Suleyman Cenk Sahinalp
S. Muthukrishnan
Ugur Dogrusoz (Eds.)

LNCS 3109

Combinatorial Pattern Matching

15th Annual Symposium, CPM 2004
Istanbul, Turkey, July 2004
Proceedings

Springer

Suleyman Cenk Sahinalp S. Muthukrishnan
Ugur Dogrusoz (Eds.)

Combinatorial Pattern Matching

15th Annual Symposium, CPM 2004
Istanbul, Turkey, July 5-7, 2004
Proceedings

Springer

Volume Editors

Suleyman Cenk Sahinalp
Simon Fraser University, School of Computing Science
Burnaby BC, V5A 1S6 Canada
E-mail: cenk@cs.sfu.ca

S. Muthukrishnan
Rutgers University, Department of Computer and Information Sciences
319 Core Bldg, 110 Frelinghuysen Rd, Piscataway, NJ 08854, USA
E-mail: muthu@cs.rutgers.edu

Ugur Dogrusoz
Bilkent University, Computer Engineering Department
06800 Ankara, Turkey
E-mail: ugur@cs.bilkent.edu.tr

Library of Congress Control Number: 2004108252

CR Subject Classification (1998): F.2.2, I.5.4, I.5.0, I.7.3, H.3.3, E.4, G.2.1, E.1

ISSN 0302-9743
ISBN 3-540-22341-X Springer-Verlag Berlin Heidelberg New York

Typesetting: Camera-ready by author, data conversion by Olgun Computergrafik
Printed on acid-free paper SPIN: 11014331 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3109

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

Preface

The 15th Annual Symposium on Combinatorial Pattern Matching was held in Ciragan Palace Hotel, Istanbul, Turkey during July 5–7, 2004. CPM 2004 repeated the success of its predecessors; it even surpassed them in terms of the number of invited speakers, the number of submissions, and the number of papers accepted and presented at the conference.

In response to the call for papers, CPM 2004 received a record number of 79 high-quality submissions. Each submission was reviewed by at least three program committee members and the comments were returned to the authors. Following an extensive electronic discussion period, the Program Committee accepted 36 of the submissions to be presented at the conference. They constitute original research contributions in combinatorial pattern matching algorithms and data structures, molecular sequence analysis, phylogenetic tree construction, and RNA and protein structure analysis and prediction.

CPM 2004 had five invited speakers. In alphabetical order they were: Evan Eichler from the University of Washington, USA, Martin Farach-Colton from Rutgers University, USA, Paolo Ferragina from the University of Pisa, Italy, Piotr Indyk from MIT, USA, and Gene Myers from the University of California, Berkeley, USA.

It is impossible to organize such a successful program without the help of many individuals. We would like to express our appreciation to the authors of the submitted papers and to the program committee members and external referees, who provided timely and significant reviews.

July 2004

S.C. Sahinalp,
S. Muthukrishnan,
U. Dogrusoz

Organization

CPM 2004 was locally organized by Bilkent University, Ankara, Turkey. Within Bilkent University, the Center for Bioinformatics (BCBI) and the Computer Engineering Department cooperated.

Executive Committee

Organization Chair	Ugur Dogrusoz (Bilkent University)
Program Chairs	Suleyman Cenk Sahinalp (Simon Fraser University) S. Muthukrishnan (Rutgers Universtiy and AT&T)
Student Volunteer Chairs	Asli Ayaz (Bilkent University) Ozgun Babur (Bilkent University) Emek Demir (Bilkent University)
Social Events	Tasmanlar Tourism, Ankara, Turkey
Administrative Support	Gurkan Bebek (Case Western Reserve University) Emek Demir (Bilkent University)

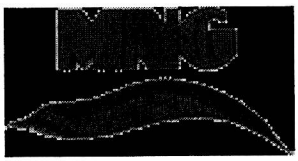
Program Committee

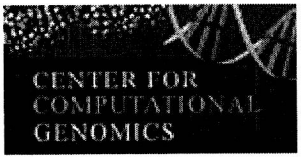
Gerth Stølting Brodal	University of Aarhus, Denmark
Jeremy Buhler	Washington University, USA
Ugur Dogrusoz	Bilkent University, Turkey
Zvi Galil	Columbia University, USA
Ramesh Hariharan	Indian Institute of Science, India
Ming Li	University of Waterloo, Canada
Stefano Lonardi	University of California, Riverside, USA
Ian Munro	University of Waterloo, Canada
Craig Neville Manning	Google Inc., USA
S. Muthukrishnan	Rutgers University and AT&T, USA
Joseph Nadeau	Case Western Reserve University, USA
Meral Ozsoyoglu	Case Western Reserve University, USA
Ely Porat	Bar-Ilan University, Israel
Kunihiko Sadakane	Kyushu University, Japan
Suleyman Cenk Sahinalp	Simon Fraser University, Canada
Mona Singh	Princeton University, USA

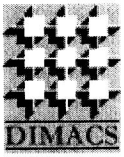
Steering Committee

Alberto Apostolico	Purdue University, USA and University of Padua, Italy
Maxime Crochemore	University of Marne la Vallée, France
Zvi Galil	Columbia University, USA
Udi Manber	A9.com, USA

Sponsoring Organizations







External Referees

Can Alkan
Abdullah Arslan
Asli Ayaz
Kensuke BaBa
Ozgun Babur
Gary Benson
Petra Berenbrink
Bernard Chazelle
Tim Ting Chen
Richard Cole
Livio Colussi
Maxime Crochemore
Sergio De Agostino
Emek Demir
Zeynep Erson
Kimmo Fredriksson
Leszek Gasieniec
Erhan Giral
Danny Hermelin
Lucian Ilie
Tao Jiang
Haim Kaplan
Emre Karakoc
Carmel Kent
Ali Reza Khodabakhshi
Takuya Kida
Carl Kingsford
Moshe Koppel
Stefan Kurtz
Jesper Larsson
Moshe Lewenstein
Guohui Lin
Hao Lin
Bin Ma
Veli Makinen
Yishay Mansour
Giovanni Manzini
Gabriel Moruz
Milan Mosny
Giovanni Motta
Giulio Pavesi
Frederic Pio
Teresa Przytycka
Mathieu Raffinot
Francesco Rizzo
Igor B. Rogozin
Wojciech Rytter
Anoop Sarkar
Nira Shafrir
Dana Shapira
Tetsuo Shibuya
Dina Sokol
Masayuki Takeda
Andrey Utis
Anil Vullikanti
Oren Weimann
Kaizhong Zhang
Qiangfeng Zhang
Zefeng Zhang
Jie Zheng

Lecture Notes in Computer Science

For information about Vols. 1–3005

please contact your bookseller or Springer-Verlag

Vol. 3120: J. Shawe-Taylor, Y. Singer (Eds.), Learning Theory. X, 648 pages. 2004.

Vol. 3109: S.C. Sahinalp, S. Muthukrishnan, U. Dogrusoz (Eds.), Combinatorial Pattern Matching. XII, 486 pages. 2004.

Vol. 3105: S. Göbel, U. Spierling, A. Hoffmann, I. Iurgel, O. Schneider, J. Dechau, A. Feix (Eds.), Technologies for Interactive Digital Storytelling and Entertainment. XVI, 304 pages. 2004.

Vol. 3099: J. Cortadella, W. Reisig (Eds.), Applications and Theory of Petri Nets 2004. XI, 505 pages. 2004.

Vol. 3098: J. Desel, W. Reisig, G. Rozenberg (Eds.), Advanced Course on Petri Nets. VIII, 849 pages. 2004.

Vol. 3096: G. Melnik, H. Holz (Eds.), Advances in Learning Software Organizations. X, 173 pages. 2004.

Vol. 3094: A. Nürnberger, M. Detyniecki (Eds.), Adaptive Multimedia Retrieval. VIII, 229 pages. 2004.

Vol. 3093: S.K. Katsikas, S. Gritzalis, J. Lopez (Eds.), Public Key Infrastructure. XIII, 380 pages. 2004.

Vol. 3092: J. Eckstein, H. Baumeister (Eds.), Extreme Programming and Agile Processes in Software Engineering. XVI, 358 pages. 2004.

Vol. 3091: V. van Oostrom (Ed.), Rewriting Techniques and Applications. X, 313 pages. 2004.

Vol. 3089: M. Jakobsson, M. Yung, J. Zhou (Eds.), Applied Cryptography and Network Security. XIV, 510 pages. 2004.

Vol. 3086: M. Odersky (Ed.), ECOOP 2004 – Object-Oriented Programming. XIII, 611 pages. 2004.

Vol. 3085: S. Berardi, M. Coppo, F. Damiani (Eds.), Types for Proofs and Programs. X, 409 pages. 2004.

Vol. 3084: A. Persson, J. Stirna (Eds.), Advanced Information Systems Engineering. XIV, 596 pages. 2004.

Vol. 3083: W. Emmerich, A.L. Wolf (Eds.), Component Deployment. X, 249 pages. 2004.

Vol. 3079: Z. Mammeri, P. Lorenz (Eds.), High Speed Networks and Multimedia Communications. XVIII, 1103 pages. 2004.

Vol. 3078: S. Cotin, D.N. Metaxas (Eds.), Medical Simulation. XVI, 296 pages. 2004.

Vol. 3077: F. Roli, J. Kittler, T. Windeatt (Eds.), Multiple Classifier Systems. XII, 386 pages. 2004.

Vol. 3076: D. Buell (Ed.), Algorithmic Number Theory. XI, 451 pages. 2004.

Vol. 3074: B. Kuijpers, P. Revesz (Eds.), Constraint Databases and Applications. XII, 181 pages. 2004.

Vol. 3073: H. Chen, R. Moore, D.D. Zeng, J. Leavitt (Eds.), Intelligence and Security Informatics. XV, 536 pages. 2004.

Vol. 3072: D. Zhang, A.K. Jain (Eds.), Biometric Authentication. XVII, 800 pages. 2004.

Vol. 3070: L. Rutkowski, J. Siekmann, R. Tadeusiewicz, L.A. Zadeh (Eds.), Artificial Intelligence and Soft Computing - ICAISC 2004. XXV, 1208 pages. 2004. (Subseries LNAI).

Vol. 3068: E. André, L. Dybkjaer, W. Minker, P. Heisterkamp (Eds.), Affective Dialogue Systems. XII, 324 pages. 2004. (Subseries LNAI).

Vol. 3067: M. Dastani, J. Dix, A. El Fallah-Seghrouchni (Eds.), Programming Multi-Agent Systems. X, 221 pages. 2004. (Subseries LNAI).

Vol. 3066: S. Tsumoto, R. S lowiński, J. Komorowski, J.W. Grzymala-Busse (Eds.), Rough Sets and Current Trends in Computing. XX, 853 pages. 2004. (Subseries LNAI).

Vol. 3065: A. Lomuscio, D. Nute (Eds.), Deontic Logic in Computer Science. X, 275 pages. 2004. (Subseries LNAI).

Vol. 3064: D. Bienstock, G. Nemhauser (Eds.), Integer Programming and Combinatorial Optimization. XI, 445 pages. 2004.

Vol. 3063: A. Llamosí, A. Strohmeier (Eds.), Reliable Software Technologies - Ada-Europe 2004. XIII, 333 pages. 2004.

Vol. 3062: J.L. Pfaltz, M. Nagl, B. Böhlen (Eds.), Applications of Graph Transformations with Industrial Relevance. XV, 500 pages. 2004.

Vol. 3061: F.F. Ramas, H. Unger, V. Larios (Eds.), Advanced Distributed Systems. VIII, 285 pages. 2004.

Vol. 3060: A.Y. Tawfik, S.D. Goodwin (Eds.), Advances in Artificial Intelligence. XIII, 582 pages. 2004. (Subseries LNAI).

Vol. 3059: C.C. Ribeiro, S.L. Martins (Eds.), Experimental and Efficient Algorithms. X, 586 pages. 2004.

Vol. 3058: N. Sebe, M.S. Lew, T.S. Huang (Eds.), Computer Vision in Human-Computer Interaction. X, 233 pages. 2004.

Vol. 3057: B. Jayaraman (Ed.), Practical Aspects of Declarative Languages. VIII, 255 pages. 2004.

Vol. 3056: H. Dai, R. Srikant, C. Zhang (Eds.), Advances in Knowledge Discovery and Data Mining. XIX, 713 pages. 2004. (Subseries LNAI).

Vol. 3055: H. Christiansen, M.-S. Hacid, T. Andreasen, H.L. Larsen (Eds.), Flexible Query Answering Systems. X, 500 pages. 2004. (Subseries LNAI).

Vol. 3054: I. Crnkovic, J.A. Stafford, H.W. Schmidt, K. Wallnau (Eds.), Component-Based Software Engineering. XI, 311 pages. 2004.

Vol. 3053: C. Bussler, J. Davies, D. Fensel, R. Studer (Eds.), The Semantic Web: Research and Applications. XIII, 490 pages. 2004.

Vol. 3052: W. Zimmermann, B. Thalheim (Eds.), Abstract State Machines 2004. Advances in Theory and Practice. XII, 235 pages. 2004.

Vol. 3051: R. Berghammer, B. Möller, G. Struth (Eds.), Relational and Kleene-Algebraic Methods in Computer Science. X, 279 pages. 2004.

Vol. 3050: J. Domingo-Ferrer, V. Torra (Eds.), Privacy in Statistical Databases. IX, 367 pages. 2004.

Vol. 3049: M. Bruynooghe, K.-K. Lau (Eds.), Program Development in Computational Logic. VIII, 539 pages. 2004.

Vol. 3047: F. Oquendo, B. Warboys, R. Morrison (Eds.), Software Architecture. X, 279 pages. 2004.

Vol. 3046: A. Laganà, M.L. Gavrilova, V. Kumar, Y. Mun, C.K. Tan, O. Gervasi (Eds.), Computational Science and Its Applications – ICCSA 2004. LIII, 1016 pages. 2004.

Vol. 3045: A. Laganà, M.L. Gavrilova, V. Kumar, Y. Mun, C.K. Tan, O. Gervasi (Eds.), Computational Science and Its Applications – ICCSA 2004. LIII, 1040 pages. 2004.

Vol. 3044: A. Laganà, M.L. Gavrilova, V. Kumar, Y. Mun, C.K. Tan, O. Gervasi (Eds.), Computational Science and Its Applications – ICCSA 2004. LIII, 1140 pages. 2004.

Vol. 3043: A. Laganà, M.L. Gavrilova, V. Kumar, Y. Mun, C.K. Tan, O. Gervasi (Eds.), Computational Science and Its Applications – ICCSA 2004. LIII, 1180 pages. 2004.

Vol. 3042: N. Mitrou, K. Kontovasilis, G.N. Rouskas, I. Iliadis, L. Merakos (Eds.), NETWORKING 2004, Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications. XXXIII, 1519 pages. 2004.

Vol. 3040: R. Conejo, M. Urretavizcaya, J.-L. Pérez-de-la-Cruz (Eds.), Current Topics in Artificial Intelligence. XIV, 689 pages. 2004. (Subseries LNAI).

Vol. 3039: M. Bubak, G.D.v. Albada, P.M. Sloot, J.J. Dongarra (Eds.), Computational Science - ICCS 2004. LXVI, 1271 pages. 2004.

Vol. 3038: M. Bubak, G.D.v. Albada, P.M. Sloot, J.J. Dongarra (Eds.), Computational Science - ICCS 2004. LXVI, 1311 pages. 2004.

Vol. 3037: M. Bubak, G.D.v. Albada, P.M. Sloot, J.J. Dongarra (Eds.), Computational Science - ICCS 2004. LXVI, 745 pages. 2004.

Vol. 3036: M. Bubak, G.D.v. Albada, P.M. Sloot, J.J. Dongarra (Eds.), Computational Science - ICCS 2004. LXVI, 713 pages. 2004.

Vol. 3035: M.A. Wimmer (Ed.), Knowledge Management in Electronic Government. XII, 326 pages. 2004. (Subseries LNAI).

Vol. 3034: J. Favela, E. Menasalvas, E. Chávez (Eds.), Advances in Web Intelligence. XIII, 227 pages. 2004. (Subseries LNAI).

Vol. 3033: M. Li, X.-H. Sun, Q. Deng, J. Ni (Eds.), Grid and Cooperative Computing. XXXVIII, 1076 pages. 2004.

Vol. 3032: M. Li, X.-H. Sun, Q. Deng, J. Ni (Eds.), Grid and Cooperative Computing. XXXVII, 1112 pages. 2004.

Vol. 3031: A. Butz, A. Krüger, P. Olivier (Eds.), Smart Graphics. X, 165 pages. 2004.

Vol. 3030: P. Giorgini, B. Henderson-Sellers, M. Winikoff (Eds.), Agent-Oriented Information Systems. XIV, 207 pages. 2004. (Subseries LNAI).

Vol. 3029: B. Orchard, C. Yang, M. Ali (Eds.), Innovations in Applied Artificial Intelligence. XXI, 1272 pages. 2004. (Subseries LNAI).

Vol. 3028: D. Neuenschwander, Probabilistic and Statistical Methods in Cryptology. X, 158 pages. 2004.

Vol. 3027: C. Cachin, J. Camenisch (Eds.), Advances in Cryptology - EUROCRYPT 2004. XI, 628 pages. 2004.

Vol. 3026: C. Ramamoorthy, R. Lee, K.W. Lee (Eds.), Software Engineering Research and Applications. XV, 377 pages. 2004.

Vol. 3025: G.A. Vouros, T. Panayiotopoulos (Eds.), Methods and Applications of Artificial Intelligence. XV, 546 pages. 2004. (Subseries LNAI).

Vol. 3024: T. Pajdla, J. Matas (Eds.), Computer Vision - ECCV 2004. XXVIII, 621 pages. 2004.

Vol. 3023: T. Pajdla, J. Matas (Eds.), Computer Vision - ECCV 2004. XXVIII, 611 pages. 2004.

Vol. 3022: T. Pajdla, J. Matas (Eds.), Computer Vision - ECCV 2004. XXVIII, 621 pages. 2004.

Vol. 3021: T. Pajdla, J. Matas (Eds.), Computer Vision - ECCV 2004. XXVIII, 633 pages. 2004.

Vol. 3019: R. Wyrzykowski, J.J. Dongarra, M. Paprzycki, J. Wasniewski (Eds.), Parallel Processing and Applied Mathematics. XIX, 1174 pages. 2004.

Vol. 3018: M. Bruynooghe (Ed.), Logic Based Program Synthesis and Transformation. X, 233 pages. 2004.

Vol. 3017: B. Roy, W. Meier (Eds.), Fast Software Encryption. XI, 485 pages. 2004.

Vol. 3016: C. Lengauer, D. Batory, C. Consel, M. Odersky (Eds.), Domain-Specific Program Generation. XII, 325 pages. 2004.

Vol. 3015: C. Barakat, I. Pratt (Eds.), Passive and Active Network Measurement. XI, 300 pages. 2004.

Vol. 3014: F. van der Linden (Ed.), Software Product-Family Engineering. IX, 486 pages. 2004.

Vol. 3012: K. Kurumatani, S.-H. Chen, A. Ohuchi (Eds.), Multi-Agnets for Mass User Support. X, 217 pages. 2004. (Subseries LNAI).

Vol. 3011: J.-C. Régin, M. Rueher (Eds.), Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems. XI, 415 pages. 2004.

Vol. 3010: K.R. Apt, F. Fages, F. Rossi, P. Szeredi, J. Váncza (Eds.), Recent Advances in Constraints. VIII, 285 pages. 2004. (Subseries LNAI).

Vol. 3009: F. Bomarius, H. Iida (Eds.), Product Focused Software Process Improvement. XIV, 584 pages. 2004.

Vol. 3008: S. Heuel, Uncertain Projective Geometry. XVII, 205 pages. 2004.

Vol. 3007: J.X. Yu, X. Lin, H. Lu, Y. Zhang (Eds.), Advanced Web Technologies and Applications. XXII, 936 pages. 2004.

Vol. 3006: M. Matsui, R. Zuccherato (Eds.), Selected Areas in Cryptography. XI, 361 pages. 2004.

Table of Contents

Sorting by Reversals in Subquadratic Time 1
Eric Tannier and Marie-France Sagot

Computational Problems in Perfect Phylogeny Haplotyping: Xor-Genotypes and Tag SNPs 14
Tamar Barzuza, Jacques S. Beckmann, Ron Shamir, and Itsik Pe'er

Sorting by Length-Weighted Reversals: Dealing with Signs and Circularity 32
Firas Swidan, Michael A. Bender, Dongdong Ge, Simai He, Haodong Hu, and Ron Y. Pinter

Optimizing Multiple Spaced Seeds for Homology Search..... 47
Jinbo Xu, Daniel G. Brown, Ming Li, and Bin Ma

Approximate Labelled Subtree Homeomorphism 59
Ron Y. Pinter, Oleg Rokhlenko, Dekel Tsur, and Michal Ziv-Ukelson

On the Average Sequence Complexity..... 74
Svante Janson, Stefano Lonardi, and Wojciech Szpankowski

Approximate Point Set Pattern Matching on Sequences and Planes..... 89
Tomoaki Suga and Shinichi Shimozono

Finding Biclusters by Random Projections 102
Stefano Lonardi, Wojciech Szpankowski, and Qiaofeng Yang

Real-Time String Matching in Sublinear Space..... 117
Leszek Gąsieniec and Roman Kolpakov

On the k-Closest Substring and k-Consensus Pattern Problems 130
Yishan Jiao, Jingyi Xu, and Ming Li

A Trie-Based Approach for Compacting Automata 145
Maxime Crochemore, Chiara Epifanio, Roberto Grossi, and Filippo Mignosi

A Simple Optimal Representation for Balanced Parentheses 159
Richard F. Geary, Naila Rahman, Rajeev Raman, and Venkatesh Raman

Two Algorithms for LCS Consecutive Suffix Alignment 173
Gad M. Landau, Eugene Myers, and Michal Ziv-Ukelson

Efficient Algorithms for Finding Submasses in Weighted Strings..... 194
Nikhil Bansal, Mark Cieliebak, and Zsuzsanna Lipták

Maximum Agreement and Compatible Supertrees (Extended Abstract) ... 205
Vincent Berry and François Nicolas

Polynomial-Time Algorithms
for the Ordered Maximum Agreement Subtree Problem ... 220
Anders Dessmark, Jesper Jansson, Andrzej Lingas, and Eva-Marta Lundell

Small Phylogeny Problem: Character Evolution Trees ... 230
Arvind Gupta, Ján Maňuch, Ladislav Stacho, and Chenchen Zhu

The Protein Sequence Design Problem in Canonical Model
on 2D and 3D Lattices ... 244
Piotr Berman, Bhaskar DasGupta, Dhruv Mubayi, Robert Sloan, György Turán, and Yi Zhang

A Computational Model for RNA Multiple Structural Alignment ... 254
Eugene Davydov and Serafim Batzoglou

Computational Design of New and Recombinant Selenoproteins ... 270
Rolf Backofen and Anke Busch

A Combinatorial Shape Matching Algorithm for Rigid Protein Docking ... 285
Vicky Choi and Navin Goyal

Multi-seed Lossless Filtration (Extended Abstract) ... 297
Gregory Kucherov, Laurent Noé, and Mikhail Roytberg

New Results for the 2-Interval Pattern Problem ... 311
Guillaume Blin, Guillaume Fertin, and Stéphane Vialette

A Linear-Time Algorithm for Computing Translocation Distance
between Signed Genomes ... 323
Guojun Li, Xingqin Qi, Xiaoli Wang, and Binhai Zhu

Sparse Normalized Local Alignment ... 333
Nadav Efraty and Gad M. Landau

Quadratic Time Algorithms for Finding Common Intervals
in Two and More Sequences ... 347
Thomas Schmidt and Jens Stoye

Maximal Common Connected Sets of Interval Graphs ... 359
Michel Habib, Christophe Paul, and Mathieu Raffinot

Performing Local Similarity Searches with Variable Length Seeds ... 373
Miklós Csűrös

Reversal Distance without Hurdles and Fortresses ... 388
Anne Bergeron, Julia Mixtacki, and Jens Stoye

A Fast Set Intersection Algorithm for Sorted Sequences 400
Ricardo Baeza-Yates

Faster Two Dimensional Pattern Matching with Rotations 409
Amihood Amir, Oren Kapah, and Dekel Tsur

Compressed Compact Suffix Arrays 420
Veli Mäkinen and Gonzalo Navarro

Approximate String Matching Using Compressed Suffix Arrays 434
Trinh N.D. Huynh, Wing-Kai Hon, Tak-Wah Lam, and Wing-Kin Sung

Compressed Index for a Dynamic Collection of Texts 445
Ho-Leung Chan, Wing-Kai Hon, and Tak-Wah Lam

Improved Single and Multiple Approximate String Matching 457
Kimmo Fredriksson and Gonzalo Navarro

Average-Case Analysis of Approximate Trie Search (Extended Abstract) 472
Moritz G. Maaß

Author Index 485

Sorting by Reversals in Subquadratic Time*

Eric Tannier[1] and Marie-France Sagot[1,2]

[1] INRIA Rhône-Alpes, Laboratoire de Biométrie et Biologie Évolutive
Université Claude Bernard, 69622 Villeurbanne cedex, France
{Eric.Tannier,Marie-France.Sagot}@inrialpes.fr
[2] King's College, London, UK

Abstract. The problem of sorting a signed permutation by reversals is inspired by genome rearrangements in computational molecular biology. Given two genomes represented as two signed permutations of the same elements (*e.g. orthologous genes*), the problem consists in finding a most parsimonious scenario of reversals that transforms one genome into the other. We propose a method for sorting a signed permutation by reversals in time $O(n\sqrt{n \log n})$. The best known algorithms run in time $O(n^2)$, the main obstacle to an improvement being a costly operation of detection of so-called "safe" reversals. We bypass this detection and, using the same data structure as a previous random approximation algorithm, we achieve the same subquadratic complexity for finding an *exact* optimal solution. This answers an open question by Ozery-Flato and Shamir whether a subquadratic complexity could ever be achieved for solving the problem.

1 Introduction

The problem of sorting a signed permutation by reversals is inspired by a problem of genome rearrangement in computational biology. Genome rearrangements such as reversals may change the order of the genes (or of other markers) in a genome, and also the direction of transcription. We identify the genes with the numbers $1, \ldots, n$, with a plus or minus sign to indicate such direction. Their order will be represented by a *signed permutation* of $\{\pm 1, \ldots, \pm n\}$, that is a permutation π of $\{\pm 1, \ldots, \pm n\}$ such that $\pi[-i] = -\pi[i]$, where $\pi[i]$ denotes the i^{th} element in π. In the following, we indicate the sign of an element in a permutation only when it is minus.

The *reversal* of the interval $[i, j] \subseteq [1, n]$ $(i < j)$ is the signed permutation $\rho = 1, \ldots, i, -j, \ldots, -(i+1), j+1, \ldots, n$. Note that $\pi\rho$ is the permutation obtained from π by reversing the order and flipping the signs of the elements in the interval. If $\rho_1, \ldots, \rho_k$ is a sequence of reversals, we say that it sorts a permutation π if $\pi\rho_1 \cdots \rho_k = Id$ (Id is the all positive identity permutation $1, \ldots, n$). We denote by $d(\pi)$ the number of reversals in a minimum size sequence sorting π.

* Work supported by the French program bioinformatique Inter-EPST 2002 "Algorithms for Modelling and Inference Problems in Molecular Biology".

S.C. Sahinalp et al. (Eds.): CPM 2004, LNCS 3109, pp. 1–13, 2004.

The problem of sorting by reversals has been the subject of an extensive literature. For a complete survey, see [2]. The first polynomial algorithm was given by Hannenhalli and Pevzner [4], and ran in $O(n^4)$. After many subsequent improvements, the currently fastest algorithms are those of Kaplan, Shamir and Tarjan [5] running in $O(n^2)$, a linear algorithm for computing $d(\pi)$ only (it does not give the sequence of reversals) by Bader, Moret and Yan [1], and an $O(n\sqrt{n \log n})$ random algorithm by Kaplan and Verbin [6] which gives most of the time an optimal sequence of reversals, but fails on some permutations with very high probability. A reversal ρ is said to be *safe* for a permutation π if $d(\pi\rho) = d(\pi) - 1$. The bottleneck for all existing exact algorithms is the detection of safe reversals. Many techniques were invented to address this problem, but none has a better time complexity than linear, immediately implying a quadratic complexity for the whole method. In a recent paper [7], Ozery-Flato and Shamir compiled and compared the best algorithms, and wrote that: "A central question in the study of genome rearrangements is whether one can obtain a subquadratic algorithm for sorting by reversals".

In this paper, we give a positive answer to Ozery-Flato and Shamir's question. A good knowledge of the Hannenhalli-Pevzner theory and of the data structure used by Kaplan-Verbin is assumed.

In the next section, we briefly describe the usual tools (in particular the omnipresent breakpoint graph) for dealing with permutations and reversals. We mention some operations for sorting by reversals without giving any details (concerning, for instance, hurdle detection and clearing for which linear methods exist). The aim of this paper is to deal only with the most costly part of the method, that is sorting a permutation without hurdles. We introduce a new and elegant way of transforming the breakpoint graph of a permutation π by applying reversals either on π *or on its inverse permutation* π^{-1}. In section 3, we describe the method to optimally sort by reversals. With any classical data structure, the time complexity of this algorithm is $O(n^2)$, but even in this case it presents a special interest because it bypasses the detection of safe reversals which is considered as the most costly operation. Then in the last section, we indicate the data structure used to achieve subquadratic time complexity.

2 Mathematical Tools

To simplify exposition, we require that the first and last elements of a permutation remain unchanged. We therefore adopt the usual transformation which consists in adding the elements 0 and $n+1$ to $\{1, \ldots, n\}$, with $\pi[0] = 0$, and $\pi[n+1] = n+1$. The obtained permutation is called an *augmented permutation.* In this paper, all permutations are augmented, and we omit to mention it from now on. The inverse permutation π^{-1} of π is the (signed) permutation such that $\pi\pi^{-1} = Id$.

2.1 Breakpoint Graphs for a Permutation and Its Inverse

The *breakpoint graph* $BG(\pi)$ of a permutation π is a graph with vertex set V defined as follows: for each integer i in $\{1, \ldots, n\}$, let i^a (the *arrival*) and i^d

(the *departure*) be two vertices in V; add to V the two vertices 0^d and $(n+1)^a$. Observe that all vertex labels are non negative numbers. For simplicity and to avoid having to use absolute values, we may later refer to vertex $(-i)^x$ (for $x = a$ or d). This will be the same as vertex i^x. The edge set E of $BG(\pi)$ is the union of two perfect matchings denoted by R, the *reality edges* and D, the *dream edges* (in the literature, reality and dream edges are sometimes called reality and desire edges, or, in a more prosaic way, black and gray edges):

- R contains the edges $(\pi[i])^d(\pi[i+1])^a$ for all $i \in \{0, \ldots, n\}$;
- D contains an edge for all $i \in \{0, \ldots, n\}$, from i^d if $\pi^{-1}[i]$ is positive, from i^a if $\pi^{-1}[i]$ is negative, to $(i+1)^a$ if $\pi^{-1}[i+1]$ is positive, and to $(i+1)^d$ if $\pi^{-1}[i+1]$ is negative.

The reality edges define the permutation π (what you have), and the dream edges define Id (what you want to have). Reality edges always go from a departure to an arrival, but in dreams, everything can happen. An example of a breakpoint graph for a permutation is given in Figure 1.

To avoid case checking, in the notation of an edge, the mention of departures and arrivals may be omitted. For instance, $i(i+1)$ is a dream edge, indicating nothing as concerns the signs of $\pi^{-1}[i]$ and $\pi^{-1}[i+1]$.

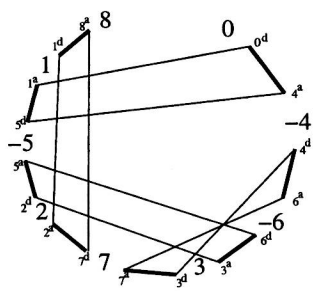


Fig. 1. The breakpoint graph of the permutation $0\ -4\ -6\ 3\ 7\ 2\ -5\ 1\ 8$. The bold edges are reality edges, and the thin ones are dream edges. The permutation should be read clockwise from 0 to $n+1$. This circular representation makes the cycles in the graph more visible. The edges that cross in the drawing correspond to crossing edges according to the definition.

A reality edge $(\pi[i])^d(\pi[i+1])^a$ is *oriented* if $\pi[i]$ and $\pi[i+1]$ have opposite signs, and *unoriented* otherwise. A dream edge $i(i+1)$ is *oriented* if $\pi^{-1}[i]$ and $\pi^{-1}[i+1]$ have opposite signs (that is, if the edge joins two departures or two arrivals), and *unoriented* otherwise. In the example of Figure 1, $(0,4)$, $(6,3)$, $(2,5)$ and $(5,1)$ are oriented reality edges, while $(3,4)$, $(6,7)$ are oriented dream edges.

To every dream edge $i(i+1)$, we associate the interval $[|\pi^{-1}[i]|, |\pi^{-1}[i+1]|]$ (or $[|\pi^{-1}[i+1]|, |\pi^{-1}[i]|]$ if $|\pi^{-1}[i]| > |\pi^{-1}[i+1]|$). Two dream edges are said to

cross if their associated intervals intersect but one is not contained in the other. Only dream edges may cross in a breakpoint graph.

Dream and reality edges are trivially and uniquely decomposed into cycles (the sets of both types of edges are perfect matchings of the vertices). By the cycles of a permutation π, we mean the cycles of $R \cup D$ in $BG(\pi)$. We call the *size* of a cycle the number of dream edges it contains (it is half the usual length of a cycle). Two cycles are said to *cross* if two of their edges cross.

A *component* $\mathcal{C}$ of $BG(\pi)$ is an inclusionwise minimal subset of its cycles, such that no cycle of $\mathcal{C}$ crosses a cycle outside $\mathcal{C}$. A component is said to be *oriented* if it contains a cycle with an oriented edge, and *unoriented* otherwise. A *hurdle* is a special type of unoriented component. We do not define it more precisely, since we deal only with permutations without unoriented components, therefore without hurdles. See for example [5] for a complete description of what a hurdle is, and how to cope with hurdles when there are some. In the example of Figure 1, there is a single oriented component.

The following operation establishes the correspondence between dream and reality in $BG(\pi)$ and $BG(\pi^{-1})$. Let $(BG(\pi))^{-1}$ be the graph resulting from applying the following transformations to $BG(\pi)$:

- change each vertex label i^a into i^d and i^d into i^a whenever $\pi^{-1}[i]$ is negative;
- change each vertex label $(\pi[i])^a$ into i^a, and $(\pi[i])^d$ into i^d;
- change dream into reality and reality into dream.

The result of such a transformation applied to the example of Figure 1 is given in Figure 2.

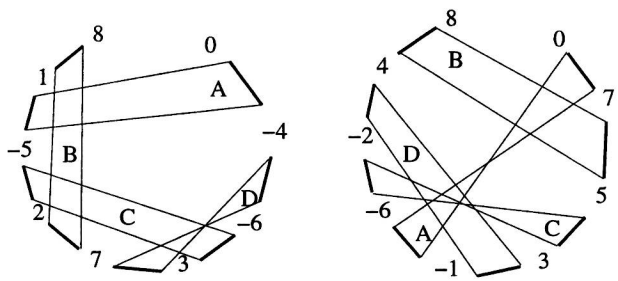


Fig. 2. A breakpoint graph and its inverse. The correspondence of the cycles in shown.

Lemma 1. $(BG(\pi))^{-1} = BG(\pi^{-1})$.

Proof. By definition, $(BG(\pi))^{-1}$ and $BG(\pi^{-1})$ have the same vertex set. There is a reality edge $(\pi[i])^d(\pi[i+1])^a$ in $BG(\pi)$ for all $i \in \{0, \ldots, n\}$. In $(BG(\pi))^{-1}$, it becomes a dream edge from i^d if $\pi[i]$ is positive or from i^a if $\pi[i]$ is negative, to $(i+1)^d$ if $\pi[i+1]$ is positive or to $(i+1)^a$ if $\pi[i+1]$ is negative. This corresponds