Lecture Notes in Computer Science 7353

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
Alfred Kobsa
University of California, Irvine, CA, 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
TU Dortmund University, Germany
Madhu Sudan
Microsoft Research, Cambridge, MA, USA
Demetri Terzopoulos
University of California, Los Angeles, CA, USA
Doug Tygar
University of California, Berkeley, CA, USA
Gerhard Weikum
Max Planck Institute for Informatics, Saarbruecken, Germany

Edward A. Hirsch   Juhani Karhumäki
Arto Lepistö   Michail Prilutskii (Eds.)

Computer Science – Theory and Applications

7th International Computer Science Symposium
in Russia, CSR 2012
Nizhny Novgorod, Russia, July 3-7, 2012
Proceedings

Springer

Volume Editors

Edward A. Hirsch
Steklov Institute of Mathematics at St.Petersburg
191023 St. Petersburg, Russia
E-mail: hirsch@pdmi.ras.ru

Juhani Karhumäki
Arto Lepistö
University of Turku, 20014 Turku, Finland
E-mail: {karhumak, alepisto}@utu.fi

Michail Prilutskii
Lobachevsky State University of Nizhny Novgorod
603950 Nizhny Novgorod, Russia
E-mail: pril@iani.unn.ru

ISSN 0302-9743
e-ISSN 1611-3349
ISBN 978-3-642-30641-9
e-ISBN 978-3-642-30642-6
DOI 10.1007/978-3-642-30642-6
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2012938206

CR Subject Classification (1998): F.2, F.3, E.3, G.2, F.1, F.4

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

© Springer-Verlag Berlin Heidelberg 2012
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.
The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

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

Preface

The 7th International Computer Science Symposium in Russia (CSR 2012) was held during July 3–7, 2012, in Nizhny Novgorod, Russia, hosted by the Lobachevsky State University. It was the seventh event in the series of regular international meetings, following CSR 2006 in St. Petersburg, CSR 2007 in Ekaterinburg, CSR 2008 in Moscow, CSR 2009 in Novosibirsk, CSR 2010 in Kazan and CSR 2011 in St. Petersburg. CSR 2012 was one of the events of the Alan Turing Year 2012. It took place under the auspices of European Association of Theoretial Computer Science (EATCS).

The opening lecture was given by Vijay Vazirani and the special Turing lecture by Yuri Matiyasevich. Other invited plenary lecturers were Lev Beklemishev, Mikołaj Bojańczyk, Julien Cassaigne, Piotr Indyk, Jaroslav Nešetřil and Pavel Pevzner.

This volume contains the accepted papers and abstracts of the invited papers. The scope of the topics of the symposium was broad and covered substantial parts of theoretical computer science and its applications. We received 66 submissions. Of these, the international Program Committee selected 28 for presentation at the conference, using the EasyChair system. A special issue of *Theory of Computing Systems* consisting of selected papers will be published.

As usual, Yandex provided the Best Paper Awards; the recipients of these were selected by the Program Committee:

– Best paper award: C. Kapoutsis and G. Pighizzini, "Two-Way Automata Characterizations of L/poly Versus NL"
– Best student paper: E. Demenkov, "A Lower Bound on Circuit Complexity of Vector Function in U_2"

The following satellite events were co-located with CSR 2012:

– The Third Workshop on Program Semantics, Specification and Verification: Theory and Applications (PSSV 2012)
– Workshop on Current Trends in Cryptology (CTCrypt 2012)

We are grateful to our sponsors:

– Dynasty Foundation
– Finnish Academy of Science and Letters, mathematics funding
– Microsoft Research
– Russian Foundation for Basic Research
– Yandex

Last but not least, we thank Springer for the smooth cooperation, and the local organizers, in particular, Roman Strongin (President of State University of Nizhni Novgorod) for their great help and support.

April 2012

Edward Hirsch
Juhani Karhumäki
Arto Lepistö
Michail Prilutskii

Organization

CSR 2012 intended to reflect the broad scope of international cooperation in computer science. It was the seventh conference in a series of regular events started with CSR 2006 in St. Petersburg. The topics covered vary from year to year, but in general try to cover as much of the field of contemporary computer science as possible.

CSR 2012 was included in schedule of the Alan Turing Year events.

Conference Chair

Michail Prilutskii	Lobachevsky State University of Nizhni Novgorod, Russia

Program Chair

Juhani Karhumäki	University of Turku, Finland

Program Committee

Lev Afraimovich	Lobachevsky State University of Nizhni Novgorod, Russia
Susanne Albers	HU Berlin, Germany
Andris Ambainis	University of Latvia, Latvia
Alberto Bertoni	University of Milan, Italy
Bruno Durand	Université Montpellier 2, France
Edward A. Hirsch	Steklov Institute of Mathematics at St. Petersburg, Russia
Juraj Hromkovic	ETH Zürich, Switzerland
Kazuo Iwama	Kyoto University, Japan
Juhani Karhumäki	University of Turku, Finland
Markus Lohrey	University of Leipzig, Germany
Ernst Mayr	TU München, Germany
Ilya Mironov	Microsoft Research Silicon Valley, USA
Anca Muscholl	Université Bordeaux, France
Alexander Okhotin	University of Turku, Finland
Alexander Razborov	University of Chicago, USA
Wojciech Rytter	Warsaw University, Poland
Jirí Sgall	Charles University, Czech Republic
Alexander Shen	Université Montpellier 2, France
Arseny Shur	Ural State University, Russia
Wolfgang Thomas	RWTH Aachen University, Germany
Nikolai Vereshchagin	Moscow State University, Russia

Gerhard Woeginger	Eindhoven University of Technology, The Netherlands
Mikhail Vyalyi	Dorodnicyn Computing Centre of RAS, Russia

Steering Committee

Volker Diekert	University of Stuttgart, Germany
Anna Frid	Sobolev Institute of Mathematics, Russia
Edward A. Hirsch	Steklov Institute of Mathematics at St. Petersburg, Russia
Juhani Karhumäki	University of Turku, Finland
Mikhail Volkov	Ural State University, Russia

Organizing Committee

Michail Prilutskii	Lobachevsky State University of Nizhni Novgorod, Russia
Victor Gergel	Lobachevsky State University of Nizhni Novgorod, Russia
Vladimir Shvetsov	Lobachevsky State University of Nizhni Novgorod, Russia
Lev Afraimovich	Lobachevsky State University of Nizhni Novgorod, Russia
Alexander Gorylev	Lobachevsky State University of Nizhni Novgorod, Russia
Vadim Saigin	Lobachevsky State University of Nizhni Novgorod, Russia
Yana Safonova	Lobachevsky State University of Nizhni Novgorod, Russia

Referees

S. Aguzzoli
A. Antoniadis
L. Babai
V. Barany
N. Benes
C. Blundo
M. Bodirsky
H.J. Boeckenhauer
P. Bonizzoni
D. Bruschi
A. Bulatov
J.Y. Cai
G. Calinescu
A. Carayol
M. Chrobak
W. Czerwinski
B. Das
O. Dubois
R. Ehlers
M. Fellows
T. Fernique
E. Formenti
P. Gawrychowski
K. Georgatos
A. Grigoriev
E. Grädel
P. Guillon
V. Gusev
L. Hellerstein
J. Hitchcock
D. Itsykson
E. Jeandel
A. Jėz
J. Johannsen
J. Kari
A. Kartzow

D. Kirsten
D. Komm
G. Kortsarz
S. Krug
J. Krugel
P. Krysta
A. Kulikov
J. Lahtonen
M. Lange
M. Lauria
T. Lehtinen
P. Lenzner
L. Lyaudet
C. Löding
D. Manlove
C. Mereghetti
I. Mezhirov
S. Miguet
I. Mironov
S. Miyazaki
H. Morizumi
D. Musatov
T. Mömke
S. Nikolenko
J. Olschewski
B.M. Paola
D. Pardubska
P. Parys
I. Petre
G. Pighizzini
C. Pinkau
V. Podolskii
Y. Pritykin
M. Raskin
K. Reinhardt
A. Reuss
H. Rivano
A. Roman
A. Romashchenko
A. Salomaa
K. Salomaa
H. Schnoor
C. Shah
V. Shpilrain
S. Smyczynski
F. Spieksma
A. Sprock
R. Stefanec
M. Steinova
P. Tesson
D. Thilikos
W. Tyczynski
H. Täubig
T. Walen
R.F.C. Walters
J. Weihmann
D. West
A. Wolff
X. Wu
M. Zimand

Sponsoring Institutions

Dynasty Foundation, Russia
Finnish Academy of Science and Letters, Finland
Microsoft Research, USA
Russian Foundation for Basic Research, Russia
Yandex, Russia

Turing Talk

Alan Turing and Number Theory

Yuri V. Matiyasevich

Steklov Institute of Mathematics (POMI)
St.Petersburg, Russia 191023
bengio@google.com

Beside well-known revolutionary contributions, Alan Turing had a number of significant results in "traditional" mathematics. In particular he was very much interested in the famous Riemann Hypothesis. This hypothesis, stated by Berhard Riemann in 1859 and included by David Hilbert in his 8th problem in 1990, still remains open, being now one of the Millennium Problems. The Riemann Hypothesis predicts positions of zeros of so called zeta function, and Alan Turing developed a rigorous method for verifying the Hypothesis for the initial zeros. He also invented a machine for calculating the values of the zeta function. In contrast to celebrated imaginable Turing machines, Turing started to implement this machine but never finished because of the War.

Plenary Invited Talks

Challenges in Comparative Genomics: From Biological Problems to Combinatorial Algorithms (and back)*

Max A. Alekseyev[1,3] and Pavel Pevzner[2,3]

[1] University of South Carolina, Columbia, SC, U.S.A.
[2] Department of Computer Science and Engineering
University of California in San Diego, La Jolla, CA, U.S.A.
[3] Academic University, St. Petersburg, Russia

Recent large-scale sequencing projects fueled the comparative genomics studies and heightened the need for algorithms to extract valuable information about genetic and phylogenomic variations. Since the most dramatic genomic changes are caused by genome rearrangement events (which shuffle genomic material), it becomes extremely important to understand their mechanism and reconstruct the sequence of such events (evolutionary history) between genomes of interest.

In this expository talk I shall describe several controversial and hotly debated topics in evolutionary biology (chromosome breakage models, mammalian phylogenomics, prediction of future rearrangements) and formulate related combinatorial challenges (rearrangement and breakpoint re-use analysis, ancestral genomes reconstruction problem). I shall further present recent theoretic and algorithmic advances in addressing these challenges and their biological implications.

* This work was partially supported by the Government of the Russian Federation (grant 11.G34.31.0018).

DKAL: A Distributed Knowledge Authorization Language and Its Logic of Information

Lev Beklemishev

Steklov Institute of Mathematics, Moscow, Russia

This talk is a report on the DKAL project developed at Microsoft Research by Yuri Gurevich and his collaborators (Itay Neeman, Michal Moskal, Andreas Blass, Guido di Caso, and the others). I will survey some of the main features of DKAL and discuss the underlying information logics.

With the advent of cloud computing, the role of formal policies grows. The personnel of brick-and-mortar businesses often exercise their judgments; all that should be replaced with formal policies when businesses move to the cloud. The logic-based policy language DKAL (Distributed Knowledge Authorization Language) [4, 5, 3] was developed with such applications in mind. The feature that distinguishes DKAL from most preceding logic-based policy languages is that it is explicitly geared toward federated scenarios (with no central authority) where trust may be in short supply.

The world of DKAL consists of communicating principals computing their own knowledge in their own states. They communicate *infons*, pieces of information, and reason in terms of infons. In [5], the original developers of DKAL distilled the basic features of the logic of infons and introduced infon logic **qI** that is an extension of the $\{\to, \wedge\}$ fragment **I** of intuitionistic logic with quotation modalities $p\,\texttt{said}\,\phi$ and $p\,\texttt{implied}\,\phi$. In addition they isolated a *primal fragment* **qP** of **qI** which is very efficient and yet sufficiently expressive for many purposes. In the case of bounded quotation depth, the derivation problem for **qP** is solvable in linear time. In particular, the quotation-free fragment **P** of **qP** is linear time in that sense.

The continuing development of DKAL (whose current implementation is found at [1]) requires further investigation of the logic of infons. In [2], we extend the four logics of [5] with one or both of disjunction and negation, and we determine the complexities of the extended logics. We provide a semantics for the extension $\mathbf{P}[\vee]$ of **P** that we call *quasi-boolean*. This allows us to give efficient mutual translations between $\mathbf{P}[\vee]$ and classical propositional logic as well as an embedding of the appropriate classical modal logic into $\mathbf{qP}[\vee]$. On the proof-theoretic side we develop cut-free Gentzen-style sequent calculi for the extensions of primal logic **P** with some or all of disjunction, negation and quotations.

References

1. DKAL at CodePlex: `http://dkal.codeplex.com/`
2. Beklemishev, L., Gurevich, Y.: Propositional primal logic with disjunction. Technical Report MSR-TR-2011-35, Microsoft Research (March 2011); To appear in Journal on Logic and Computation
3. Blass, A., Gurevich, Y., Moskal, M., Neeman, I.: Evidential authorization. In: Nanz, S. (ed.) The Future of Software Engineering, pp. 77–99. Springer (2011)
4. Gurevich, Y., Neeman, I.: DKAL: Distributed-Knowledge Authorization Language. In: Proc. of CSF 2008, pp. 149–162. IEEE Computer Society (2008)
5. Gurevich, Y., Neeman, I.: DKAL 2 — A Simplified and Improved Authorization Language. Technical Report MSR-TR-2009-11, Microsoft Research (February 2009)

Infinite Sets That Are Finite Up to Permutations

Mikołaj Bojańczyk*

University of Warsaw

Fraenkel-Mostowski sets are a variant of set theory, where sets can contain atoms. The existence of atoms is postulated as an axiom. The key role in the theory of Fraenkel-Mostowski sets is played by permutations of atoms. For instance, if a, b, c, d are atoms, then the sets

$$\{a, \{a, b, c\}, \{a, c\}\} \qquad \{b, \{b, c, d\}, \{b, d\}\}$$

are equal up to permutation of atoms.

Fraenkel-Mostowski sets were rediscovered for the computer science community, by Gabbay and Pitts [3]. It turns out that atoms are a good way of describing variable names in programs or logical formulas, and the permutations of atoms are a good way of describing renaming of variables. Fraenkel-Mostowski sets are now widely studied in the semantics community, under the name of nominal sets (the name is so chosen because atoms describe variables names).

This talk is about a new use application of these sets in computer science. (We use the name nominal sets.) The new application has its roots in database theory, but it touches other fields, such as verification or automata theory. The motivation in database theory is that atoms can be used as an abstraction for data values, which can appear in a relational database or in an XML document. Like in nominal sets, there are infinitely many possible atoms (or data values), but properties of databases (relational or XML) are closed under permutations of atoms. Atoms can also be used to model sources of infinite data in other applications, such as software verification, where an atom can represent a pointer or the contents of an array cell.

Nominal sets are a good abstraction for infinite systems because they have a different, more relaxed, notion of finiteness. A nominal sets is considered finite if it is finite up to permutation of atoms. (We call such a set orbit-finite.) For instance, the set of atoms is orbit-finite, because every atom can be mapped to every other atom by a permutation. Likewise, the set of pairs of atoms has two elements up to permutation, namely (a, a) and (a, b) for $a \neq b$. Another example is the set of λ-terms which represents the identity, up to α-conversion:

$$\{\lambda a.a : a \text{ is an atom}\}.$$

Yet another example concerns automata with registers for storing atoms [4]: up to permutation, there are finitely many configurations of an automaton which has four control states and three registers which can store atoms.

* Author supported by ERC Starting Grant "Sosna".

The language of nominal sets is so robust that one can meaningfully restate all of the results in a textbook on automata theory, replacing sets by nominal sets and finite sets by orbit-finite sets, see [2] for examples. Some of the restated theorems are true, some are not. Results that work in the nominal setting include the Myhill-Nerode theorem, or the equivalence of pushdown automata with context free grammars. Results that fail in the nominal setting include all results which depend on the subset construction, such as determinization of finite automata, or equivalence of two-way and one-way finite automata.

Perhaps most importantly, the theory of computability still works in the nominal setting. More specifically, one can design a programming language which manipulates orbit-finite nominal sets, just like other programming languages manipulate lists or trees [1]. (This programming language is not a violation of the Church-Turing thesis – after some translation, the programming language can be executed on a normal computer.)

References

1. Bojańczyk, M., Braud, L., Klin, B., Lasota, S.: Towards nominal computation. In: POPL, pp. 401–412 (2012)
2. Bojańczyk, M., Klin, B., Lasota, S.: Automata with group actions. In: LICS, pp. 355–364 (2011)
3. Gabbay, M., Pitts, A.M.: A new approach to abstract syntax with variable binding. Formal Asp. Comput. 13(3-5), 341–363 (2002)
4. Kaminski, M., Francez, N.: Finite-memory automata. Theor. Comput. Sci. 134(2), 329–363 (1994)

Dynamics of Rauzy Graphs for Low-Complexity Words

Julien Cassaigne

Institut de mathématiques de Luminy
case 907, 163, avenue de Luminy, 13288 Marseille Cedex 9
cassaigne@iml.univ-mrs.fr

Let $u \in A^{\mathbb{N}}$ be an infinite word. The *factor complexity* of u is the function $p\colon \mathbb{N} \to \mathbb{N}$ defined by: $p(n)$ is the number of words of length n occurring in u (*factors* of u). Morse and Hedlund [5] proved that $p(n) \geq n+1$ for non-eventually-periodic words. Words for which $p(n) = n+1$ are called Sturmian words. Words for which $p(n) = n+c$ for some constant c can be deduced from them [3]. We are interested in words "just above" this, roughly $n+1 \leq p(n) \leq 2n$. Let $\alpha_u = \liminf \frac{p(n)}{n}$ and $\beta_u = \limsup \frac{p(n)}{n}$, and $\Omega = \{(\alpha_u, \beta_u)\colon u \in A^{\mathbb{N}}\} \subseteq (\mathbb{R}^+ \cup \{+\infty\})^2$. Then the general problem, essentially open, is: what is the structure of Ω?

Heinis proved [4] that $\beta - \alpha \geq \frac{(2-\alpha)(\alpha-1)}{\alpha}$. In particular, $1 < \alpha = \beta < 2$ is impossible.[1] Aberkane [1] contructed a sequence of points of Ω converging to $(1,1)$; on the other hand, Turki [6] proved that $(\frac{3}{2}, \frac{5}{3})$ is an isolated point in Ω.

The main tool to study these words is the sequence of *Rauzy graphs*: Γ_n is the directed graph with vertices $L_n(u)$ (the factors of length n of u) and edges $L_{n+1}(u)$, with an edge from x to y labelled with z if and only if $z \in xA \cap Ay$. For Sturmian words, only two shapes of graphs are possible. For recurrent words with $p(n) \leq \frac{4}{3}n+1$, two new shapes appear. The language of such a word is then defined by a path in the *graph of shapes*, that controls how its Rauzy graphs evolve, and from which (α, β) and other properties may be deduced. The path also provides an *s-adic representation* (infinite composition of substitutions), which can be viewed as a generalized continued fraction expansion.

References

1. Aberkane, A.: Words whose complexity satisfies $\lim \frac{p(n)}{n} = 1$. Theoret. Comput. Sci. 307, 31–46 (2003)
2. Cassaigne, J., Nicolas, F.: Factor complexity. In: Berthé, V., Rigo, M. (eds.) Combinatorics, Automata and Number Theory, pp. 163–247. Cambridge University Press (2010)
3. Coven, E.M.: Sequences with minimal block growth II. Math. Systems Theory 8, 376–382 (1975)

[1] This is generalized in [2]: if $\lim \frac{p(n)}{n}$ exists, then it must be an integer.

4. Heinis, A.: The $P(n)/n$ function for bi-infinite words. Theoret. Comput. Sci. 273, 35–46 (2002)
5. Morse, M., Hedlund, G.A.: Symbolic Dynamics II. Sturmian trajectories. American J. Math. 62, 1–42 (1940)
6. Turki, R.: An isolated point in the Heinis spectrum (submitted manuscript, 2012)

Faster Algorithms for Sparse Fourier Transform

Piotr Indyk

MIT Computer Science and Artificial Intelligence Lab
32 Vassar Street
Cambridge, Massachusetts 02139

The Fast Fourier Transform (FFT) is one of the most fundamental numerical algorithms. It computes the Discrete Fourier Transform (DFT) of an n-dimensional signal in $\mathcal{O}(nlogn)$ time. The algorithm plays a key role in many areas.

In many applications (e.g., audio, image or video compression), most of the Fourier coefficients of a signal are "small" or equal to zero, i.e., the output of the transform is (approximately) sparse. In this case, there are algorithms that enable computing the non-zero coefficients faster than the FFT. However, in practice, the exponents in the runtime of these algorithms and their complex structure have limited their applicability to only very sparse signals.

In this talk, I will describe a new set of algorithms for sparse Fourier Transform. Their key feature is simplicity, which leads to efficient running time with low overhead, both in theory and in practice. In particular, one of the algorithms achieves a runtime of $\mathcal{O}(klogn)$, where k is the number of non-zero Fourier coefficients of the signal. This improves over the runtime of the FFT for any $k = \wr(n)$.

Joint work with Haitham Hassanieh, Dina Katabi and Eric Price.

Algorithms, Dichotomy and Statistics for Geometric and Sparse Graphs

Jaroslav Nešetřil

Department of Applied Mathematics
Faculty of Mathematics and Physics
Charles University, Prague

Sparse graphs present a problem: on the one side we often have better algorithms on the other side their structure and the lack of proper models makes the difficult to study. In this talk we present a new dichotomy "somewhere dense vs nowhere dense" classes a show the robustness and algorithmic relevance of this dichotomy. Particularly, we determine the asymptotic logarithmic density of subgraphs of large geometric graphs (and, more generally, of certain classes of sparse graphs). This leads to a unified approach to graph limits for both sparse and dense classes.

Joint work with Patrice Ossona de Mendez, Paris.

Table of Contents

Opening Talk

Full Papers