ALINEA. The optimal ALINEA gain (7000) and set-point (27.2%) found in [25] were used in our simulations.

Different ramp-metering algorithms, including 1) switching LQI plus queue regulation, 2) switching LQI only, without queue regulation, and 3) ALINEA plus queue regulation, were tested with the I-210W VISSIM model. Under each scenario, 8 simulation runs were carried out, with 8 different VISSIM random seeds. The random seed was chosen to be the second of the computer clock at the time when it was changed, to ensure its randomness.

Some of the performance measures for this freeway segment, as defined in Section 17.5.5, are listed in Table 17.3. The listed numbers are the averages from the 8 simulation runs for each scenario. In calculating these quantities, the average passenger numbers per one low- and high-occupancy vehicle are assumed to be 1.2 and 2.5, respectively, and the nominal free-flow speed v_0 is 63 miles per hour.

It should be noted that the I-605 interchange into I-210 cannot be metered and provides a large volume of inflow traffic into the upstream portion of I-210W (on-ramp number 6 in Fig. 17.3). Consequently, congestion takes place on I-210W, even when all metered on-ramp queues are allowed to overflow into arterial streets.

Under all the scenarios, the freeway segment served almost the same amount of demand, as measured by the Total Vehicle Distance $D_{\mathrm{V,tot}}$ or Total Passenger Distance $D_{\mathrm{P,tot}}$. Ramp-metering was able to reduce the congestion under all the metered scenarios. For example, with the switching LQI mainline control and queue length regulation, the Total Vehicle Delay (also known as Congestion Delay) $DL_{\mathrm{V,tot}}$ was reduced by 16%, while with the switching LQI mainline control only, $DL_{\mathrm{V,tot}}$ was reduced by 20%.

When only the switching mainline traffic responsive metering was used, without activating the queue length regulator, on-ramp queues could grow to arbitrary lengths, sometimes hundreds of vehicles. In this case, almost all the congestion on the mainline was eliminated, as evidenced by the average mainline vehicle speed $\bar{v}_{\mathrm{V,ml}}$, which was 55.8 mph. Another interesting phenomenon in this case is that the relative improvements in terms of passenger-weighted performance measures were greater than those in terms of vehicle performance measures. This is because many of the metered on-ramps on this freeway segment have designated lanes for HOVs to bypass the long queues.

Table 17.3. Performance measures for the I-210 test segment under different ramp-metering algorithms. Q/R means queue estimation and regulation.

	$D_{\mathrm{V,tot}}$ (10^3 mile)	$T_{\mathrm{V,tot}}$ (10^3 hour)	$DL_{\mathrm{V,tot}}$ (10^3 hour)	$\bar{v}_{\mathrm{V,tot}}$ (mph)	$\bar{v}_{\mathrm{V,ml}}$ (mph)
No metering	973	24.0	8.52	40.6	37.8
Switching + Q/R	972	22.6	7.19	43.0	40.8
Improvement	–	5.6%	16%	5.8%	7.9%
Switching	974	22.3	6.81	43.7	55.8
Improvement	–	7.1%	20%	7.7%	47%
ALINEA + Q/R	974	23.5	8.01	41.5	39.2
Improvement	–	2.0%	5.8%	2.2%	3.6%
	$D_{\mathrm{P,tot}}$ (10^6 mile)	$T_{\mathrm{P,tot}}$ (10^3 hour)	$DL_{\mathrm{P,tot}}$ (10^3 hour)	$\bar{v}_{\mathrm{P,tot}}$ (mph)	$\bar{v}_{\mathrm{P,ml}}$ (mph)
No metering	1.32	31.7	10.8	41.6	39.4
Switching + Q/R	1.32	29.9	9.0	44.0	42.4
Improvement	–	5.5%	16%	5.7%	7.4%
Switching	1.32	29.3	8.3	45.1	56.4
Improvement	–	7.7%	23%	8.4%	43%
ALINEA + Q/R	1.32	31.0	10.1	42.5	40.7
Improvement	–	2.1%	6.3%	2.3%	3.4%

It can also be seen from the numbers in Table 17.3 that the switching control algorithm outperforms the modified ALINEA, when both algorithms are combined with the queue length estimator and regulator.

17.6 Conclusions

In this chapter, we first presented a macroscopic freeway traffic model, the Switching-Mode Model (SMM), which is a piecewise-linearized version of Daganzo's CTM [3, 4]. The SMM is computationally efficient and well-suited for implementation in real-time control, estimation, and traffic monitoring applications. The observability and controllability properties of the individual modes of the SMM, which are of fundamental importance in the design of data estimators and ramp-metering control systems, were stated. It was revealed that the free-flow traffic mode is observable from a downstream measurement and controllable from an upstream on-ramp, and that the congested mode is observable from an upstream measurement and controllable from a downstream on-ramp.

A procedure for calibrating the CTM and SMM parameters was summarized. A calibrated CTM model was tested on the full 14-mile test section of I-210W, and has been shown to reproduce the main features of the observed traffic congestion on the freeway, such as approximate location of bottlenecks and duration and spatial extent of congestion. In addition, the model accurately predicts the total travel time (TTT) in the freeway. A main benefit of the overall calibration method is that it provides a well-defined, automatable procedure for using loop detector data to estimate free-flow speeds, congestion parameters, and bottleneck capacities for the CTM.

A congestion mode and vehicle density estimator was designed and implemented on the I-210W test segment. Using the mixture Kalman filtering (MKF) algorithm on the switching-mode traffic model, the estimator is able to provide the estimated vehicle densities at unmeasured locations, as well as the most probable traffic congestion modes (*free-flow* or *congested*), which are not directly observed. The test results on a short freeway section show that on average, a mean percentage error of about 10% in density estimation is achieved and the performance is consistent over different days. The algorithm approximately maintains its performance even with a relatively small number of sample sequences and runs efficiently, thus making it possible to carry out estimation in real time. The availability of the congestion modes enables us to design more effective ramp metering algorithms, utilizing the appropriate switching-mode model dynamics under different flow conditions.

We also presented a localized ramp-metering strategy that achieves the control goal of reducing the spatial and temporal extent of the congestion, using locally available information. This control strategy works with a switching mainline traffic responsive ramp-metering controller that adapts to the different traffic dynamics under different congestion conditions, and a PI queue length regulator that yields improved performance over the currently used “queue-override” scheme and keeps the queue under the ramp storage capacity limit. In addition, a queue length estimator was designed to provide feedback to the queue length regulator, using the queue-detector speed data that are available in the field. Test results on the calibrated VISSIM I-210W microscopic model demonstrated the performance and effectiveness of the switching ramp-metering controller, the queue length estimator and regulator, and the overall control strategy. The Total Vehicle and Passenger Delays were both reduced by 16%, while the Total Vehicle Time and the Total Average Vehicle Speed were improved by 5.6% and 5.8%. As a comparison, simulation results of ALINEA were also presented. The switching mainline traffic responsive control was able to outperform ALINEA, when both algorithms were combined with the same queue length estimator and regulator.

References

[1] D. Schrank and T. Lomax, *The 2004 Urban Mobility Report*, Tech. rep., Texas Transportation Institute, 2004, URL `http://mobility.tamu.edu/`

[2] Cambridge Systematics, Inc. and Texas Transportation Institute *Traffic Congestion and Reliability: Linking Solutions to Problems*,

Tech. rep., Federal Highway Administration, U.S. Department of Transportation, 2004, URL `http://www.ops.fhwa.dot.gov/congestion_report/`

[3] C. F. Daganzo, The Cell Transmission Model: A Dynamic Representation of Highway Traffic Consistent with the Hydrodynamic Theory, *Transportation Research - B*, **28**(4):269–287, 1994.

[4] C. F. Daganzo, The Cell Transmission Model, Part II: Network Traffic, *Transportation Research - B*, **29**(2):79–93, 1995.

[5] L. Muñoz, X. Sun, R. Horowitz, and L. Alvarez, Traffic density estimation with the cell transmission model, In: *Proceedings of the 2003 American Control Conference*, Denver, Colorado, USA, 3750–3755, 2003.

[6] L. Muñoz, X. Sun, D. Sun, G. Gomes, and R. Horowitz, Methodological calibration of the cell transmission model, In: *Proceedings of the 2004 American Control Conference*, Boston, Massachusetts, USA, 798–803, 2004.

[7] L. M. Muñoz, Macroscopic modeling and identification of freeway traffic flow, Ph.D. Dissertation, University of California, Berkeley, 2004.

[8] X. Sun, L. Muñoz, and R. Horowitz, Highway traffic state estimation using improved mixture Kalman filters for effective ramp metering control, In: *Proceedings of the 42nd IEEE Conference on Decision and Control*, Maui, Hawaii, USA, 6333–6338, 2003.

[9] X. Sun, L. Muñoz, and R. Horowitz, Mixture Kalman filter based highway congestion mode and vehicle density estimator and its application, In: *Proceedings of the 2004 American Control Conference*, Boston, Massachusetts, USA, 2098–2103, 2004.

[10] R. Chen and J. S. Liu, Mixture Kalman filters, *Journal of the Royal Statistical Society, Series B—Statistical Methodology*, **62**:493–508, 2000.

[11] G. Gomes, A. D. May, and R. Horowitz, Calibration of VISSIM for a congested highway, In: *The 83rd Annual Meeting of the Transportation Research Board*, Washington, D.C., USA, 2004.

[12] M. Papageorgiou, H. Hadj-Salem, and J.-M. Blosseville, ALINEA: A local feedback control law for on-ramp metering, *Transportation Research Record*, (1320):58–64, 1991.

[13] X. Sun and R. Horowitz, A localized switching ramp-metering controller with a queue length regulator for congested freeways, In: *Proceedings of the 2005 American Control Conference*, Portland, Oregon, USA, 2005, to appear.

[14] X. Sun and R. Horowitz, Localized switching ramp-metering control with queue length estimation and regulation and microscopic simulation results, In: *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, 2005, to appear.

[15] R. L. Gordon, Algorithm for controlling spillback from ramp meters, *Transportation Research Record* (1554):162–171, 1996.

[16] E. Smaragdis and M. Papageorgiou, Series of new local ramp metering strategies, *Transportation Research Record* (1856):74–86, 2004.

[17] M. J. Lighthill and G. B. Whitham On kinematic waves, 2: A theory of traffic flow on long crowded roads. *Proceedings of the Royal Society of London, Series A—Mathematical and Physical Sciences* **229**(1178):317–345, 1955.

[18] P. I. Richards Shock-waves on the highway, *Operations Research*, **4**(1):42–51, 1956.

[19] C. F. Daganzo, A Finite Difference Approximation of the Kinematic Wave Model of Traffic Flow, *Transportation Research - B* **29**(4):261–276, 1995.

[20] G. Gomes and R. Horowitz, Globally optimal solutions to the onramp metering problem, part I, In: *Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems*, Washington, D.C., USA, 509–514, 2004.

[21] G. Gomes and R. Horowitz, Globally optimal solutions to the onramp metering problem, part II, In: *Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems*, Washington, D.C., USA, 515–520, 2004.

[22] PeMS Team, The freeway performance measurement system, Web Site, URL `http://pems.eecs.berkeley.edu/`, version 5.4, 2005.

[23] PTV AG, 2004, VISSIM. Web page, URL `http://www.english.ptv.de/cgi-bin/traffic/traf_vissim.pl`

[24] P. J. Rousseeuw, Least median of squares regression, *Journal of the American Statistical Association*, **79**(388):871–880, 1984.

[25] G. C. Gomes, Optimization and microsimulation of on-ramp metering for congested freeways, Ph.D. dissertation, University of California, Berkeley, 2004.