

LNCS 4056

Paola Flocchini
Leszek Gąsieniec (Eds.)

Structural Information and Communication Complexity

13th International Colloquium, SIROCCO 2006
Chester, UK, July 2006
Proceedings



Springer

TP274-53

S927
2006

Paola Flocchini Leszek Gąsieniec (Eds.)

Structural Information and Communication Complexity

13th International Colloquium, SIROCCO 2006
Chester, UK, July 2-5, 2006
Proceedings



Springer



E200603654

Volume Editors

Paola Flocchini
University of Ottawa
School of Information Technology and Engineering (SITE)
Ottawa, ON, K1N 6N5, Canada
E-mail: flocchin@site.uottawa.ca

Leszek Gąsieniec
The University of Liverpool
Department of Computer Science
Liverpool, L69 7ZF, UK
E-mail: leszek@csc.liv.ac.uk

Library of Congress Control Number: 2006927796

CR Subject Classification (1998): F.2, C.2, G.2, E.1

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

ISSN 0302-9743
ISBN-10 3-540-35474-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-35474-1 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media

springer.com

© Springer-Verlag Berlin Heidelberg 2006
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11780823 06/3142 5 4 3 2 1 0

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Friedemann Mattern

ETH Zurich, Switzerland

John C. Mitchell

Stanford University, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

Oscar Nierstrasz

University of Bern, Switzerland

C. Pandu Rangan

Indian Institute of Technology, Madras, India

Bernhard Steffen

University of Dortmund, Germany

Madhu Sudan

Massachusetts Institute of Technology, MA, USA

Demetri Terzopoulos

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Moshe Y. Vardi

Rice University, Houston, TX, USA

Gerhard Weikum

Max-Planck Institute of Computer Science, Saarbruecken, Germany

Preface

The Colloquium on Structural Information and Communication Complexity (SIROCCO) is an annual meeting focused on the relationship between algorithmic aspects of computing and communication. Over its 13 years of existence, SIROCCO has become an acknowledged forum bringing together specialists interested in the fundamental principles underlying the interplay between information, communication and computing. SIROCCO covers topics such as distributed computing, high-speed networks, interconnection networks, mobile computing, optical computing, parallel computing, sensor networks, wireless networks, autonomous robots, and related areas.

SIROCCO 2006 was the 13th in this series, held in Chester, UK, July 3–5, 2006. Previous SIROCCO colloquia took place in Ottawa (1994), Olympia (1995), Siena (1996), Ascona (1997), Amalfi (1998), Lacanau-Océan (1999), L'Aquila (2000), Val de Nuria (2001), Andros (2002), Umeå (2003), Smolenice Castle (2004), and Mont Saint-Michel (2005).

In the tradition of previous occasions, this year's SIROCCO was a lively event, encouraging the emergence of new research areas (related to distributed computing in a broad sense) and the dissemination of original ideas. This was achieved by dedicating ample time for informal discussions and open problem sessions in addition to regular conference activities.

The 68 contributions submitted to SIROCCO 2006 were subject to a thorough refereeing process and 24 high-quality submissions were selected for publication. We would like to thank the authors of all the submitted papers. The excellent quality of the final program is also due to the dedicated and careful work of the Program Committee members. Our gratitude extends to the numerous subreferees for their valuable help.

We also thank the invited speakers: Hagit Attiya (Technion), Danny Krizanc (Wesleyan), and Roger Wattenhofer (ETH) for accepting our invitation to share their insights on new developments in their areas of interest, and for providing such entertaining talks.

We would like to express our sincere gratitude to the conference Chair David Peleg (Weizmann) for his enthusiasm and invaluable consultations, as well as the local organizing team, in particular Christoph Ambühl, Catherine Atherton, Alexey Fishkin, Dave Shield and Prudence Wong. Finally we would like to thank EPSRC and the University of Liverpool for their support.

These proceedings include all the accepted papers revised according to the feedback provided by the Program Committee, as well as the paper versions of the three invited talks. We hope you will enjoy them as much as we did at the conference.

July 2006

Paola Flocchini and Leszek Gąsieniec
Program Committee Co-chairs

Organization

Program Committee

Paolo Boldi, Milan	Andrzej Lingas, Lund
Shlomi Dolev, Ben-Gurion	Flaminia Luccio, Trieste
Thomas Erlebach, Leicester	Andrzej Pelc, Québec
Paola Flocchini, Ottawa (Co-chair)	Giuseppe Prencipe, Pisa
Fedor Fomin, Bergen	Kirk Pruhs, Pittsburgh
Pierre Fraigniaud, Paris Sud	Tomasz Radzik, Kings College
Leszek Gaśieniec, Liverpool (Co-chair)	Violet R. Syrotiuk, Arizona State
Christos Kaklamanis, Patras	Roger Wattenhofer, ETH
Idit Keidar, Technion	Masafumi Yamashita, Kyushu
Ralf Klasing, Bordeaux	Shmuel Zaks, Technion
Evangelos Kranakis, Carleton	

Organizing Committee

Christoph Ambühl, Liverpool	Leszek Gaśieniec, Liverpool (Chair)
Catherine Atherton, Liverpool	Dave Shield, Liverpool
Alexei Fishkin, Liverpool	Prudence Wong, Liverpool

Steering Committee

Paola Flocchini, Ottawa	Bernard Mans, Macquarie
Leszek Gaśieniec, Liverpool	Andrzej Pelc, Québec
Christos Kaklamanis, Patras	David Peleg, Weizmann (Chair)
Lefteris Kirovsi, Patras	Michel Raynal, Rennes
Rastislav Kráľovič, Bratislava	Nicola Santoro, Ottawa
Evangelos Kranakis, Carleton	Paul Spirakis, Patras
Danny Krizanc, Wesleyan	

Referees

Luca Allulli	Amos Beimel
Christoph Ambühl	Sivan Bercovici
Gal Badishi	Edward Bortnikov
René Beier	Olga Brukman

VIII Organization

Nicolas Burri
Ioannis Caragiannis
Carlos Castillo
Stefano Chessa
David Coudert
Shantanu Das
Stefan Dobrev
Michael Elkin
Leah Epstein
Alexey Fishkin
Hen Fitoussi
Michele Flammini
Roland Flury
Vincenzo Gervasi
Olga Goussevskaia
Yinnon Haviv
Jesper Jansson
Panagiotis Kanellopoulos
Ronen Kat
Dariusz Kowalski
Rastislav Kralovic
Danny Krizanc
Michael Kuhn
Łukasz Kuszner
Christos Levcopoulos
Thomas Locher
Violetta Lonati
Zvi Lotker
Fabrizio Luccio

Euripides Markou
Morten Mjælde
Manuela Montangero
Luca Moscardelli
Thomas Moscibroda
Alfredo Navarra
Linda Pagli
Evi Papaioannou
Paolo Penna
Mia Persson
Alessandro Proveti
Geppino Pucci
Pascal von Rickenbach
Fabiano Sarracco
Stefan Schmid
Hadas Shachnai
Alexander Shraer
Francesco Silvestri
Savio Tse
Nir Tzachar
Chi-Hung Tzeng
Ugo Vaccaro
Sebastiano Vigna
Martin Wahlen
Mirjam Wattenhofer
Yves Weber
Prudence Wong
Qin Xin

Sponsoring Institutions

The Engineering and Physical Sciences Research Council (EPSRC)

Lecture Notes in Computer Science

For information about Vols. 1–3958

please contact your bookseller or Springer

- Vol. 4060: K. Futatsugi, J.-P. Jouannaud, J. Meseguer (Eds.), *Algebra, Meaning and Computation*. XXXVIII, 643 pages. 2006.
- Vol. 4058: L.M. Batten, R. Safavi-Naini (Eds.), *Information Security and Privacy*. XII, 446 pages. 2006.
- Vol. 4056: P. Flocchini, L. Gąsieniec (Eds.), *Structural Information and Communication Complexity*. X, 357 pages. 2006.
- Vol. 4055: J. Lee, J. Shim, S.-g. Lee, C. Bussler, S. Shim (Eds.), *Data Engineering Issues in E-Commerce and Services*. IX, 290 pages. 2006.
- Vol. 4054: A. Horváth, M. Telek (Eds.), *Formal Methods and Stochastic Models for Performance Evaluation*. VIII, 239 pages. 2006.
- Vol. 4053: M. Ikeda, K.D. Ashley, T.-W. Chan (Eds.), *Intelligent Tutoring Systems*. XXVI, 821 pages. 2006.
- Vol. 4044: P. Abrahamsson, M. Marchesi, G. Succi (Eds.), *Extreme Programming and Agile Processes in Software Engineering*. XII, 230 pages. 2006.
- Vol. 4043: A.S. Atzeni, A. Lioy (Eds.), *Public Key Infrastructure*. XI, 261 pages. 2006.
- Vol. 4041: S.-W. Cheng, C.K. Poon (Eds.), *Algorithmic Aspects in Information and Management*. XI, 395 pages. 2006.
- Vol. 4040: R. Reulke, U. Eckardt, B. Flach, U. Knauer, K. Polthier (Eds.), *Combinatorial Image Analysis*. XII, 482 pages. 2006.
- Vol. 4039: M. Morisio (Ed.), *Reuse of Off-the-Shelf Components*. XIII, 444 pages. 2006.
- Vol. 4038: P. Ciancarini, H. Wiklicky (Eds.), *Coordination Models and Languages*. VIII, 299 pages. 2006.
- Vol. 4037: R. Gorrieri, H. Wehrheim (Eds.), *Formal Methods for Open Object-Based Distributed Systems*. XVII, 474 pages. 2006.
- Vol. 4036: O. H. Ibarra, Z. Dang (Eds.), *Developments in Language Theory*. XII, 456 pages. 2006.
- Vol. 4034: J. Münch, M. Vierimaa (Eds.), *Product-Focused Software Process Improvement*. XVII, 474 pages. 2006.
- Vol. 4033: B. Stiller, P. Reichl, B. Tuffin (Eds.), *Performativity Has its Price*. X, 103 pages. 2006.
- Vol. 4031: M. Ali, R. Dapoigny (Eds.), *Innovations in Applied Artificial Intelligence*. XXIII, 1353 pages. 2006. (Sublibrary LNAI).
- Vol. 4027: H.L. Larsen, G. Pasi, D. Ortiz-Arroyo, T. Andreassen, H. Christiansen (Eds.), *Flexible Query Answering Systems*. XVIII, 714 pages. 2006. (Sublibrary LNAI).
- Vol. 4026: P. Gibbons, T. Abdelzaher, J. Aspnes, R. Rao (Eds.), *Distributed Computing in Sensor Systems*. XIV, 566 pages. 2006.
- Vol. 4025: F. Eliassen, A. Montresor (Eds.), *Distributed Applications and Interoperable Systems*. XI, 355 pages. 2006.
- Vol. 4024: S. Donatelli, P. S. Thiagarajan (Eds.), *Petri Nets and Other Models of Concurrency - ICATPN 2006*. XI, 441 pages. 2006.
- Vol. 4021: E. André, L. Dybkjær, W. Minker, H. Neumann, M. Weber (Eds.), *Perception and Interactive Technologies*. XI, 217 pages. 2006. (Sublibrary LNAI).
- Vol. 4020: A. Bredendfeld, A. Jacoff, I. Noda, Y. Takahashi, *RoboCup 2005: Robot Soccer World Cup IX*. XVII, 727 pages. 2006. (Sublibrary LNAI).
- Vol. 4018: V. Wade, H. Ashman, B. Smyth (Eds.), *Adaptive Hypermedia and Adaptive Web-Based Systems*. XVI, 474 pages. 2006.
- Vol. 4016: J.X. Yu, M. Kitsuregawa, H.V. Leong (Eds.), *Advances in Web-Age Information Management*. XVII, 606 pages. 2006.
- Vol. 4013: L. Lamontagne, M. Marchand (Eds.), *Advances in Artificial Intelligence*. XIII, 564 pages. 2006. (Sublibrary LNAI).
- Vol. 4012: T. Washio, A. Sakurai, K. Nakashima, H. Takeda, S. Tojo, M. Yokoo (Eds.), *New Frontiers in Artificial Intelligence*. XIII, 484 pages. 2006. (Sublibrary LNAI).
- Vol. 4011: Y. Sure, J. Domingue (Eds.), *The Semantic Web: Research and Applications*. XIX, 726 pages. 2006.
- Vol. 4010: S. Dunne, B. Stoddart (Eds.), *Unifying Theories of Programming*. VIII, 257 pages. 2006.
- Vol. 4009: M. Lewenstein, G. Valiente (Eds.), *Combinatorial Pattern Matching*. XII, 414 pages. 2006.
- Vol. 4007: C. Álvarez, M. Serna (Eds.), *Experimental Algorithms*. XI, 329 pages. 2006.
- Vol. 4006: L.M. Pinho, M. González Harbour (Eds.), *Reliable Software Technologies – Ada-Europe 2006*. XII, 241 pages. 2006.
- Vol. 4005: G. Lugosi, H.U. Simon (Eds.), *Learning Theory*. XI, 656 pages. 2006. (Sublibrary LNAI).
- Vol. 4004: S. Vaudenay (Ed.), *Advances in Cryptology – EUROCRYPT 2006*. XIV, 613 pages. 2006.
- Vol. 4003: Y. Koucheryavy, J. Harju, V.B. Iversen (Eds.), *Next Generation Teletraffic and Wired/Wireless Advanced Networking*. XVI, 582 pages. 2006.
- Vol. 4001: E. Dubois, K. Pohl (Eds.), *Advanced Information Systems Engineering*. XVI, 560 pages. 2006.

- Vol. 1999: C. Kop, G. Fliedl, H.C. Mayr, E. Métails (Eds.), *Natural Language Processing and Information Systems*. XIII, 227 pages. 2006.
- Vol. 1998: T. Calamoneri, I. Finocchi, G.F. Italiano (Eds.), *Algorithms and Complexity*. XII, 394 pages. 2006.
- Vol. 1997: W. Grieskamp, C. Weise (Eds.), *Formal Approaches to Software Testing*. XII, 219 pages. 2006.
- Vol. 1996: A. Keller, J.-P. Martin-Flatin (Eds.), *Self-Managed Networks, Systems, and Services*. X, 185 pages. 2006.
- Vol. 1995: G. Müller (Ed.), *Emerging Trends in Information and Communication Security*. XX, 524 pages. 2006.
- Vol. 1994: V.N. Alexandrov, G.D. van Albada, P.M.A. Sloot, J. Dongarra (Eds.), *Computational Science – ICCS 2006, Part IV*. XXXV, 1096 pages. 2006.
- Vol. 1993: V.N. Alexandrov, G.D. van Albada, P.M.A. Sloot, J. Dongarra (Eds.), *Computational Science – ICCS 2006, Part III*. XXXVI, 1136 pages. 2006.
- Vol. 1992: V.N. Alexandrov, G.D. van Albada, P.M.A. Sloot, J. Dongarra (Eds.), *Computational Science – ICCS 2006, Part II*. XXXV, 1122 pages. 2006.
- Vol. 1991: V.N. Alexandrov, G.D. van Albada, P.M.A. Sloot, J. Dongarra (Eds.), *Computational Science – ICCS 2006, Part I*. LXXXI, 1096 pages. 2006.
- Vol. 1990: J. C. Beck, B.M. Smith (Eds.), *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*. X, 301 pages. 2006.
- Vol. 1989: J. Zhou, M. Yung, F. Bao, *Applied Cryptography and Network Security*. XIV, 488 pages. 2006.
- Vol. 1987: M. Hazas, J. Krumm, T. Strang (Eds.), *Location- and Context-Awareness*. X, 289 pages. 2006.
- Vol. 1986: K. Stølen, W.H. Winsborough, F. Martinelli, F. Massacci (Eds.), *Trust Management*. XIV, 474 pages. 2006.
- Vol. 1984: M. Gavrilova, O. Gervasi, V. Kumar, C.J. K. Tan, D. Taniar, A. Laganà, Y. Mun, H. Choo (Eds.), *Computational Science and Its Applications - ICCSA 2006, Part V*. XXV, 1045 pages. 2006.
- Vol. 1983: M. Gavrilova, O. Gervasi, V. Kumar, C.J. K. Tan, D. Taniar, A. Laganà, Y. Mun, H. Choo (Eds.), *Computational Science and Its Applications - ICCSA 2006, Part IV*. XXVI, 1191 pages. 2006.
- Vol. 1982: M. Gavrilova, O. Gervasi, V. Kumar, C.J. K. Tan, D. Taniar, A. Laganà, Y. Mun, H. Choo (Eds.), *Computational Science and Its Applications - ICCSA 2006, Part III*. XXV, 1243 pages. 2006.
- Vol. 1981: M. Gavrilova, O. Gervasi, V. Kumar, C.J. K. Tan, D. Taniar, A. Laganà, Y. Mun, H. Choo (Eds.), *Computational Science and Its Applications - ICCSA 2006, Part II*. XXVI, 1255 pages. 2006.
- Vol. 1980: M. Gavrilova, O. Gervasi, V. Kumar, C.J. K. Tan, D. Taniar, A. Laganà, Y. Mun, H. Choo (Eds.), *Computational Science and Its Applications - ICCSA 2006, Part I*. LXXV, 1199 pages. 2006.
- Vol. 1979: T.S. Huang, N. Sebe, M.S. Lew, V. Pavlović, M. Kölsch, A. Galata, B. Kisačanin (Eds.), *Computer Vision in Human-Computer Interaction*. XII, 121 pages. 2006.
- Vol. 1978: B. Hnich, M. Carlsson, F. Fages, F. Rossi (Eds.), *Recent Advances in Constraints*. VIII, 179 pages. 2006. (Sublibrary LNAI).
- Vol. 1977: N. Fuhr, M. Lalmas, S. Malik, G. Kazai (Eds.), *Advances in XML Information Retrieval and Evaluation*. XII, 556 pages. 2006.
- Vol. 1976: F. Boavida, T. Plagemann, B. Stiller, C. Westphal, E. Monteiro (Eds.), *Networking 2006. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems*. XXVI, 1276 pages. 2006.
- Vol. 1975: S. Mehrotra, D.D. Zeng, H. Chen, B.M. Thuraishingham, F.-Y. Wang (Eds.), *Intelligence and Security Informatics*. XXII, 772 pages. 2006.
- Vol. 1973: J. Wang, Z. Yi, J.M. Zurada, B.-L. Lu, H. Yin (Eds.), *Advances in Neural Networks - ISNN 2006, Part III*. XXIX, 1402 pages. 2006.
- Vol. 1972: J. Wang, Z. Yi, J.M. Zurada, B.-L. Lu, H. Yin (Eds.), *Advances in Neural Networks - ISNN 2006, Part II*. XXVII, 1444 pages. 2006.
- Vol. 1971: J. Wang, Z. Yi, J.M. Zurada, B.-L. Lu, H. Yin (Eds.), *Advances in Neural Networks - ISNN 2006, Part I*. LXVII, 1442 pages. 2006.
- Vol. 1970: T. Braun, G. Carle, S. Fahmy, Y. Koucheryavy (Eds.), *Wired/Wireless Internet Communications*. XIV, 350 pages. 2006.
- Vol. 1969: Ø. Ytrehus (Ed.), *Coding and Cryptography*. XI, 443 pages. 2006.
- Vol. 1968: K.P. Fishkin, B. Schiele, P. Nixon, A. Quigley (Eds.), *Pervasive Computing*. XV, 402 pages. 2006.
- Vol. 1967: D. Grigoriev, J. Harrison, E.A. Hirsch (Eds.), *Computer Science – Theory and Applications*. XVI, 684 pages. 2006.
- Vol. 1966: Q. Wang, D. Pfahl, D.M. Raffo, P. Wernick (Eds.), *Software Process Change*. XIV, 356 pages. 2006.
- Vol. 1965: M. Bernardo, A. Cimatti (Eds.), *Formal Methods for Hardware Verification*. VII, 243 pages. 2006.
- Vol. 1964: M. Ü. Uyar, A.Y. Duale, M.A. Fecko (Eds.), *Testing of Communicating Systems*. XI, 373 pages. 2006.
- Vol. 1963: O. Dikenelli, M.-P. Gleizes, A. Ricci (Eds.), *Engineering Societies in the Agents World VI*. XII, 303 pages. 2006. (Sublibrary LNAI).
- Vol. 1962: W. IJsselstein, Y. de Kort, C. Midden, B. Eggen, E. van den Hoven (Eds.), *Persuasive Technology*. XII, 216 pages. 2006.
- Vol. 1960: R. Vieira, P. Quaresma, M.d.G.V. Nunes, N.J. Mamede, C. Oliveira, M.C. Dias (Eds.), *Computational Processing of the Portuguese Language*. XII, 274 pages. 2006. (Sublibrary LNAI).
- Vol. 1959: J.-Y. Cai, S. B. Cooper, A. Li (Eds.), *Theory and Applications of Models of Computation*. XV, 794 pages. 2006.

Table of Contents

Mobile Agent Rendezvous: A Survey <i>Evangelos Kranakis, Danny Krizanc, Sergio Rajsbaum</i>	1
Adapting to Point Contention with Long-Lived Safe Agreement <i>Hagit Attiya</i>	10
Sensor Networks: Distributed Algorithms Reloaded – or Revolutions? <i>Roger Wattenhofer</i>	24
Local Algorithms for Autonomous Robot Systems <i>Reuven Cohen, David Peleg</i>	29
How to Meet in Anonymous Network <i>Dariusz R. Kowalski, Adam Malinowski</i>	44
Setting Port Numbers for Fast Graph Exploration <i>David Ilcinkas</i>	59
Distributed Chasing of Network Intruders <i>Lélia Blin, Pierre Fraigniaud, Nicolas Nisse, Sandrine Vial</i>	70
Election in the Qualitative World <i>Jérémie Chalopin</i>	85
Fast Deterministic Distributed Algorithms for Sparse Spanners <i>Bilel Derbel, Cyril Gavoille</i>	100
Efficient Distributed Weighted Matchings on Trees <i>Jaap-Henk Hoepman, Shay Kutten, Zvi Lotker</i>	115
Approximation Strategies for Routing Edge Disjoint Paths in Complete Graphs <i>Adrian Kosowski</i>	130
Short Labels by Traversal and Jumping <i>Nicolas Bonichon, Cyril Gavoille, Arnaud Labourel</i>	143
An Optimal Rebuilding Strategy for a Decremental Tree Problem <i>Nicolas Thibault, Christian Laforest</i>	157
Optimal Delay for Media-on-Demand with Pre-loading and Pre-buffering <i>Amotz Bar-Noy, Richard E. Ladner, Tami Tamir</i>	171

Strongly Terminating Early-Stopping k -Set Agreement in Synchronous Systems with General Omission Failures
Philippe Raïpin Parvédy, Michel Raynal, Corentin Travers 182

On Fractional Dynamic Faults with Threshold
Stefan Dobrev, Rastislav Kráľovič, Richard Kráľovič, Nicola Santoro 197

Discovering Network Topology in the Presence of Byzantine Faults
Mikhail Nesterenko, Sébastien Tixeuil 212

Minimum Energy Broadcast and Disk Cover in Grid Wireless Networks
Tiziana Calamoneri, Andrea E.F. Clementi, Miriam Di Ianni, Massimo Lauria, Angelo Monti, Riccardo Silvestri 227

3-D Minimum Energy Broadcasting
Alfredo Navarra 240

Average-Time Complexity of Gossiping in Radio Networks
Bogdan S. Chlebus, Dariusz R. Kowalski, Mariusz A. Rokicki..... 253

$L(h,1,1)$ -Labeling of Outerplanar Graphs
Tiziana Calamoneri, Emanuele G. Fusco, Richard B. Tan, Paola Vocca 268

Combinatorial Algorithms for Compressed Sensing
Graham Cormode, S. Muthukrishnan 280

On the Existence of Truthful Mechanisms for the Minimum-Cost Approximate Shortest-Paths Tree Problem
Davide Bilò, Luciano Gualà, Guido Proietti 295

Dynamic Asymmetric Communication
Travis Gagie 310

Approximate Top- k Queries in Sensor Networks
Boaz Patt-Shamir, Allon Shafir 319

Self-stabilizing Space Optimal Synchronization Algorithms on Trees
Doina Bein, Ajoy K. Datta, Lawrence L. Larmore 334

Distance- k Information in Self-stabilizing Algorithms
Wayne Goddard, Stephen T. Hedetniemi, David P. Jacobs, Vilmar Trevisan 349

Author Index 357

Mobile Agent Rendezvous: A Survey

Evangelos Kranakis¹, Danny Krizanc², and Sergio Rajsbaum³

¹ School of Computer Science, Carleton University, Ottawa, ON, Canada

² Department of Mathematics and Computer Science, Wesleyan University,
Middletown, Connecticut 06459, USA

³ Instituto de Matemáticas, Universidad Nacional Autónoma de México (UNAM),
Ciudad Universitaria, D. F. 04510, Mexico

Abstract. Recent results on the problem of mobile agent rendezvous on distributed networks are surveyed with an emphasis on outlining the various approaches taken by researchers in the theoretical computer science community.

1 Introduction

Consider the following problem originally proposed by Alpern [1] (as quoted in [3]):

Two astronauts land on a spherical body that is much larger than the detection radius (within which they can see each other). The body does not have fixed orientation in space, nor does it have an axis of rotation, so that no common notion of position or direction is available to the astronauts for coordination. Given unit walking speeds for both astronauts, how should they move about so as to minimize the expected meeting time T (before they come within the detection radius)?

This is just one version of a problem that has been studied under many guises under most of which it is referred to as *rendezvous*. In all settings, a set of agents are placed in a domain and are required to all meet at the same place and time within the domain, i.e., rendezvous. The settings differ mainly in the types and properties of the agents and the types and properties of domains. Besides the astronaut example above, rendezvous has been studied in settings as various as ships at sea, mother and child at a mall and autonomous robots on a hilly terrain.

Recently the theoretical computer science community has taken up the challenge of the problem of rendezvous for autonomous software agents moving through a distributed network. Requiring such agents to meet in order to synchronize, share information, divide up duties, etc. would seem to be a natural fundamental operation useful as a subroutine in more complicated applications such as web-crawling, peer-to-peer lookup, meeting scheduling, etc. In this paper, we provide a short (perhaps biased) survey of recent work done on this version of the problem. The research done in other settings is extensive and many of

the solutions can be applied here. But it is often the case that the models used and the concerns studied are sufficiently different as to require new approaches. Having said that, one ignores the earlier work at one's peril. For an excellent discussion of mainly continuous domains with randomized agents see the book by Alpern and Gal [3]. For work on robot rendezvous consider [26] and [28] as possible starting points.

2 The Model

The definition of rendezvous must begin with establishing the properties of the agents that will rendezvous and the domain in which rendezvous will occur. The model below captures the essence of what has been termed *the theory of mobile agent computing*.

2.1 Mobile Agents

We are interested in modeling a set of software entities that act more or less autonomously from their originator and have the ability to move from node to node in a distributed network maintaining some sort of state with the nodes of the network providing some amount of (possibly longterm) storage and computational support. Either explicitly or implicitly such a mobile (software) agent has most often been modeled using a finite automaton consisting of a set of states and a transition function. The transition function takes as input the agent's current state as well as possibly the state of the node it resides in and outputs a new agent state, possible modifications to the current node's state and a possible move to another node. In some instances we consider probabilistic automata which have available a source of randomness that is used as part of their input. Such agents are referred to as *randomized agents*.

An important property to consider is whether or not the agents are distinguishable, i.e., if they have distinct labels or identities. Agents without identities are referred to as *anonymous agents*. Anonymous agents are limited to running precisely the same program, i.e., they are identical finite automata. As the identity is assumed to be part of the starting state of the automaton, agents with identities have the potential to run different programs.

The knowledge the agent has about the network it is on and about the other agents can make a difference in the solvability and efficiency of rendezvous. For example, knowledge of the size of the network or its topology or the number of and identities of the other agents may be used as part of the program for rendezvous. If available to the agents, this information is assumed to be part of its starting state. (One could imagine situations where the information is made available by the nodes of the network and not necessarily encoded in the agent.)

Other properties that may be considered in mobile agent computing include whether or not the agents have the ability to "clone" themselves, whether or not they have the ability to "merge" upon meeting (sometimes referred to as "sticky" agents) or whether or not they can send self-generated messages. At this point, most of the research on rendezvous ignores these properties and they will not be discussed below.

2.2 Distributed Networks

The model of a distributed network is essentially inherited directly from the theory of distributed computing. We model the network by a graph whose vertices comprise the computing nodes and edges correspond to communication links.

The nodes of the network may or may not have distinct identities. In an *anonymous* network the nodes have no identities. In particular this means that an agent can not distinguish two nodes except perhaps by their degree. The outgoing edges of a node are usually thought of as distinguishable but an important distinction is made between a globally consistent edge-labelling versus a locally independent edge-labelling. A simple example is the case of a ring where clockwise and counterclockwise edges are marked consistently around the ring in one case, and the edges are arbitrarily - say by an adversary - marked 1 and 2 in the other case. If the labelling satisfies certain coding properties it is called a *sense of direction* [13]. Sense of direction has turned out to greatly effect the solvability and efficiency of solution of a number of problems in distributed computing and has been shown to be important in rendezvous as well.

Networks are also classified by how they deal with time. In a synchronous network there exists a global clock available to all nodes. This global clock is inherited by the agents. In particular it is usually assumed that in a single step an agent arrives at a node, performs some calculation, and exits the node and that all agents are performing these tasks "in sync". In an asynchronous network such a global clock is not available. The speed with which an agent computes or moves between nodes, while guaranteed to be finite, is not a priori determined.

Finally we have to consider the resources provided by the nodes to the agents. All nodes are assumed to provide enough space to store the agent temporarily and computing power for it to perform its tasks. (The case of malicious nodes refusing agents or even worse destroying agents - so-called *blackholes* - is also sometimes considered.) Beyond these basic services one considers nodes that might provide some form of long-term storage, i.e., state that is left behind when the agent leaves. In the rendezvous problem the idea of leaving an indistinguishable mark or *token* at a node (introduced in [5]) has been studied. More accommodating nodes might provide a *whiteboard* for agents to write messages to be left for themselves or for other agents.

3 The Rendezvous Problem

Given a particular agent model (e.g., deterministic, anonymous agents with knowledge they are on a ring of size n) and network model (e.g., anonymous, synchronous with tokens) a set of k agents distributed arbitrarily over the nodes of the network are said to *rendezvous* if after running their programs after some finite time they all occupy the same node of the network at the same time. It is generally assumed that two agents occupying the same node can recognize this fact (though in many instances this fact is not required for rendezvous to occur). As stated, rendezvous is assumed to occur at nodes. In some instances one considers the possibility of rendezvous on an edge, i.e., if both agents use the

same edge (in opposite directions) at the same time. (For physical robots this makes sense. For software agents this perhaps is not so realistic but sometimes necessary to allow for the possibility of rendezvous at all - especially in instances where the network lacks a sense of direction.)

The first question one asks for an instance of rendezvous is whether or not it is solvable. There are many situations where it is not possible to rendezvous at all. This will depend upon both the properties of the agents (deterministic or randomized, anonymous or with identities, knowledge of the size of the network or not, etc.) and the network (synchronous or asynchronous, anonymous or with identities, tokens available or not, etc.). The solvability is also a function of the starting positions chosen for the agents. For example, if the agents start at the same node and can recognize this fact, rendezvous is possible in this instance. Given a situation where some starting positions are not solvable (i.e., rendezvous is not possible) but others are, we distinguish between algorithms that are guaranteed to finish for all starting positions, with successful rendezvous when possible but otherwise recognizing that rendezvous is impossible, versus algorithms that are only guaranteed to halt when rendezvous is possible. Algorithms of the former type are said to solve *rendezvous with detection*. (The distinction is perhaps analogous to Turing machines deciding versus accepting a language.)

For solvable instances of rendezvous one is interested in comparing the efficiency of different solutions. Much of the research focuses on the time required to rendezvous. In the synchronous setting the time is measured via the global clock. (In some situations, it makes a difference if the agents begin their rendezvous procedure at the same time or there is possible delay between start times.) In the asynchronous setting we adapt the standard time measures from the distributed computing model. Also of interest is the size of the program required by the agents to solve the problem. This is referred to as the memory requirement of the agents and is considered to be proportional to the base two logarithm of the number of states required by the finite state machine encoding the agent.

As is often the case, researchers are interested in examining the extremes in order to get an understanding of the limits a problem imposes. Over time it has become clear that for rendezvous symmetry (of the agents and the network) plays a central role in determining its solvability and the efficiency of its solutions. As such we divide our discussion below into the asymmetric and symmetric cases. For simplicity we restrict ourselves to the case of just two agents in most of the discussion below.

4 Asymmetric Rendezvous

Asymmetry in a rendezvous problem may arise from either the network or the agents.

4.1 Network Asymmetry

A network is asymmetric if it has one or more uniquely distinguishable vertices. A simple example is the case of a network where all of the nodes have

unique identities chosen from a subset of some totally ordered set such as the integers. In this case, the node labelled with the smallest identity (for example) is unique and may be used as a meeting point for a rendezvous algorithm. Uniqueness need not be conferred using node labels. For example, in a network where there is a unique node of degree one, it may be used as a focal point.

If a “map” of the graph with an agent’s starting position marked on it is available to the agents then the problem of rendezvous is easily solved by just traversing the path to an agreed upon unique node. Algorithms that use an agreed upon meeting place are referred to by Alpern and Gal [3] as FOCAL strategies. In the case where the graph is not available in advance but the agents know that a focal point exists (e.g., they know the nodes are uniquely labelled and therefore there exists a unique minimum label node) this strategy reduces to the problem of graph traversal or graph exploration whereby all of the nodes (sometimes edges) of the graph are to be visited by an agent. This has been extensively studied in a number of contexts, e.g., [9, 25]. Much of the work in this area has looked at improving the efficiency (time or memory) for restricted classes of graphs, e.g., trees [11]. A closely related problem is that of robot exploration of an unknown environment with obstacles which can often be modeled using graphs. See for example [6].

4.2 Agent Asymmetry

By agent asymmetry one generally means the agents have unique identities that allow them to act differently depending upon their values. In the simplest scenario of two agents, the agent with the smaller value could decide to wait at its starting position for the other agent to find it by exploring the graph as above. Alpern and Gal [3] refer to this as the Wait For Mommy (WFM) strategy and they show it to be optimal under certain conditions.

WFM depends upon the fact that the agents know in advance the identities associated with the other agents. In some situations this may be an unrealistic assumption. Yu and Yang [29] were the first to consider this problem. Under the assumption that the algorithm designer may assign the identities to the agents (as well as the existence of distinct whiteboards for each agent), they show that rendezvous may be achieved deterministically on a synchronous network in $O(nl)$ steps where n is the size of the network and l is the size of the identities assigned. The perhaps more interesting case where an adversary assigns the labels was first considered in [10]. Extensions to this work including showing rendezvous on an arbitrary graph is possible in time polynomial in n and l and that there exist graphs requiring $\Omega(n^2)$ time for rendezvous are described in [17, 18]. The case of an asynchronous network is considered in [8] where a (non-polynomial) upper bound is set for rendezvous in arbitrary graphs (assuming the agents have an upper bound on the size of the graph). Improvements (in some cases optimal) for the case of the ring network are discussed in each of the above papers.

5 Symmetric Rendezvous

In the case of symmetric rendezvous, both the (generally synchronous) network and the agents are assumed to be anonymous. Further one considers classes of networks that in the worst case contain highly symmetric networks that do not submit to a FOCAL strategy. As might be expected some mechanism is required to break symmetry in order for rendezvous to be possible. The use of randomization and of tokens to break symmetry have both been studied extensively.

5.1 Randomized Rendezvous

Many authors have observed that rendezvous may be solved by anonymous agents on an anonymous network by having the agents perform a random walk. The expected time to rendezvous is then a (polynomial) function of the (size of the) network and is directly related to the cover time of the network. (See [24] for definitions relating to random walks.)

For example, it is straightforward to show that two agents performing a symmetric random walk on ring of size n will rendezvous within expected $O(n^2)$ time. This expected time can be improved by considering the following strategy (for a ring with sense of direction). Repeat the following until rendezvous is achieved: flip a (fair) coin and walk $n/2$ steps to the right if the result is heads, $n/2$ steps to the left if the result is tails. If the two agents choose different directions (which they do with probability $1/2$) then they will rendezvous (at least on an edge if not at a node). It is easy to see that expected time until rendezvous is $O(n)$. Alpern refers to this strategy as Coin Half Tour and studies it in detail in [2]. Note that the agents are required to count up to n and thus seem to require $O(\log n)$ bits of memory to perform this algorithm (whereas the straightforward random walk requires only a constant number of states to implement). It can be shown that $O(\log \log n)$ bits are sufficient for achieving linear expected rendezvous time[23].

5.2 Rendezvous Using Tokens

The idea of using tokens or marks to break symmetry for rendezvous was first suggested in [5] and expanded upon for the case of the ring in [27]. The first observation to make is that rendezvous is impossible for deterministic agents with tokens (or whiteboards) on an even size ring when the agents start at distance $n/2$ as the agents will remain in symmetric positions indefinitely. However, this is the only starting position for the agents for which rendezvous is impossible. This leads one to consider algorithms for rendezvous with detection where rendezvous is achieved when possible and otherwise the agents detect they are in an impossible to rendezvous situation. In this case, a simple algorithm suffices (described here for the oriented case). Each agent marks their starting position with a token. They then travel once around the ring counting the distances between their tokens. If the two distances are the same, they halt declaring rendezvous impossible. If they are different they agree to meet (for example) in the middle of the shorter side.