Third
Russian-Finnish Symposium
on Discrete Mathematics

Третий
Российско-Финский симпозиум
по дискретной математике

September 15-18, 2014
Petrozavodsk, Russia

EXTENDED ABSTRACTS

Научный редактор / Editor: В. В. Мазалов / V. V. Mazalov
Отв. за выпуск / Resp. for volume: А. А. Ивашко / A. A. Ivashko

Main organizers

Russian Academy of Sciences
Steklov Mathematical Institute of RAS, St. Petersburg
Institute of Applied Mathematical Research of Karelian Research Centre of RAS, Petrozavodsk
University of Turku

Scientific committee

co-chair: Juhani Karhumäki (Turku)
co-chair: Yuri Matiyasevich (St.Petersburg)
Vesa Halava (Turku)
Luca Zamboni (Turku and Lyon)
Svetlana Puzynina (Turku)
Tero Laihonen (Turku)
Michail Volkov (Ekaterinburg)
Il'ya Ponomarenko (St.Petersburg)
Dmitry Karpov (St.Petersburg)
Igor Lysënok (Moscow)
Vladimir Mazalov (Petrozavodsk)
Yuri Pavlov (Petrozavodsk)

Organizing committee
IAMR KarRC RAS, Petrozavodsk, Russia

chair: Vladimir Mazalov
secretary: Anna Ivashko
Yuri Pavlov
Evsey Morozov
Anna Rettieva
Julia Chirkova
Evgeny Ivashko
Alexey Kondratev

Foreword

The present volume contains full papers and extended abstracts accepted for the Third Russian Finnish Symposium on Discrete Mathematics held in the Institute of Applied Mathematical Research KarRC RAS, Petrozavodsk, Russia, September 15-18, 2014.

Topics of the symposium include:

- Combinatorics on Words
- Graph Theory
- Automata Theory
- Tilings
- Decidability Problems
- Random Graphs
- Networking Games

and related areas.

Acknowledgements

The Program and Organizing Committee thank the Russian Foundation for Basic Research (project 14-01-20148-r) for the financial support.

On behalf of the Organization Committee

Yuri Matiyasevich
Juhani Karhumäki
Vladimir Mazalov

Contents

Invited lectures

Contributed talks

Invited lectures

Decomposition of a Language of Factors into Sets of Bounded Complexity*

Julien Cassaigne

Institut de mathématiques de Marseille, Marseille, France

We explore a new hierarchy of classes of languages and infinite words and its connection with complexity classes. Namely, we say that a language belongs to class L_k if it is a subset of the catenation of k languages S_1, ..., S_k , where the number of words of length n in each S_i is bounded by a constant. An infinite word belongs to class W_k if its language of factors is in L_k. We focus on the relations between classes W_k and the factor complexity of infinite words. In particular, we prove that class W_2 coincides with the class of infinite words of linear complexity, but there is no such simple characterization for other W_k and for L_k.

*joint work with Anna Frid, Svetlana Puzynina, and Luca Q. Zamboni

Piecewise Affine Functions, Sturmian Sequences and Aperiodic Tilings

Jarkko Kari

Department of Mathematics, University of Turku
Turku, Finland

We discuss constructions of Wang tilings that simulate iterations of piecewise rational affine functions. Our simulations use representations of real numbers as Sturmian sequences. The method provides smallest known aperiodic Wang tile sets, and leads to a simple proof of the undecidability of the domino problem. One also easily establishes analogous undecidability results about tilings of the hyperbolic plane and of Baumslag-Solitar groups.

Combinatorial Geometry and Coding Theory*

Andrei M. Raigorodskii

Moscow State University, Mechanics and Mathematics Faculty,
Department of Mathematical Statistics and Random Processes,
Moscow, Russia
Moscow Institute of Physics and Technology, Faculty of Innovations and
High Technology, Department of Discrete Mathematics
Moscow Region, Dolgoprudny, Russia
Yandex Division of Theoretical and Applied Research
Moscow, Russia

In our talk, we will be mainly concerned with subjects that lie on the edge of combinatorial geometry and coding theory. As for combinatorial geometry, the two problems, which are most important and closely connected to each other, are the Nelson–Hadwiger problem on finding the space chromatic number and the Borsuk problem on partitioning sets in spaces into parts of smaller diameter. It turns out that the strongest existing results for both problems are obtained with the help of systems of (0,1)-vectors with forbidden scalar products and their generalizations onto arbibtrary finite systems of vectors in $\mathbb{Z}^n$ with similar restrictions. Here we naturally come to some classical as well as completely novel concepts and problems of coding theory.

In the lecture, we will first give a survey of results for the Nelson–Hadwiger and Borsuk problems. Then we shall proceed to discussing related coding-theoretic questions concerning (0,1)-vectors and other integer point systems. We will also speak about probabilistic versions of

*This work is done under the financial support of the following grants: the grant 12-01-00683 of Russian Foundation for Basic Research, the grant MD-6277.2013.1 of the Russian President, the grant NSh-2519.2012.1 supporting Leading scientific schools of Russia.

these questions, by defining random subgraphs of some important "distance graphs" and treating of their properties. In particular, we will exhibit very recent results on the stability of some classical extremal values in coding theory. Finally, we shall formulate conjectures and open questions.

On Polynomials of Birch-Chowla-Hall-Schinzel-Davenport-Stothers-Zannier-Beukers-Stewart and Weighted Plane Trees

Alexander Zvonkin

University of Bordeaux, France

In 1965, Birch, Chowla, Hall, and Schinzel, motivated by certain problems in number theory, raised the following question: given two complex polynomials A and B, what could be the least possible degree of A^3-B^2 (if this difference is not identically zero)? They conjectured a lower bound, and also supposed that this bound was sharp. The bound itself was proved the same year by Davenport, its sharpness was proved by Stothers in 1981. In 1995, Zannier generalized the problem and established both a lower bound and its sharpness for the degree of the difference of two polynomials with given multiplicities of their roots. In 2010, Beukers and Stewart returned once again to a particular case of the difference of the type A^p-B^q but this time they looked for polynomials with rational coefficients. In our joint work with Fedor Pakovich (University of Beer Sheva, Israel) we established a correspondence between the pairs of polynomials satisfying Zannier's bound, on the one hand, and bicolored plane trees whose edges are endowed with integral weights. If, for a given set of degrees of black and white vertices, the tree in question is unique, then the corresponding polynomials have rational coefficients. We give a complete classification of such trees, and have computed all the corresponding polynomials. We have also considered various combinatorial invariants of the Galois action on weighted trees. In particular, in a joint work with Nikolai Adrianov (Moscow University) we obtained a complete classification of the primitive monodromy groups of weighted trees.

Contributed talks

Approximation of Reset Thresholds with Greedy Algorithms

Dimitry S. Ananichev, Vladimir V. Gusev

Institute of Mathematics and Computer Science,
Ural Federal University, Ekaterinburg, Russia

Abstract

The problem of approximate computation of the reset thresholds of automata gained a lot of attention recently. We introduce a broad class of the algorithms and analyze approximation ratios of algorithms in this class. We present two different series of automata that reveal inherent limitations of greedy strategies for approximation of the reset thresholds.

1. Introduction

Let $\mathscr{A}$ be a deterministic finite automaton over the finite alphabet Σ with the set of states Q and the transition function δ. Automaton $\mathscr{A}$ is called *synchronizing* if there exist a word w and a state p such that for every state $q \in Q$ we have $\delta(q, w) = p$. Any such word w is called *reset* (or *synchronizing*) word for $\mathscr{A}$. The length of the shortest synchronizing word is called *reset threshold* ($\mathrm{rt}(\mathscr{A})$) of $\mathscr{A}$. Survey of the theory of synchronizing automata may be found in [6].

In the present note we deal with algorithmic aspects of the following problem: given an automaton $\mathscr{A}$, find its reset threshold. We will refer to this problem as `RT`. Decision version of `RT` is NP-complete [3], i.e. for a fixed k, determine whether the reset threshold of a given automaton

is at most k. The problem itself is $FP^{NP[log]}$-complete [5]. Due to this facts, we wonder whether the reset threshold of an automaton can be approximated well. Unfortunately [4], for a fixed constant C there is no $C \cdot \log(n)$-approximation algorithm for RT, where n is the number of states (unless $P = NP$). This statement remains true even if we restrict ourselves to two-letter automata [2]. At the same time, for every $k \geq 2$ there is a polynomial-time algorithm with approximation ratio $\lceil \frac{n-1}{k-1} \rceil$. Such approximation is the best among currently known algorithms. Thus, precise value of the best possible approximation ratio for RT remains unclear. In the present note we introduce a broad class of the algorithms and analyze approximation ratios of algorithms in this class. We present two different series of automata that reveal inherent limitations of greedy strategies for approximation of the reset thresholds.

2. Main results

We will write $|S|$ for the size of a set S and ε for the empty word. Arguably the most common type of algorithms for RT is the following one:

$SYNCH - SUBSET(k)$
Input: Synchronizing automaton $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$
1. $S = Q$
2. $u = \varepsilon$
3. Until $|S| == 1$ do
4. If $|S| \geq k$:
5. Choose a set of states $P \subseteq S$ s.t. $|P| = k$
6. Else:
7. $P = S$
8. Choose a word v s.t. $|\delta(P, v)| = 1$
9. $u = u \cdot v$
10. $S = \delta(S, v)$
11. Return length of u

For a particular algorithm of this type we need to fix the constant $k \geq 2$ and the way we make choices in the lines 5 and 8. It is an easy check that the latest value of the word u is a reset word for the automaton $\mathscr{A}$.

Performance of $SYNCH-SUBSET(k)$ vary drastically depending on the "choice-oracles" $\mathscr{O}_{set}$ and $\mathscr{O}_{word}$ in the lines 5 and 8, respectively. For example, if $\mathscr{O}_{word}$ always returns the shortest reset word of an automaton

$\mathscr{A}$ then the value returned by the algorithm is equal to the reset threshold of $\mathscr{A}$. At the same time, if $\mathscr{O}_{word}$ returns the α-power of the shortest reset word of $\mathscr{A}$ then approximation ratio of such algorithm is equal to α.

It is clear, that the presented oracles are not polynomial-time computable (otherwise we could solve RT in polynomial-time). The most common choice of "practical" oracles is based on a greedy strategy. We will denote the length of a word v by $|v|$. We say that $\mathscr{O}_{word}$ oracle is *greedy* if for any other word v' with the property $|\delta(P, v')| = 1$ we have $|v| \leq |v'|$. We say that $\mathscr{O}_{set}$ oracle is *greedy* if for any other subset P' and a word v' such that $|\delta(P', v')| = 1$ there is a word v such that $|\delta(P, v)| = 1$ and $|v| \leq |v'|$. Note, there are polynomial-time computable greedy oracles $\mathscr{O}_{set}$ and $\mathscr{O}_{word}$.

The analysis of $SYNCH-SUBSET(k)$ with a greedy $\mathscr{O}_{word}$ oracle is presented in [4]. The following simple lemma is given in the latter paper.

Lemma 1 *Approximation ratio of $SYNCH-SUBSET(k)$ with a greedy $\mathscr{O}_{word}$ oracle is at most $\lceil \frac{n-1}{k-1} \rceil$.*

The authors also presented an n-state automaton with $\lceil \frac{n-1}{k-1} \rceil$ letters for which approximation ratio of $SYNCH - SUBSET(k)$ with a particular greedy $\mathscr{O}_{word}$ oracle and a particular $\mathscr{O}_{set}$ oracle is equal to $\lceil \frac{n-1}{k-1} \rceil$. Thus, the bound in lemma 1 is tight. We present an easier example $\mathscr{A}_{n,k}$ with only two letters. The set of states of $\mathscr{A}_{n,k}$ is equal to the set of ordered pairs (α, β) such that $n - 1 \geq \alpha(k-1) + \beta$, $0 \leq \alpha$ and $0 \leq \beta < k - 1$. Additionally, we append a sink state $\boldsymbol{z}$. Note, the automaton $\mathscr{A}_{n,k}$ has n states. The alphabet of $\mathscr{A}_{n,k}$ is equal to $\{a, b\}$. The transition function δ is defined as follows:

$$\delta((\alpha, \beta), a) = \begin{cases} (\alpha - 1, \beta), \text{ if } \alpha > 0 \\ \boldsymbol{z}, \text{ otherwise} \end{cases} \qquad \delta((\alpha, \beta), b) = \boldsymbol{z}$$

Also, for every letter ℓ we define $\delta(\boldsymbol{z}, \ell) = \boldsymbol{z}$. The automaton $\mathscr{A}_{n,2}$ is presented on the fig. 1.

It is a straightforward check that for a particular choice of subsets by $\mathscr{O}_{set}$ and a particular greedy oracle $\mathscr{O}_{word}$ the value returned by $SYNCH-SUBSET(k)$ on input $\mathscr{A}_{n,k}$ is $\lceil \frac{n-1}{k-1} \rceil$-times larger than $\mathrm{rt}(\mathscr{A}_{n,k})$. We want to emphasize, that quite a natural oracle $\mathscr{O}_{word}$ suffice for this statement.

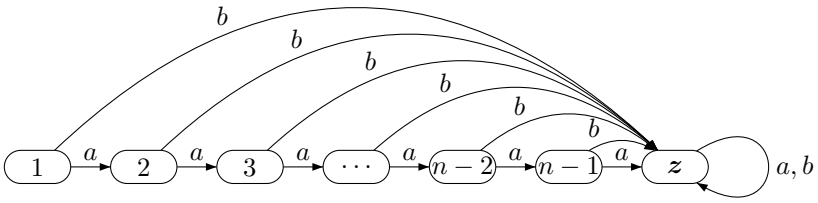

Figure 1: The automata $\mathscr{A}_{n,2}$

For example, the one that returns lexicographically the first word v with the property $|\delta(P, v)| = 1$.

The most important drawback of such example is a complete lack of foresight by $\mathscr{O}_{word}$. We can show that approximation ratio of $SYNCH - SUBSET(k)$ with an arbitrary greedy $\mathscr{O}_{word}$ oracle is at least $C \cdot n$ for some constant C, but we require greediness of $\mathscr{O}_{set}$. The following theorem is one of the main results of our note.

Theorem 1 *Approximation ratio of $SYNCH - SUBSET(k)$ with arbitrary greedy $\mathscr{O}_{word}$ and $\mathscr{O}_{set}$ oracles is at least $\frac{n}{6(k-1)}$.*

Proof. Automaton $\mathscr{B}_\ell$ is defined as in fig. 2. The shortest synchronizing word for $\mathscr{B}_n$ is $ab^{2\ell-1}a$ of length $2\ell + 1$.

Let $k = 2$. Then the greedy algorithm returns the word $a(b^{\ell+1}a)^{\ell-1}$ of length $1 + (\ell + 2)(\ell - 1) = \ell^2 + \ell - 1$. The number of states in $\mathscr{B}_\ell$ is equal to $n = 3\ell - 1$. Thus, approximation ratio

$$\frac{(\frac{n+1}{3})^2 + (\frac{n+1}{3}) - 1}{2(\frac{n+1}{3}) + 1} \sim \frac{n}{6}$$

Now let k be arbitrary. Note that for the set $S = \{\ell, \ell - 1, \ell - 2, ..., t\}$ with $t \in \{1, .., \ell - k + 1\}$ the shortest word v with the property $|S.v| \leq |S| - k + 1$ is $b^{\ell+k-1}a$. If $S = \{\ell, \ell - 1, \ell - 2, ..., t\}$ with $t \in \{\ell - k + 2, .., \ell - 1\}$ the shortest word v with the property $|S.v| = 1$ is $b^{2\ell-t}a$. Let $\ell - 1 = q(k - 1) + r$, where $r \in \{1, ..., k - 1\}$ and q is integer. Therefore $SYNCH - SUBSET(k)$ with arbitrary greedy $\mathscr{O}_{word}$ and $\mathscr{O}_{set}$ oracles returns the word $a(b^{\ell+k-1}a)^q b^{\ell+r}a$ of length $1+(\ell+1)(q+1)+(k-1)q+r = (\ell + 1)(q + 2) - 1 = (\ell + 1)(\left\lceil \frac{\ell-1}{k-1} \right\rceil + 1) - 1 \geq \frac{\ell^2-1}{k-1} + \ell \geq \frac{(\ell+(1/2))^2}{k-1}$. The

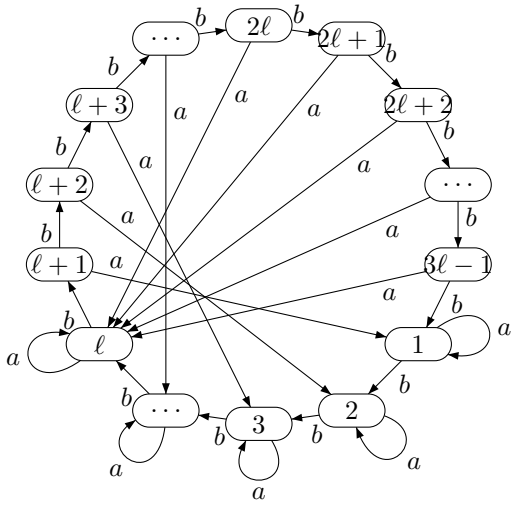

Figure 2: Automaton $\mathscr{B}_\ell$

number of states in $\mathscr{B}_\ell$ is equal to $n = 3\ell - 1$. Thus, approximation ratio is at least

$$\frac{(\ell + (1/2))^2}{k-1} \cdot \frac{1}{2(\ell + (1/2))} = \frac{\ell + (1/2)}{2(k-1)} > \frac{n}{6(k-1)}.$$

□

In the theorem 1 the oracle $\mathscr{O}_{set}$ is greedy. In this regard, we state the following open problem: what is the approximation ratio of $SYNCH - SUBSET(k)$ with an arbitrary greedy oracle $\mathscr{O}_{word}$ and an arbitrary oracle $\mathscr{O}_{set}$? Essentially, we are interested whether it is $O(n)$ for a polynomial-time computable $\mathscr{O}_{set}$. There is a slight evidence for a positive answer in the experimental study of $SYNCH - SUBSET(2)$ performed by E. Tipikin in his master thesis on the series from [6]. He noticed that $SYNCH - SUBSET(2)$ with a greedy oracle $\mathscr{O}_{word}$ and a particular oracle $\mathscr{O}_{set}$ performs extremely well on this series. Although, from algorithmical point of view we are mostly interested in polynomial-time computable oracles $\mathscr{O}_{set}$ deep understanding of the general case may reveal additional properties of synchronizing automata.

References

[1] D.S. Ananichev, M.V. Volkov, and V.V. Gusev. Primitive digraphs with large ex- ponents and slowly synchronizing automata. *Journal of Mathematical Sciences*, **192**(3) (2013),263–278.

[2] Mikhail V. Berlinkov. On two algorithmic problems about synchronizing automata. *CoRR*, abs/1312.2226, 2013.

[3] David Eppstein. Reset sequences for monotonic automata. *SIAM J. Computing*, **19**(3) (June 1990), 500–510.

[4] Michael Gerbush and Brent Heeringa. Approximating minimum reset sequences. In Michael Domaratzki and Kai Salomaa, editors, *Implementation and Application of Automata*, volume 6482 of *Lecture Notes in Computer Science*, pages 154–162. Springer Berlin Heidelberg, 2011.

[5] Jrg Olschewski and Michael Ummels. The complexity of finding reset words in finite automata. In Petr Hlinn and Antonn Kuera, editors, *Mathematical Foundations of Computer Science 2010*, volume 6281 of *Lecture Notes in Computer Science*, pages 568–579. Springer Berlin Heidelberg, 2010.

[6] Mikhail V. Volkov. Synchronizing automata and the Černý conjecture. In Carlos Martn-Vide, Friedrich Otto, and Henning Fernau, editors, *Language and Automata Theory and Applications*, volume 5196 of *Lecture Notes in Computer Science*, pages 11–27. Springer Berlin Heidelberg, 2008.

Limit Theorems for the Number of Multiple Edges in the Configuration Graph*

Irina Cheplyukova

Institute of Applied Mathematical Research,
Karelian Research Centre RAS, Petrozavodsk, Russia

Many real complex systems can be represented and studied in terms of random graphs (see e.g. [2, 1, 3]). The present work focuses on a random graph constructed by the configuration models with the degree of vertices distributed identically and independently. We consider two special cases of such graphs consisting of $N+1$ vertices with numbers $0, 1, \dots, N$. One of them is the power–law random graph with the degree of vertices $1, \dots, N$ distributed according to the law

$$\mathbf{P}\{\xi > k\} = k^{-\tau}, k = 1, 2, \dots, \qquad \tau \in (1, 2).$$

In the second model the degrees of vertices have a binomial distribution with parameters (N, p). The vertex 0 has the degree 0 if the sum of degrees of other vertices is even, otherwise the degree is 1. Those models admit multiple edges and loops. We study the number of multiple edges with given vertex degrees.

Choose two vertices of the random graph, for example, vertices with numbers j and $j-1$. Let k_j and k_{j-1} be the degrees of these vertices. Assume that $k_j \le k_{j-1}$. Let λ be random variable equal to the number of edges joining vertices with numbers j and $j-1$. Denote

$$p(m|k_j, k_{j-1}) = \mathbf{P}\{\lambda = m|\xi_j = k_j, \xi_{j-1} = k_{j-1}\}, m = 0, 1, 2, \dots$$

*The work is supported by the Russian Foundation for Basic Research, grant 13-01-00009.

We obtain the limit theorems for $p(m|k_j, k_{j-1})$. As an example, we get the following results. Theorem 1 is valid for the random graph where vertex degrees have the binomial distribution.

Theorem 1. *Let $N \to \infty$ such that $k_j, k_{j-1} = o(N^a), a < 1/2$. Then for $m = 0, 1, 2, \ldots$*

$$p(m|k_j, k_{j-1}) = \frac{k_j! k_{j-1}! (1 + o(1))}{m!(k_j - m)!(k_{j-1} - m)!(\lambda N)^m} \exp\left\{-\frac{k_j^2}{\lambda N} - \frac{k_j k_{j-1}}{\lambda N}\right\}.$$

For the power–law random graph we have Theorem 2.

Theorem 2. *Let $N, k_j, k_{j-1} \to \infty$ such that*

$$\frac{k_j^2}{\zeta(\tau)N} = o(1), \frac{k_j k_{j-1}}{\zeta(\tau)N} = O(1), k_{j-1} = O(N^{2/3}).$$

Then

$$p(m|k_j, k_{j-1}) = \frac{1 + o(1)}{m!} \left(\frac{k_j k_{j-1}}{\zeta(\tau)N}\right)^m \exp\left\{-\frac{k_j k_{j-1}}{\zeta(\tau)N}\right\},$$

where $\zeta(\tau)$ is the Rimann's zeta-function.

References

[1] W. Aiello, F. Chung, L. Lu, A random graph model for power–law graphs, *Experiment Math.* **10** 1 (2001), 53-66.

[2] H. Esker, R. Hofstad, G. Hooghiemstra, D. Znamenski, Distances in random graphs with infinite mean degrees, *Extremes* **8** (2006), 111-114.

[3] C. Faloutsos, P. Faloutsos, M. Faloutsos On power–law relationships of the Internet topology, *Computer Communications Rev.* **29** (1999), 251-262.

About the Linear Complexity of Sequences Over the Finite Field of Four Elements

Vladimir Edemskiy

Novgorod State University, Veliky Novgorod, Russia

The linear complexity and the autocorrelation are important parameters of pseudo-random sequences significant for practical applications. New balanced quaternary sequences with optimal autocorrelation values ware constructed in [1, 2, 3] using the interval structure and the inverse Gray map. We investigate the linear complexity and the minimal polynomial of above mentioned sequences over the finite field of four elements.

Let $\mathbb{F}_4 = \{0, 1, \mu, \mu + 1\}$ be a finite field of four elements. The Gray mapping $\phi : \mathbb{Z}_4 \to \mathbb{Z}_2 \times \mathbb{Z}_2$ is defined as

$$\phi(0) = (0, 0),\ \phi(1) = (0, 1),\ \phi(2) = (1, 1),\ \phi(3) = (1, 0).$$

If we view $\mathbb{F}_4$ as a vector space over $\mathbb{F}_2$ with basis $\mu,\ 1$, then we can define sequence $v, v_i \in \mathbb{F}_4$ by the inverse Gray map derived from two binary sequences c, d as:

$$v_i = \begin{cases} 0, & \text{if } c_i = d_i = 0, \\ 1, & \text{if } c_i = 0,\ d_i = 1, \\ \mu + 1, & \text{if } c_i = d_i = 1, \\ \mu, & \text{if } c_i = 1,\ d_i = 0. \end{cases} \tag{1}$$

Let a, b be Legendre sequences, Hall sequences or twin-prime sequences. We investigate the linear complexity of three patterns sequences:

1) $c = I(a, L^{1/2}a)$ and $d = I(b, L^{1/2}b + 1)$, where I and L denote the interleaved operator and the left cyclic shift operator respectively [3].

2) [2]

$$c_i = \begin{cases} a_i, & \text{if } i \equiv 0(\text{mod}2), \\ a_i, & \text{if } i \equiv 1(\text{mod}2). \end{cases} \quad d_i = \begin{cases} b_i, & \text{if } i \equiv 0(\text{mod}2), \\ b_i + 1, & \text{if } i \equiv 1(\text{mod}2). \end{cases}$$

3) c, d are obtained from 2 various Legendre sequences [1]

We derive parameters of sequences with optimal autocorrelation values and high linear complexities over the finite field of four elements.

References

[1] Kim Y-S., Jang J-W., Kim S-H., No J-S.: New Quaternary Sequences with Ideal Autocorrelation Constructed from Legendre Sequences.*IEICE Trans. Fund. Electron.* **E96-A** (9) (2013), 1872-1882.

[2] Lim T., No J-S., Chung H.: New Construction of Quaternary Sequences with Good Correlation Using Binary Sequences with Good Correlation. *IEICE Trans. Fundamentals.* **E94-A**(8) (2011), 1701-1705.

[3] Tang X.H., Ding C.: New classes of balanced quaternary and almost balanced binary sequences with optimal autocorrelation value. *IEEE Trans. Inf. Theory.* **56** (2010), 6398-6405.

Dynamic Programming Usage in Steiner Problem in Graphs

Joseph V. Romanovsky

Saint-Petersburg State University, Saint-Petersburg, Russia

Dmitry Eibozhenko

Software developer, Yandex, Moscow, Russia

The object of our research is a classical problem in graph theory — the Steiner problem. The Steiner problem is widely used in practice, including projecting computational circuits and microprocessors, telecommunication, gas and oil industries.

The statement of the *Steiner problem in graph* is the following: There is weighted graph (M, N) and some subset of vertices T called *terminal set*. Find a minimal subgraph that connects all vertices of the terminal set.

We study *the problem in directed graphs* where the required graph must contain paths from the given root b to a set of terminals T. Also we study *the metric problem* where the initial graph is placed on plane and the edge lengths correspond to "distance" between vertices. As is known, the Steiner problem in all these statements is NP-hard [1].

Algorithm for finding exact solution for Steiner problem in directed graphs that uses dynamic programming and Bellman equation is known [2]. The solution of the posed problem should be found in the context of the solutions of the set of Steiner problems: the problem is posed for any vertex of graph considered as a root and any subset of initial terminal set is considered as a new terminal set.

This method is hardly applicable in practice, because it has exponential complexity and allows to solve only problems with the set of dozens

terminal vertices, while there is often a need to solve problems with thousands and even millions terminals. At the same time, it can be considered as a base to construct some approximative and heuristic algorithms.

In the course of the research we have worked out a set of heuristic algorithms based on a dynamic programming method. The talk includes some of them:

- k-cluster algorithm, based on dissection of the initial graph into not more than k disjoint subgraphs, on every of which a new problem, induced by the source problem, is posed [3]. Solution of the original problem is the composition of solutions of all the mentioned problems. Solutions provided by this algorithm differs from exact solution on experimental data (set of 400 problems where $|M| \in [80, 640]$, $|N| \in [120, 204480]$, $|T| \in [6, 160]$ from *SteinLib* problem library [6]) for not more than 5% on average.

- algorithm S^* for Steiner problem on metric graphs. Its idea implies reducing amount of the considered subsets of terminal set based on the information about relative positions of vertices [5]. Several different ways of reducing are presented. Experiments were conducted on set of 150 generated problems where $|M| \in [160, 640]$, $|N| \in [2371, 204156]$, $|T| \in [20, 120]$. Its efficiency was compared to efficiency of well-known approximation algorithms, with better results on average.

References

[1] Garey M. R., Johnson D. S. Computers and Intractability: A Guide to the Theory of NP-Completeness, Macmillan Higher Education, 1979.

[2] Romanovsky J. V. Discrete analysis. Saint-Petersburg: BHV-Petersburg, 2003 (in Russian).

[3] Eibozhenko D. A. k-Clustering approximation for directed Steiner problem, *Vestnik SPbSU, series 10 — Applied Mathematics.* Is. 2, (2011), 29-39 (in Russian).

[4] Romanovsky J. V., Eibozhenko D. A. Modifications of dynamic programming method for Steiner problem in directed graphs. *The Computer Tools in Education Journal.* Is. 5, (2010), 22-28 (in Russian).

[5] Eibozhenko D. A. Approximate algorithm S^* for Steiner problem in metric graphs. *Vestnik SPbSU, series 10 — Applied Mathematics.* Is. 3, (2012), 23-32 (in Russian).

[6] SteinLib Testdata Library, 2009. [Electronic source]. URL: http://steinlib.zib.de/steinlib.php

Clustering Words and Interval Exchanges

Sébastien Ferenczi

Institut de Mathématiques de Marseille, Marseille, France

Luca Q. Zamboni

Institut Camille Jordan, Université Claude Bernard Lyon 1
Villeurbanne Cedex, France
Department of Mathematics and Turku Centre for Computer Science,
University of Turku, Turku, Finland

In 1994 Michael Burrows and David Wheeler [1] introduced a transformation on words which proved very powerful in data compression. The aim of the present note is to characterize those words which cluster under the Burrows-Wheeler transform, that is to say which are transformed into such expressions as $4^a3^b2^c1^d$ or $2^a5^b3^c1^d4^e$. Clustering words on a binary alphabet have already been extensively studied (see, for instance, [8, 11]) and identified as particular factors of the Sturmian words. Some generalizations and partial characterizations to r letters appear in Restivo and Rosone [13], but it had not yet been observed that clustering words are intrinsically related to a dynamical object called *interval exchange transformations*.

1. Definitions

Let $A = \{a_1 < a_2 < \cdots < a_r\}$ be an ordered alphabet and $w = w_1 \cdots w_n$ a *primitive* word on the alphabet A, i.e., w is not a power of another word. For simplification we suppose that *each letter of A occurs in w*.

The *Parikh vector* of w is the integer vector $(n_1, \ldots, n_k)$ where n_i is the number of occurrences of a_i in w. The *(cyclic) conjugates* of w are the words $w_i \cdots w_n w_1 \cdots w_{i-1}$, $1 \leq i \leq n$. As w is primitive, w has precisely n-cyclic conjugates. Let $w_{i,1} \cdots w_{i,n}$ denote the i-th conjugate of w where the n-conjugates of w are ordered by ascending lexicographical order. Then the *Burrows-Wheeler transform* of w, denoted by $B(w)$, is the word $w_{1,n} w_{2,n} \cdots w_{n,n}$. In other words, $B(w)$ is obtained from w by first ordering its cyclic conjugates in ascending order in a rectangular array, and then reading off the last column. For instance $B(2314132) = 4332211$. We say w is π-*clustering* if $B(w) = a_{\pi 1}^{n_{\pi 1}} \cdots a_{\pi r}^{n_{\pi r}}$, where $\pi \neq Id$ is a permutation on $\{1, \ldots, r\}$. We say w is *perfectly* clustering if it is π-clustering for $\pi i = r + 1 - i$, $1 \leq i \leq r$. For instance 2314132 is perfectly clustering. Restivo and Rosone [13] showed that if w perfectly clusters, then w is strongly (or circularly) rich, i.e., w^2 has $|w^2| + 1$ distinct palindromic factors. But this condition is not a characterization of perfectly clustering words (see Example 6.4 in Restivo and Rosone[13]).

Definition 1 A (continuous) *r-interval exchange transformation* T with probability vector
$(\alpha_1, \alpha_2, \ldots, \alpha_r)$, and permutation π is defined on the interval $[0, 1[$, partitioned into r intervals

$$\Delta_i = \left[\sum_{j<i} \alpha_j, \sum_{j \leq i} \alpha_j \right[,$$

by

$$Tx = x + \tau_i \quad \text{when} \quad x \in \Delta_i,$$

where $\tau_i = \sum_{\pi^{-1}(j) < \pi^{-1}(i)} \alpha_j - \sum_{j<i} \alpha_j$.

Intuitively this means that the intervals Δ_i are re-ordered by T following the permutation π. Note that our use of the word "continuous" does not imply that T is a continuous map on $[0, 1[$ (though it can be modified to be made so); it is there to emphasize the difference with its discrete analogue.

Definition 2 A *discrete r-interval exchange transformation* T with length vector $(n_1, n_2, \ldots, n_r)$, and permutation π is defined on a set of $n_1 + \cdots + n_r$

points $x_1, \ldots, x_{n_1+\cdots+n_r}$ partitioned into r intervals

$$\Delta_i = \{x_k, \sum_{j<i} n_j < k \leq \sum_{j\leq i} n_j\}$$

by

$$Tx_k = x_{k+s_i} \quad \text{when} \quad x_k \in \Delta_i,$$

where $s_i = \sum_{\pi^{-1}(j)<\pi^{-1}(i)} n_j - \sum_{j<i} n_j$.

We recall the following notions, defined for any transformation T on a set X equipped with a partition Δ_i, $1 \leq i \leq r$.

Definition 3 The *trajectory* of a point x under T is the infinite sequence $(x_n)_{n\in\mathbb{N}}$ defined by $x_n = i$ if $T^n x$ belongs to Δ_i, $1 \leq i \leq r$. The mapping T is *minimal* if whenever E is a nonempty closed subset of X and $T^{-1}E = E$, then $E = X$.

2. Main result

Theorem 1 *Let $w = w_1 \cdots w_n$ be a primitive word on $A = \{1, \ldots, r\}$, such that every letter of A occurs in w. The following are equivalent:*

1. *w is π-clustering,*
2. *ww occurs in a trajectory of a minimal discrete r-interval exchange transformation with permutation π,*
3. *ww occurs in a trajectory of a discrete r-interval exchange transformation with permutation π,*
4. *ww occurs in a trajectory of a continuous r-interval exchange transformation with permutation π.*

Some of the hypotheses of Theorem 1 may be weakened.

Alphabet. $\{1, \ldots, r\}$ can be replaced by any ordered set $A = \{a_1 < a_2 < \cdots < a_r\}$ by using a letter-to-letter morphism. Thus for a given word w, we can restrict the alphabet to the letters occurring in w.

Primitivity. The Burrows-Wheeler transformation can be extended to a non-primitive word $w_1 \cdots w_n$, by ordering its n (non necessarily different) conjugates $w_i \cdots w_n w_1 \cdots w_{i-1}$ by non-strictly increasing lexicographical order and taking the word made by their last letters. Our Theorem 1 is still valid for non-primitive words: the proof in the first direction does not use the primitivity, while in the reverse direction we write $w = u^k$, apply our proof to the primitive u, and check that u^{2k} occurs also in a trajectory.

Two permutations. An extension of Theorem 1 which fails is to consider, as dynamicists do, interval exchange transformations defined by permutations π and π'; this amounts to coding the interval Δ_i by $\pi' i$ instead of i. A simple counter-example will be clearer than a long definition: take points $x_1, \ldots, x_9$ labelled 223331111 and send them to 111133322 by a (minimal) discrete 3-interval exchange transformation, but where the points are not labelled as in Definition 3 (namely $Tx_1 = x_8$, $Tx_3 = x_5$ etc...). Then $w = 123131312$ is such that ww occurs in trajectories of T but $B(w) = 323311112$.

3. Building clustering words

Theorem 1 provides two different ways to build clustering words, from infinite trajectories either of discrete (or rational) interval exchange transformations or of continuous aperiodic interval exchange transformations. For $r = 2$ and the permutation $\pi 1 = 2, \pi 2 = 1$, the first way gives all the periodic balanced words, and the second way gives all infinite Sturmian words: both ways of building clustering words on two letters are used, explicitly or implicitly, in Jenkinson and Zamboni [8].

The use of discrete interval exchange transformations leads naturally to the question of characterizing all minimal discrete r-interval exchange transformations through their length vector; this has been solved by Pak and Redlich [12] for $n = 3$ and $\pi 1 = 3, \pi 2 = 2, \pi 3 = 1$: if the length vector is (n_1, n_2, n_3), minimality is equivalent to $(n_1 + n_2)$ and $(n_2 + n_3)$ being coprime. Thus

Example 1 *With the discrete interval exchange* $11223333 \to 333322111$, *we get the perfectly clustering word* 313131223.

The same reasoning can be extended to other permutations: for $\pi 1 = 2, \pi 2 = 3, \pi 3 = 1$, minimality is equivalent to n_1 and $(n_2 + n_3)$ being coprime; for $\pi 1 = 3, \pi 2 = 1, \pi 3 = 2$, minimality is equivalent to n_3 and $(n_2 + n_1)$ being coprime; for other permutation on these three letters, T is never minimal.

For $r \geq 4$ intervals, the question is still open. An immediate equivalent condition for non-minimality is $\sum_{i=1}^{m} s_{w_i} = 0$ for $m < n_1 + \cdots + n_r$ and $w_1 \cdots w_m$ a word occurring in a trajectory. It is easy to build non-minimal examples satisfying such an equality for simple words w, for example for $r = 4$ and $\pi 1 = 4, \pi 2 = 3, \pi 3 = 2, \pi 4 = 1$, $n_1 = n_2 = n_3 = 1$ gives non-minimal examples for any value of n_4, the equality being satisfied for $w = 24^q$ if $n_4 = 3q$, $w = 14^{q+1}$ if $n_4 = 3q+1$, $w = 34^q$ if $n_4 = 3q+2$. Similarly, the following example shows how we still do get clustering words, but they may be somewhat trivial.

Example 2 *The discrete interval exchange* $111233444 \to 444332111$ *satisfies the above equality for* $w = 14$*; it is non-minimal and gives two perfectly clustering words on smaller alphabets,* 41 *and* 323.

Trajectories of interval exchange transformations may be explicitly constructed via the *self-dual induction* algorithms of [5] for $r = 3$ and $\pi 1 = 3, \pi 2 = 2, \pi 3 = 1$, [6] for all r and $\pi i = r+1-i$, and the forthcoming [4] in the most general case. Many explicit examples of ww have been built in this way.

- For $r = 3$, $w = A_k$, $w = B_k$ (see Proposition 2.10 in Ferenczi, Holton and Zamboni [5]),

  Example 3 13131312222 *and* 131312221312213122 *are perfectly clustering.*

- For $r = 4$, $w = M_2(k)$, $w = P_3(k)M_1(k)$ (see Lemma 4.1 and Lemma 5.1 in Ferenczi and Zamboni [7]),

  Example 4 $2^m(3141)^n32$ *are perfectly clustering for any* m *and* n.

- For all $r = n$, $w = P_{k,1,1}$, $w = P_{k,n-i,i+1}P_{k,i+1,n-i}$, $w = M_{k,n+1-i,i-1}M_{k,i-1,n+1-i}$ (see Theorem 12 in Ferenczi [3]);

Example 5 525243425251615251616152516 1 *is perfectly clustering.*

For other permutations, we describe in Ferenczi [4] an algorithm generalizing the one in Ferenczi and Zamboni [6], but we do not know if every interval exchange transformation produces infinitely many ww. For the permutation $\pi 1 = 4, \pi 2 = 3, \pi 3 = 1, \pi 4 = 2$, examples can be found in Theorem 5.2 of [6], with $w = P_{1,q_n} M_{2,q_n}$, $w = P_{2,q_n} M_{3,q_n}$, $w = P_{3,q_n} M_{1,q_n}$,

Example 6 4123231312412 *is π-clustering.*

We remark that our self-dual induction algorithms for aperiodic interval exchange transformations generate families of nested clustering words with increasing length, and thus may be more efficient in producing very long clustering words than the more immediate algorithm using discrete interval exchange transformations.

References

[1] M. Burrows and D.J. Wheeler, A block sorting data compression algorithm, *Tech. report, Digital System Research Center* (1994).

[2] M. Crochemore, J. Désarménien, and D. Perrin, A note on the Burrows-Wheeler transformation, *Theoret. Comput. Sci.* **332** (2005), no. 1-3, 567–572.

[3] S. Ferenczi, Billiards in regular $2n$-gons and the self-dual induction,*J. Lond. Math. Soc.* (2) **87** (2013), p. 766–784.

[4] S. Ferenczi, A generalization of the self-dual induction to every interval exchange transformation, , in course of acceptation by *Ann. Inst. Fourier (Grenoble)*, http://iml.univ-mrs.fr/˜ferenczi/fie.pdf.

[5] S. Ferenczi, C. Holton and L.Q. Zamboni, Structure of three-interval exchange transformations. II. A combinatorial description of the trajectories, *J. Anal. Math.* **89** (2003), 239–276.

[6] S. Ferenczi and L.Q. Zamboni, Structure of $k-$interval exchange transformations: induction, trajectories, and distance theorems, *J. Anal. Math.* **112** (2010), 289–328.

[7] S. Ferenczi and L.Q. Zamboni, Eigenvalues and simplicity of interval exchange transformations, *Ann. Sci. Éc. Norm. Supér. (4)* **44** (2011), 361–392.

[8] O. Jenkinson and L.Q. Zamboni, Characterisations of balanced words via orderings, *Theoret. Comput. Sci.* **310** (2004), 247–271.

[9] M. Keane, Interval exchange transformations, *Math. Z.* **141** (1975), 25–31.

[10] S. Mantaci, A. Restivo, G. Rosone and M. Sciortino, An extension of the Burrows-Wheeler transform, *Theoret. Comput. Sci.* **387** (2007), 298–312.

[11] S. Mantaci, A. Restivo and M. Sciortino, Burrows-Wheeler transform and Sturmian words. *Inform. Process. Lett.* **86** (2003), 241–246.

[12] I. Pak and A. Redlich, Long cycles in abc-permutations, *Funct. Anal. Other Math.* **2** (2008), 87–92.

[13] A. Restivo and G. Rosone, Burrows-Wheeler transform and palindromic richness. *Theoret. Comput. Sci.* **410** (2009), 3018–3026.

[14] A. Restivo and G. Rosone, Balancing and clustering of words in the Burrows-Wheeler transform, *Theoret. Comput. Sci.* **412** (2011), 3019–3032.

Resolution complexity of Perfect Mathcing principles for Sparse Graphs

Dmitry Itsykson

Steklov Institute of Mathematics at St.Petersburg the Russian Academy of Sciences, St.Petersburg, Russia

Mikhail Slabodkin

St. Petersburg Academic University, St.Petersburg, Russia

Dmitry Sokolov

Steklov Institute of Mathematics at St.Petersburg the Russian Academy of Sciences, St.Petersburg, Russia

Abstract

The resolution complexity of the perfect matching principle was studied by Razborov [11], who developed a technique for proving its lower bounds for dense graphs. We construct a a constant degree bipartite graph G_n such that the resolution complexity of the perfect matching principle for G_n is $2^{\Omega(n)}$, where n is the number of vertices in G_n. This lower bound matches with the upper bound $2^{O(n)}$ up to an application of a polynomial. Our result implies the $2^{\Omega(n)}$ lower bounds for the complete graph K_n and the complete bipartite graph $K_{n,O(n)}$ that improve the lower bounds followed from [11]. Our results also implies the well-known exponential lower bounds on the resolution complexity of the pigeonhole principle, the

functional pigeonhole principle and the pigeonhole principle over a graph.

We also prove the following corollary. For every natural number d, for every n large enough, for every function $h : \{1, 2, \ldots, n\} \to \{1, 2, \ldots, d\}$, we construct a graph with n vertices that has the following properties. There exists a constant D such that the degree of the i-th vertex is at least $h(i)$ and at most D, and it is impossible to make all degrees equal to $h(i)$ by removing the graph's edges. Moreover, any proof of this statement in the resolution proof system has size $2^{\Omega(n)}$. This result implies well-known exponential lower bounds on the Tseitin formulas as well as new results: for example, the same property of a complete graph.

1. Introduction

The resolution proof system is one of the simplest and well-studied proof systems. There are well known methods of proving lower and upper bounds on the complexity of several types of formulas. However, there are no known universal methods to determine an asymptotic resolution complexity of a given family of formulas. We say that a family of unsatisfiable CNF formulas F_n is weaker than a family of unsatisfiable formulas H_n if every clause of H_n is an implication of a constant number of clauses of F_n. Since the resolution proof system is implication complete, the size of any resolution proof of H_n is at least the size of the minimal resolution proof of F_n. Thus it is interesting to prove lower bounds for for fomulas as weak as possible.

CNF formulas PHP_n^m encode the pigeonhole principle; PHP_n^m states that it is possible to put m pigeons into n holes in such a way that every pigeon is contained in at least one hole and every hole contains at most one pigeon. PHP_n^m depends on variables $p_{i,j}$ for $i \in [m]$ and $j \in [n]$ and $p_{i,j} = 1$ iff the i-th pigeon is in the j-th hole. PHP_n^m is unsatisfiable iff $m > n$. Haken [4] proved the lower bound $2^{\Omega(n)}$ on the resolution complexity of PHP_n^{n+1}. Raz [8] proved the lower bound 2^{n^ϵ} on the resolution complexity of PHP_n^m for some positive constant ϵ and arbitrary $m > n$. This lower bound was simplified and improved to $2^{\Omega(n^{1/3})}$ by Razborov [9].

Urqhart [14] and Ben-Sasson, and Wigderson [2] consider formulas $G-\mathrm{PHP}_m^n$ that are defined by a bipartite graph G; the first part of G corresponds to pigeons and consists of m vertices, and the second part corresponds to holes and consists of n vertices. Every pigeon must be

contained in one of adjacent holes. Formulas $G-\mathrm{PHP}_n^m$ may be obtained from PHP_n^m by substituting variables which do not have corresponding edges in G with zeroes. The paper [2] presents the lower bound $2^{\Omega(n)}$ for formulas $G-\mathrm{PHP}_n^m$ where $m = O(n)$ and G is a bipartite constant degree expander.

Razborov [10] considers a so called functional pigeonhole principle FPHP_n^m that is a weakening of PHP_n^m; the formula FPHP_n^m is the conjunction of PHP_n^m and additional conditions stating that every pigeon is contained in at most one hole. Razborov proved a lower bound $2^{\Omega\left(\frac{n}{(\log m)^2}\right)}$ for FPHP_n^m that implies a lower bound $2^{\Omega\left(n^{1/3}\right)}$ depending only on n.

Let for every graph G a formula PMP_G (from the Perfect Matching Principle) encode that G has a perfect matching. Variables of PMP_G correspond to edges, and for every vertex of G exactly one incident edge has value 1. Razborov [11] proved that if G has no perfect matchings, then the resolution complexity of PMP_G is at least $2^{\frac{\delta(G)}{\log^2 n}}$, where $\delta(G)$ is the minimal degree of the graph and n is the number of vertices.

Alekhnovich [1] and Dantchev and Riis [3] consider the graphs of the chessboard $2n \times 2n$ with two opposite corners removed. The perfect matching principle for such graphs is equivalent to the possibility to tile such chessboards with domino. The strongest lower bound $2^{\Omega(n)}$ was proved in [3] and this lower bound is polynomially connected with the upper bound $2^{O(n)}$. We note that the number of variables is $\Theta(n^2)$.

Our results For all n and all $m \in [n+1, O(n)]$ we give an example of a bipartite graph $G_{m,n}$ with m and n vertices in its parts such that all degrees are bounded by a constant and the resolution complexity of $\mathrm{PMP}_{G_{m,n}}$ is $2^{\Omega(n)}$. The number of variables in such formulas is $O(n)$, therefore the lower bound matches (up to an application of a polynomial) the trivial upper bound $2^{O(n)}$ that holds for every formula with $O(n)$ variables. This is the first lower bound for perfect matching principle that is exponential in the number of variables. In particular, our results imply that the resolution complexity of $\mathrm{PMP}_{K_{m,n}}$ is $2^{\Omega(n)}$, where $K_{m,n}$ is the complete bipartite graph and $m = O(n)$. And this lower bound improves the lower bound $2^{\Omega(n/\log^2 n)}$ that follows from [11] and matches (up to a polynomial application) the upper bound $n2^n$ that follows from the upper bound for PHP_n^{n+1} [12]. Our result implies the lower bound $2^{\Omega(n)}$

on the resolution complexity of PMP_{K_n}, where K_n is a complete graph on n vertices, and it is also better than the lower bound $2^{\Omega(n/\log^2 n)}$ that follows from [11]. We note that $PMP_{G_{m,n}}$ is weaker than $G_{m,n} - \mathrm{PHP}_n^m$, PHP_n^m and FPHP_n^m, therefore our lower bound implies the same lower bound for $G_{m,n} - \mathrm{PHP}_n^m$, PHP_n^m and FPHP_n^m. To put it more precisely, we we prove the following theorem:

Theorem 2 *Let G be a bipartite graph with parts X and Y such that the folowing holds:*

1. *G is a (r, c)-boundary expander; i.e. for all $A \subseteq X$, if $|A| \leq r$ then $|\delta(A)| \geq c|A|$, where $\delta(A)$ is the set of all vertices in Y that are connected with exactly one vertex in A;*

2. *There is a matching in G that covers all vertices from Y.*

Then the width of all resolution proofs of PMP_G is at least $cr/2$. If additionally degrees of all vertices are at most D, then (using [2] we get that) the size of any resolution proof of PHP_G is at least $2^{\Omega\left(\frac{(cr/2-D)^2}{n}\right)}$, where n is the number of edges in G.

The condition that G has a matching covering all vertices from Y can not be removed for free since for every (r, c)-boundary expander it is possible to add one vertex to X and $\lceil c \rceil$ vertices to Y such that the new vertex in X is connected with all new vertices in Y. The resulting graph is also (r, c)-boundary expander but the resulting formula will contain unsatisfiable subformula that depends on $\lceil c \rceil$ variables, hence it can be refuted with width $\lceil c \rceil$. We do not know whether it is possible to replace the second condition in the theorem by a weaker condition.

To estimate the width we use the method introduced by Ben-Sasson and Wigderson in [2]. However, we use a non-standard notion of a semantic implication and a non-standard measure on the set of clauses.

An example of a graph that suits the conditions of Theorem 2 can be constructed from every lossless expander by removing vertices of high degrees as it was shown in [6], and by adding a matching that covers all vertices from Y. For example, we can use the explicit construction of lossless expanders from [7] (or the randomized construction [5]).

Theorem 2 implies a more general theorem:

Theorem 3 *For graph $G(V,E)$ and function $h : V \to \{1,2,\ldots,d\}$ we define a formula $\Psi_G^{(h)}$, that code that G has a subgraph H such that for all v in H the degree of v equals $h(v)$. For any $d \in \mathbb{N}$, there exists $D \in \mathbb{N}$ that for all n large enough and every function $h : V \to \{1,2,\ldots,d\}$, where $|V| = n$, there exists graph $G(V,E)$ with degrees of vertices at most D such that the formula $\Psi_G^{(h)}$ is unsatisfiable and the size of any resolution proof of $\Psi_G^{(h)}$ is at least $2^{\Omega(n)}$.*

If h maps V to $\{1,2\}$, then $\Psi_G^{(h)}$ is weaker than Tseitin formulas based on graph G. Thus our result implies the lower bound $2^{\Omega(n)}$ on the resolution complexity of Tseitin formulas that was proved in [13].

2. Preliminaries

We consider simple graphs without loops and multiple edges. The graph G is called bipartite if its vertices can be divided into two disjoint parts X and Y in such a way that any edge is incident to one vertex from X and one vertex from Y. We denote $G(X,Y,E)$ a bipartite graph with parts X and Y and set of edges E. A matching in a graph $G(V,E)$ is such a set of edges $E' \subseteq E$ that any vertex $v \in V$ has at most one incident edge from E'. A matching E' covers a vertex v if there exists $e \in E'$ that is incident to v. A perfect matching is a matching that covers all vertices of G. For a bipartite graph $G(X,Y,E)$ and a set $A \subseteq X$ we denote $\Gamma(A)$ a set of all neighbors of vertices from A.

Lemma 1 (Hall) *Consider such a bipartite graph $G(X,Y,E)$ that for some $A \subseteq X$ for all $B \subseteq A$ the following inequality holds: $\Gamma(B) \geq |B|$. Then there is a matching that covers all vertices from A.*

For a CNF formula φ a proof of its unsatisfiability in the resolution proof system is a sequence of clauses with the following properties: the last clause is an empty clause (we denote it by $\Box$); any other clause is either a clause of initial formula φ or can be obtained from previous ones by the resolution rule. The resolution rule admits to infer a clause $(B \vee C)$ from clauses $(x \vee B)$ and $\neg x \vee C$. The size of a resolutional proof is the number of clauses in it.

In [2] E. Ben-Sasson and A. Wigderson introduced a notion of formula width. A width of a clause is a number of literals contained it it. For a

k-CNF formula φ a width of φ is a maximum width of clauses of φ. A width of a resolution proof is a width of the largest clause used in it.

Theorem 4 ([2]) *For any k-CNF unsatisfiable formula φ the size of resolution proof is at least $2^{\Omega\left(\frac{(w-k)^2}{n}\right)}$, where w is a minimal width of a resolutional proof and n is a number of variables used in φ.*

Lemma 2 *Let ϕ be a formula that is obtained from unsatisfiable formula ψ by a substitution of several variables. Then ϕ is unsatisfiable and the size of the minimal resolution proof of ψ is at least the size of the minimal resolution proof of ϕ.*

3. Subgraph extraction

3.1. Existence of a perfect matching

For an undirected graph $G(V, E)$ we construct a formula PMP_G that encodes that G has a perfect matching. We assign a binary variable x_e for all $e \in E$. PMP_G is the conjunction of the following conditions: for all $v \in V$ exactly one edge that incident to v has value 1. Such conditions can be written as the conjunction of the statement that at least one edge takes value 1: $\bigvee_{(v,u)\in E} x_{(v,u)}$ and the statement that for any pair of edges e_1, e_2 incident to v at most one of them takes value 1: $\neg x_{e_1} \vee \neg x_{e_2}$.

Note that if degrees of all vertices are at most D, then PMP_G is a D-CNF formula.

In this section we prove the following theorem:

Theorem 5 *There exists a constant D such that for all C that for all n large enough and for all $m \in [n+1, Cn]$ it is possible to construct in polynomial in n time such bipartite graph $G(V, E)$ with m and n vertices in parts that all degrees are at most D and the formula PMP_G is unsatisfiable and the size of any resolution proof of PMP_G is at least $2^{\Omega(n)}$.*

Definition 4 *A bipartite graph $G(X, Y, E)$ is (r, c)-boundary expander if for any set $A \subseteq X$ such that $|A| \leq r$ the following inequality holds $|\delta(A)| \geq c|A|$, where $\delta(A)$ denotes the set of all such vertices in Y that are connected with the set A by the unique edge.*

Lemma 3 *Let bipartite graph $G(X, Y, E)$ have two matchings, the first one covers all vertices from Y and the second covers all vertices from $A \subseteq X$. Then there exists a matching in G that covers A and Y simultaneously.*

Proof. Let L denote the matching that covers all vertices from the set A and let F be a matching that covers all vertices from Y. We prove that if F does not cover all vertices from A, then one may construct a matching F' that covers more vertices of A than F and also covers all vertices from Y. Therefore there is such a matching that covers A and Y.

Consider some vertex $v_1 \in A$ that is not covered by F and such path $v_1, u_1, v_2, u_2, \ldots, u_{k-1}, v_k$ that $(v_i, u_i) \in L$, $(u_i, v_{i+1}) \in F$ and $v_1, v_2, \ldots, v_{k-1} \in A$ and $v_k \notin A$.

For any fixed $v_1 \in A$ such a path can be constructed deterministically: starting at vertex v_1 the edges of the path belong to alternating matchings L and F. For every vertex from X at most one of outgoing edges belongs to L. For every vertex from Y exactly one of outgoing edges belongs to F. The path can't become a cycle because v_1 has no incident edges from F, therefore the constructed path will lead to some vertex $v_k \notin A$.

Let matching F' be constructed from F by removing all edges (v_i, v_{i+1}) and adding edges (u_i, v_i) for $1 \leq i < k$. Now F' covers all Y and covers one additional vertex of A in comparison with F. □

Lemma 4 *Let $G(X, Y, E)$ be a bipartite (r, d, c)-boundary expander with $c > 2$ and $|X| > |Y|$. Let G have a matching that covers all vertices from the part Y. Then the formula PMP_H is unsatisfiable and the width of its resolution refutation is at least $cr/2$.*

Proof. Parts X and Y have different number of vertices, hence there are no perfect matchings in G and PMP_G is unsatisfiable.

We call an assignment to variables of PMP_G proper if for every vertex v at most one edge incident to v has value 1. For some subset $S \subseteq V$ and for a clause C we say that S properly implies C if any proper assignment that satisfies all constraints in vertices from S, also satisfies C. We denote it as $S \vdash C$.

Now we define a measure on clauses from a resolution refutation of PMP_G: $\mu(C) = \min\{|S \cap X| \mid S \vdash C\}$.

The measure μ has the following properties:

1) The measure of any clause from PMP_G equals 0 or 1.

2) Semiadditivity: $\mu(C) \leq \mu(C_1)+\mu(C_2)$, if C is obtained by applying of resolution rule to C_1 and C_2.

Let $S_1 \vdash C_1$, $|S_1 \cap X| = \mu(C_1)$ and $S_2 \vdash C_2$, $|S_2 \cap X| = \mu(C_2)$. Hence $S_1 \cup S_2 \vdash C_1$ and $S_1 \cup S_2 \vdash C_2$, so $S_1 \cup S_2 \vdash C$, therefore $\mu(C) \leq |S_1 \cap X| + |S_2 \cap X| = \mu(C_1) + \mu(C_2)$.

3) The measure of the empty clause $\square$ is more than r.

Let $\mu(\square) \leq r$, then there is such $S \subseteq V$ that $S \vdash \square$ and $|S \cap X| \leq r$. For all $A \subseteq S \cap X$ the following holds $|\Gamma(A)| \geq |\delta(A)| \geq (c-1)|A| \geq |A|$, and Hall's Lemma (Lemma 1) implies that there is a matching in H that covers all $S \cap X$. By construction of H it has a matching that covers all vertices of Y, therefore Lemma 3 implies that there exists a matching that covers $S \cap X$ and Y, hence it covers S. This matching corresponds to an assignment that satisfies all constraints for vertices from S, but it is impossible to satisfy the empty clause and we get a contradiction with the fact that $\mu(\square) \leq r$.

The semiadditivity of the measure implies that any resolution proof of the formula PMP_G contains a clause C with the measure in the interval $\frac{r}{2} \leq \mu(C) \leq r$. Let $S \vdash C$ and $|S \cap X| = \mu(C)$. For the sake of brevity let $A = S \cap X$. Since G is a (r, c)-boundary expander, $\delta(A) \geq c|A|$. Let F denote the set of edges between A and $\delta(A)$. Every vertex from $\delta(A)$ has exactly one incident edge leading to A, therefore $|F| = |\delta(A)|$. Consider one particular edge $f \in F$, let $f = (u, v)$, where $u \in A$. Since $|(S \setminus \{u\}) \cap X| < |S \cap X|$, clause C is not properly implied from the set $S \setminus \{u\}$, i. e. there exists a proper assignment σ that satisfies all restrictions in the vertices $S \setminus \{u\}$, but refutes the clause C. Such assignment σ can not satisfy the constraint in the vertex u, since otherwise σ would satisfy S and therefore satisfy C. Since σ is a proper assignment, σ assigns value 0 to all edges that are incident with u .

We consider two cases: 1) σ refutes a constraint in the vertex v; 2) σ satisfies a constraint in the vertex v.

In the first case we consider another assignment σ' that differs from σ in the value of the edge f. Note that σ' is proper and satisfies all constraints from S, so it satisfies C. Since σ does not satisfy C, the variable f is contained in C.

In the second case σ satisfies v. There is an edge e incident to v such that $\sigma(e) = 1$. The vertex v is a boundary vertex for A, therefore the

other endpoint of e does not belong to A. Consider an assignment σ'' that is obtained from σ by changing the values of f and e, σ'' is proper and it satisfies all constraints from S, and hence it satisfies C. Thus C contains either e or f. Thus for all $v \in \delta(A)$ at least one of the edges incident to v occurs in C. Therefore the size of the clause C is at least $|\delta(A)| \geq c|A| \geq cr/2$.

□

We say that a graph is explicit if it can be constructed in time polynomial in the number of its vertices.

Lemma 5 ([6], lemma 6.2) *For all d large enough and for all m there exists explicit construction of $(r, 0.5d)$-boundary expander $G(X, Y, E)$ with $|X| = |Y| = m$, $r = \Omega(m)$ such that degrees of all vertices from X are at most d and degrees of all vertices from Y are at most d^2.*

Corollary 1 *For all d large enough and for all C and all n and $m \in [n+1, Cn]$ there is an explicit construction of $(r, 0.4d)$-boundary expander $G(X, Y, E)$ with $|X| = m$, $|Y| = n$ and $r = \Omega(n)$ such that degrees of all vertices from X are at most d and degrees of all vertices from Y are at most d^2.*

Proof. The required graph can be obtained from Lemma 5 by deleting several vertices from the part Y. □

Proof. [Proof of Theorem 5] Consider some $d > 5$ that satisfies Corollary 1; consider $(r, 0.4d)$-boundary expander H from the Corollary 1 that has m and n vertices in parts. Let graph G be obtained from H by adding any matching that covers all vertices from the part Y. Graph G is a $(r, c-1)$-boundary expander, since the addition of a matching increases degrees of vertices in X at most by 1 and for every $A \subseteq X$ the size of $\delta(A)$ decreases by at most $|A|$.

Lemma 4 implies that the width of any resolution proof of PMP_G is at least $\Omega(n)$. Theorem 4 implies that the size of any resolution proof of PMP_G is at least $2^{\Omega(n)}$. □

4. Subgraph extraction

Let $G(V, E)$ be an undirected graph and h be a function $V \to \mathbb{N}$ such that for every vertex $v \in V$, $h(v)$ is at most the degree of v. We consider

formula $\Psi_G^{(h)}$; its variables corresponds to edges of G. $\Psi_G^{(h)}$ is a conjunction of the following statements: for every $v \in V$ exactly $h(v)$ edges that are incident to v have value 1. Formula PMP_G is a particular case of $\Psi_G^{(h)}$ for $h \equiv 1$.

Lemma 6 *For all $d \in \mathbb{N}$ and for all n large enough for any set V of cardinality n and any function $h : V \to \{1, 2, \ldots, d\}$ there exists explicit construction of a graph $G(V, E)$ with the following properties: 1)V consists of two disjoint sets U and T with no edges between them; 2)The degree of every vertex $u \in U$ equals $h(u) - 1$ and the degree of every vertex $v \in T$ equals $h(v)$; 3) $|U| \geq \frac{n}{2} - 2d^2$.*

Proof. Let $n \geq 4d^2$ and the vertices $v_1, v_2, \ldots, v_n$ be arranged in non-decreasing order of $h(v_i)$. Let k be the largest number that satisfies the inequality $\sum_{i=1}^{k}(h(v_i) - 1) < \sum_{i=k+1}^{n} h(v_i) - d(d-1)$. We denote $U = \{v_1, v_2, \ldots, v_k\}$ and $T = V \setminus U$. Obviously, $|U| = k \geq n/2 - d(d-1)$. Now we construct a graph G based on the set of vertices V. We start with an empty graph and will add edges one by one. For every vertex $v \in T$ we call co-degree of v the difference between $h(v)$ and the current degree of v. From every $u \in U$ we add $h(u) - 1$ edges to G that lead from u to distinct vertices of $V \setminus U$. Doing so, we maintain degrees of all $v \in T$ under the value $h(v)$. This always can be done since by the construction of U the total co-degree of all vertices from T is greater than $d(d-1)$, hence for all big enough n there exists at least d vertices with co-degree at least 1.

While the number of vertices in T with positive co-degree is greater than d, we will choose one of those vertices $w \in T$ and add to graph exactly co-degree of w edges that connect w with other vertices from T. Finally we have that T contains at most d vertices with co-degrees at most d. Now we connect them with distinct vertices from the set U, remove that vertices from U and add them to T. It is possible that in the last step some vertex $v \in T$ is already connected with several vertices from U, in that case we should connect v with new vertices. By this operation we deleted at most d^2 vertices from U and therefore $|U| \geq n/2 - 2d^2$. □

Theorem 6 *For all $d \in \mathbb{N}$ there is such $D \in \mathbb{N}$ that for all n large enough and for any function $h : V \to \{1, 2, \ldots, d\}$, where V is a set of cardinality n, there exists such explicit graph $G(V, E)$ with maximum degree at most*

D, that formula $\Psi_G^{(h)}$ *is unsatisfiable and the size of any resolution proof for* $\Psi_G^{(h)}$ *is at least* $2^{\Omega(n)}$.

Proof. By Lemma 6 we construct a graph $G_1(V, E_1)$ and a set $U \subseteq V$ of size at least $\frac{n}{2} - 2d^2$ such that for all $v \in U$, the degree of v is equal to $h(v) - 1$ and for all $v \in V \setminus U$ the degree of v is equal to $h(v)$. Consider graph $G(U, E_2)$ from Theorem 5 with U as the set of its vertices. Define a new graph $G(V, E)$, where the set of edges E equals $E_1 \cup E_2$. Recall that edges from the set E_2 connect vertices of the set U and edges from E_1 do not connect pairs of vertices from U (that follows from the construction of the graph in Lemma 6).

For every vertex $v \in V \setminus U$ its degree equals $h(v)$. Therefore if $\Psi_G^{(h)}$ is satisfiable, then in any satisfying assignment of $\Psi_G^{(h)}$ all edges that are incident to vertices $V \setminus U$ must have the value 1. After substitution the value 1 for all these variables $\Psi_G^{(h)}$ becomes equal to the formula PMP_{G_2} that is unsatisfiable because of Theorem 5.

Formula PMP_{G_2} is obtained from $\Psi_G^{(h)}$ by substitution of several variables, thus Lemma 2 implies that the size of any resolution proof of $\Psi_G^{(h)}$ is at least the size of the minimal proof for PMP_G, that is at least $2^{\Omega(n)}$ by Theorem 5. □

4.1. Colloraries

Tseitin formulas. A Tseitin formula $T_G^{(f)}$ can be constructed by an arbitrary graph $G(V, E)$ and a function $f : V \to \{0, 1\}$; variables of $T_G^{(f)}$ corresponds to edges of G. The formula $T_G^{(f)}$ is a conjunction of the following conditions: for every vertex v we write down a CNF condition that encode that the parity of the number of edges incident to v that have value 1 is the same as the parity of $f(v)$.

Based on the function $f : V \to \{0, 1\}$ we define a function $h : V \to \{1, 2\}$ by the following way: $h(v) = 2 - f(v)$. In other words if $f(v) = 1$, then $h(v) = 1$, and if $f(v) = 0$, then $h(v) = 2$. By Theorem 6 there exists such number D, that for all n large enough it is possible to construct graph G with n vertices of degree at most D such that the size of any resolution proof of the formula Ψ_G^h is at least $2^{\Omega(n)}$.

Note that every condition corresponding to a vertex of the formula $T_G^{(h)}$ is implied from the condition corresponding to the formula Ψ_G^h. Since the resolution proof system is implication complete, every condition of $T_G^{(h)}$ may be derived from a condition of Ψ_G^h by derivation of size at most 2^D Hence all clauses of the Tseitin formula may be obtained from clauses of formula Ψ_G^h by the derivation of size $O(n)$. Thus the size of any resolution proof of $T_G^{(f)}$ is at least $2^{\Omega(n)}$. This lower bound was proved in the paper [13].

Complete graph. Let K_n be a complete graph with n vertices and $h : V \to \{0, 1, \ldots, d\}$, where d is a some constant. Let formula $\Psi_{K_n}^{(h)}$ be unsatisfiable. By Theorem 6 there exists D such that for all n large enough there exists an explicit graph G with n vertices of degree at most D that the size of any resolution proof of Ψ_G^h is at least $2^{\Omega(n)}$. The graph G can be obtained from K_n by removing of several edges, hence the formula $\Psi_G^{(h)}$ can be obtained from $\Psi_{K_n}^{(h)}$ by the substitution zeros to edges that do not present in G. Therefore by Lemma 2 the size of the resolution proof of $\Psi_{K_n}^{(h)}$ is at least $2^{\Omega(n)}$.

Acknowledgements

The authors are grateful to Vsevolod Oparin for fruitfull discussions.

References

[1] Michael Alekhnovich. Mutilated chessboard problem is exponentially hard for resolution. *Theor. Comput. Sci.*, **310(1-3)**, 513–525, January 2004.

[2] E. Ben-Sasson and A. Wigderson. Short proofs are narrow — resolution made simple. *Journal of ACM*, **48**(2) (2001), 149–169.

[3] Stefan S. Dantchev and S?ren Riis. "planar" tautologies hard for resolution. In *FOCS*, 220–229, 2001.

[4] Armin Haken. The intractability of resolution. *Theoretical Computer Science*, **39** (1985), 297–308.

[5] S. Hoory, N. Linial, and A. Wigderson. Expander graphs and their applica- tions. *Bulletin of the American Mathematical Society*, **43** (2006), 439–561.

[6] Dmitry Itsykson and Dmitry Sokolov. Lower bounds for myopic DPLL al- gorithms with a cut heuristic. In *Proceedings of the 22nd international con- ference on Algorithms and Computation*, ISAAC'11, pages 464–473, Berlin, Heidelberg, 2011. Springer-Verlag, available as ECCC Report TR12-141.

[7] S. Vadhan M. Capalbo, O. Reingold and A. Wigderson. Randomness con- ductors and constant-degree expansion beyond the degree/2 barrier. In *Pro- ceedings of the 34th Annual ACM Symposium on Theory of Computing*, (2002), 659–668.

[8] Ran Raz. Resolution lower bounds for the weak pigeonhole principle. Technical Report 01-021, Electronic Colloquium on Computational Complexity, 2001.

[9] Alexander A. Razborov. Resolution lower bounds for the weak pigeonhole principle. Technical Report 01-055, Electronic Colloquium on Computational Complexity, 2001.

[10] Alexander A. Razborov. Resolution lower bounds for the weak functional pigeonhole principle. *Theoretical Computer Science*, **303**(1) (2003), 233—243, .

[11] Alexander A. Razborov. Resolution lower bounds for perfect matching prin- ciples. *Journal of Computer and System Sciences*,**69**(1) (2004) 3–27.

[12] T. Pitassi S. Buss. Resolution and the weak pigeonhole principle. In *Pro- ceedings of the CSL97, Lecture Notes in Computer Science*, V. 1414 (1997), 149–156.

[13] A. Urquhart. Hard examples for resolution. JACM, **34**(1) (1987), 209–219. 9

[14] Alasdair Urquhart. Resolution proofs of matching principles. *Annals of Math- ematics and Artificial Intelligence*, **37**(3), 241–250, March 2003.

Discrete Time Two-Sided Mate Choice Problem with Age Preferences*

Anna A. Ivashko

Institute of Applied Mathematical Research,
Karelian Research Centre RAS, Petrozavodsk, Russia

Elena N. Konovalchikova

Transbaikal State University, Chita, Russia

We consider the two-sided mate choice model with age dependent payoffs (see Alpern, Katrantzi and Ramsey (2010) [3]). The problem is following. There are two groups of individuals: males and females. The individuals want to form a long-term relationship with a member of the other group, i.e. to form a couple. In the model males and females can form a couple during m and n periods, respectively, $m \geq n$. Each group has steady state distribution for the age of individuals. We consider asymmetric case in which the total number of unmated males is greater than the total number of unmated females.

We present a discrete time game in which the individuals from different groups are matching randomly in each period. If they accept each other, they form a couple and leave the game, otherwise they go into the next period unmated and older. It is assumed that individuals of both sexes enter the game at age 1 and stay until they are mated or males (females) pass the age m (n). The initial ratio of age 1 males to age 1 females is

*The work is supported by Russian Fund for Basic Research (project 13-01-91158-ГФEH_a, project 13-01-00033-a) and the Division of Mathematical Sciences of RAS (the program "Algebraic and combinatorial methods of mathematical cybernetics and new information system").

given. The payoff of mated player is the number of future joint periods with selected partner. Payoff of a male age i and a female age j if they accept each other is equal to $\min\{m-i+1, n-j+1\}$. The aim of each player is to maximize his/her expected payoff. In each period players use threshold strategies: to accept exactly those partners who give them at least the same payoff as the expected payoff from the next period.

Denote a_i — the number of unmated males of age i relative to the number of females of age 1 and b_j — the number of unmated females of age j relative to the number of females of age 1 ($b_1 = 1$). The vectors of the relative numbers of unmated males and females of each age $a = (a_1, ..., a_m)$, $b = (b_1, ..., b_n)$ remain constant over time. Denote the ratio of the rates at which males and females enter the adult population by R, $R = \frac{a_1}{b_1} = a_1$. The total groups of unmated males and females are $A = \sum\limits_{i=1}^{m} a_i$, $B = \sum\limits_{i=1}^{n} b_j$. Denote the total ratio $\frac{A}{B}$ by r and assume that $r > 1$.

Denote U_i, $i = 1, ..., m$ — the expected payoff of male of age i and V_j, $j = 1, ..., n$ — the expected payoff of female of age j. Players use the threshold strategies $F = [f_1, ..., f_m]$ for males and $G = [g_1, ..., g_n]$ for females, where $f_i = k$, $k = 1, ..., n$ — to accept a female of age $1, ..., k$, $g_j = l$, $l = 1, ..., m$ — to accept a male of age $1, ..., l$:

$$i \text{ accepts } j \text{ if } \min\{m-i+1, n-j+1\} \geq U_{i+1};$$
$$j \text{ accepts } i \text{ if } \min\{m-i+1, n-j+1\} \geq V_{j+1}.$$

The equilibrium age distributions are equal:

$$a_{i+1} = a_i\left(1 - \sum_{i \leftrightarrow j} \frac{b_j}{A}\right),\ i = 1, ..., m-1;$$
$$b_{j+1} = b_j\left(1 - \sum_{i \leftrightarrow j} \frac{a_i}{A}\right),\ j = 1, ..., n-1.$$

In the literature such problems are called also marriage problems or job search problems. We use here the terminology of "mate choice problem". In papers [2, 1, 7] the mutual mate choice problems with homotypic and common preferences are investigated. In [4] a continuous time model with age preferences is considered. Other two-sided mate choice models were considered in papers [5, 6, 8]. Alpern, Katrantzi and Ramsey [3] derive

properties of equilibrium threshold strategies and analyse the model for small m and n. The case $n = 2$ was considered in paper [9]. In this paper using dynamic programming method we derive the equilibrium threshold strategies and investigate players' payoffs for different values n and m.

References

[1] S. Alpern, and D. Reyniers, Strategic mating with homotypic preferences. *Journal of Theoretical Biology*, **198** (1999), 71–88.

[2] S. Alpern and D. Reyniers, Strategic mating with common preferences. *Journal of Theoretical Biology*, **237** (2005), 337–354.

[3] S. Alpern, I. Katrantzi, and D. Ramsey, Strategic mating with age dependent preferences. The London School of Economics and Political Science, 2010.

[4] S. Alpern, I. Katrantzi, and D. Ramsey, Partnership formation with age-dependent preferences. *European Journal of Operational Research*, **225** (2013), 91–99.

[5] D. Gale, and L.S. Shapley, College Admissions and the Stability of Marriage. *The American Mathematical Monthly*, **69(1)** (1962), 9–15.

[6] S.M. Kalick, and T.E. Hamilton, The matching hypothesis reexamined. *J. Personality Soc. Psychol.*, **51** (1986), 673–682.

[7] V. Mazalov, and A. Falko, Nash equilibrium in two-sided mate choice problem. *International Game Theory Review.* **10(4)** (2008), 421–435.

[8] A. Roth, and M. Sotomayor, *Two-sided matching: A study in game-theoretic modeling and analysis*. Cambridge University Press, 1992.

[9] Konovalchikova E. Model of mutual choice with age preferences. *Mathematical Analysis and Applications.* Transbaikal State University, (2012), 10–25 (in Russian).

On Vertices of Degree k in Minimal k-connected Graphs

Dmitri V. Karpov

St. Petersburg Department of V.A. Steklov Institute of Mathematics of the Russian Academy of Sciences, St. Petersburg, Russia

Definition 1 1) *A graph is called* k-connected *if* $v(G) \geq k+1$ *and* G *remains connected after deleting any its* $k-1$ *vertices.*

2) A k*-connected graph is called* minimal*, if it becomes not* k*-connected after deleting any edge.*

Clearly, all vertices of a k-connected graph have degree at least k. We denote by $v_k(G)$ the number of vertices of degree k of a graph G.

In 1967 minimal biconnected graphs were considered in the papers [1] and [2]. It can be deduced from the results of these papers that

$$v_2(G) \geq \frac{v(G)+4}{3}$$

for a minimal biconnected graph G.

In 1979 W. Mader [5, 5] has proved a very strong result that generalize for arbitrary k the one written above:

$$v_k(G) \geq \frac{(k-1)v(G)+2k}{2k-1} \tag{1}$$

for a minimal k-connected graph G. This bound is tight: there are infinite series of graphs for which the inequality (1) turns to equality.

Definition 2 *Let $k \geq 2$ and T be a tree with $\Delta(T) \leq k+1$. The graph $G_{k,T}$ is constructed from k disjoint copies $T_1, \ldots, T_k$ of the tree T. For any vertex $a \in V(T)$ we denote by a_i the correspondent vertex of the copy T_i. If $d_G(a) = j$ then we add $k+1-j$ new vertices of degree k that are adjacent to $\{a_1, \ldots, a_k\}$.*

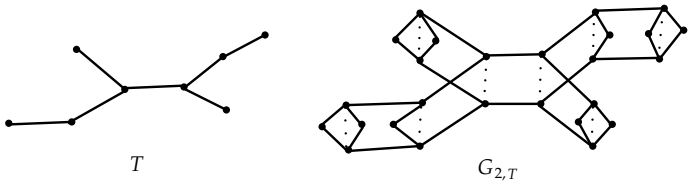

Figure 3: A tree T and correspondent extremal minimal biconnected graph $G_{2,T}$.

Clearly, if $v(T) = n$ then $v(G_{k,T}) = (2k-1)n+2$. It is not difficult to verify that $G_{k,T}$ is a minimal k-connected graph. The inequality (mad) is attained for $G_{k,T}$. A tree T with $\Delta(T) = 3$ and the graph $G_{2,T}$ are shown on the picture. The next theorem shows us that the inequality (mad) is attained only for minimal biconnected graphs of such type.

Theorem 1 *(D. Karpov, 2014.) Any minimal k-connected graph G with $v_k(G) = \frac{(k-1)v(G)+2k}{2k-1}$ is a graph $G_{k,T}$ for some tree T with $\Delta(T) \leq k+1$.*

Definition 3 *A minimal k-connected graph G is called extremal if*

$$v_k(G) = \left\lceil \frac{(k-1)v(G)+2k}{2k-1} \right\rceil. \tag{2}$$

Denote by $\mathrm{GM_k(n)}$ *the set of all extremal minimal k-connected graphs on n vertices.*

The theorem 1 shows us that $\mathrm{GM_k((2k-1)n+2)}$ consists of graphs of type $G_{k,T}$ for all trees T with $\Delta(T) \leq k+1$ and $v(T) = n$.

In 1982 Oxley [7] presented an algorithm of constructing all extremal biconnected graphs. These graphs can be obtained f by several operations of substituting a vertex of degree two by a graph $K_{2,2}$ (joint by two

edges to two vertices of the neighborhood of the vertex that have been substituted) from one of the initial graphs: K_3, three graphs of more complicated structure and two infinite series of graphs. In [11] the author has described extremal biconnected graphs in terms of the graphs $G_{2,T}$.

The situation is quite different for $k \geq 3$. In 1979 W. Mader has shown that $\mathrm{GM_k}((2\mathrm{k}-1)\mathrm{n}+4) = \varnothing$ for $k \geq 3$ and positive integer n. Moreover, Mader has conjectured that

$$\mathrm{GM_k}((2\mathrm{k}-1)\mathrm{n}+2\ell) = \varnothing$$

for $2 \leq \ell \leq k-1$ and any positive integer n. Mader has shown that $\mathrm{GM_k}((2\mathrm{k}-1)\mathrm{n}+\mathrm{t}) \neq \varnothing$ for all other positive integer $t < 2k-1$ and $n \geq 1$.

Why the case of $v(G) = (2k-1)n + 2\ell$ can be such an exclusion? Let $f(G) = (2k-1)v_k(G) - (k-1)v(G)$. It is easy to see that for an extremal minimal k-connected graph G we have that $f(G)$ is equal to the residue of $-(k-1)v(G)$ modulo $2k-1$. For $G \in \mathrm{GM_k}((2\mathrm{k}-1)\mathrm{n}+2\ell)$ that is $f(G) = \ell - 1$. Thus the Mader's conjecture means that if $f(G) > 0$ for a k-connected graph then $f(G) \geq k-1$.

We prove a particular case of Mader's conjecture.

Theorem 2 *(D. Karpov) Let k, n, ℓ be positive integers such that $k \geq 3$, and $2 \leq \ell < \frac{4k+7+4\sqrt{k^2-k-2}}{9}$. Let G be a mininal k-connected graph with $v(G) = (2k-1)n + 2\ell$. Then*

$$v_k(G) \geq \left\lceil \frac{(k-1)v(G) + 2k}{2k-1} \right\rceil + 1. \tag{3}$$

One can easily verify that $\frac{4k+7+4\sqrt{k^2-k-2}}{9} \geq \frac{8k+3}{9}$ for $k \geq 3$. That is we have proved Mader's conjecture for $\ell \leq \frac{8k+3}{9}$.

References

[1] G. A. Dirac. Minimally 2-connected graphs. *J. reine and angew. Math.* Vol. 268 (1967), 204-216.

[2] M. D. Plummer. On minimal blocks. *Trans. Amer. Math. Soc.*, Vol. 134 (1968), 85-94.

[3] W. T. Tutte. *Connectivity in graphs.* Toronto, Univ. Toronto Press, 1966.

[4] W. T. Tutte. A theory of 3-connected graphs. *Indag. Math.*, Vol. 23 (1961), 441-455.

[5] W. Mader. On vertices of degree n in minimally n-connected graphs and digraphs. Combinatorics, Paul Erdös is Eighty, Budapest, Vol.2 (1996) 423-449.

[6] W. Mader. Zur Struktur minimal n-fach zusammenhängender Graphen. *Abh. Math. Sem. Univ. Hamburg*, Vol. 49 (1979), 49-69.

[7] J. G. Oxley. On some extremal connectivity results for graphs and matroids. *Discrete Math.*, Vol.41 (1982), 181-198.

[8] W. Hohberg. The decomposition of graphs into k-connected components. *Discr. Math.*, **109** (1992), 133-145.

[9] D. V. Karpov, A. V. Pastor. On the structure of a k-connected graph. *Zap. Nauchn. Sem. S.-Peterburg. Otdel. Mat. Inst. Steklov. (POMI)* **266** (2000), 76-106, in Russian. English translation *J. Math. Sci. (N. Y.)* **113** (2003), no. 4, 584-597.

[10] D. V. Karpov. Separating sets in a k-connected graph. *Zap. Nauchn. Sem. S.-Peterburg. Otdel. Mat. Inst. Steklov. (POMI)*, **340** (2006), 33-60, in Russian. *English translation J. Math. Sci. (N. Y.)* **145** (2007), no. 3, 4953-4966.

[11] D. V. Karpov. Minimal biconnected graphs. *Zap. Nauchn. Sem. S.-Peterburg. Otdel. Mat. Inst. Steklov. (POMI)*, **417** (2013), 106-127.

[12] D. V. Karpov. The tree of decomposition of a biconnected graph. *Zap. Nauchn. Sem. S.-Peterburg. Otdel. Mat. Inst. Steklov. (POMI)*, **417** (2013), 86-105.

On the Number of Trees of a Giving Size in a Conditional Poisson Galton-Watson Forest*

Elena V. Khvorostyanskaya

Institute of Applied Mathematical Research,
Karelian Research Centre RAS, Petrozavodsk, Russia

Let a subcritical or critical homogeneous Galton–Watson process start with N particles and the number of offspring of each particle have a Poisson distribution with parameter λ. The set of this process realizations is infinite and consists of rooted trees with a finite number of vertices. The branching process induces probability distribution on this set. Such random forests are known as Galton–Watson forests. Let μ_r denote the number of trees of a giving size in the Galton–Watson forest. As $N \to \infty$ for a subset of trajectories with a known identical number of vertices limit distributions of μ_r were obtained in [2] using a generalized allocation scheme [1]. We derived similar results for a subset of trajectories such that the number of vertices does not exceed n with different behavior of parameters λ and n. In particular, the following assertion is true.

Theorem. *Let $N \to \infty$ and one of the following conditions be fulfilled:*

1) $r \to \infty$, $N(1-\lambda) \to \gamma$, $0 \leqslant \gamma < \infty$, $n/N^2 \geqslant C > 0$;

2) $r \to \infty$, $\lambda \geqslant \lambda_1 > 0$, $N(1-\lambda) \to \infty$, $\sqrt{1-\lambda}\,(N - n(1-\lambda)) \leqslant C\sqrt{N}$, $C \geqslant 0$;

3) $r \geqslant 3$, $\lambda \to 0$, $N\lambda^3 \to \infty$, $N - n(1-\lambda) \leqslant C\sqrt{\lambda N}$, $C \geqslant 0$;

4) $r = 2$, $\lambda \to 0$, $N\lambda^6 \to \infty$, $(n(1-\lambda) - N)/\sqrt{\lambda N} \to \infty$.

*The work was supported by the Russian Foundation for Basic Research, grant 13-01-00009.

Then

$$\mathbf{P}\{\mu_r = k\} = \frac{(Np_r)^k}{k!} e^{-Np_r}(1 + o(1))$$

uniformly on $(k - Np_r)/\sqrt{Np_r}$ *in any fixed interval.*

For the conditional random forests in question the proved theorems generalize the results obtained in [3].

References

[1] Yu. L. Pavlov. Limit theorems for the number of trees of a giving size in a random forest. *Mathematics of the USSR–Sbornik.* Vol. 32, N 3 (1977), 335–345.

[2] V. F. Kolchin. *Random mappings.* 1986. Springer, New York, 206 p.

[3] A. N. Chuprunov, I. Fazekas. An analogue of the generalized allocation scheme: limit theorems for the number of cells containing a given number of particles. *Discrete Mathematics and Applications.* Vol. 22, N 1 (2012), 101–122.

On Application of the Probabilistic Method to Analysing the Partitions of an Integer

Andrei V. Kolchin

Steklov Institute of Mathematics, Moscow, Russia

Abstract

In this research, we demonstrate how to apply the probabilistic approach to investigating the asymptotic behaviour of the number of partitions of an integer.

Any representation of a positive integer in the form of a sum of positive integers (parts) is referred to as a partition of the number, which is primarily of combinatorial and number theoretical nature. Questions concerning the partitions have played an important part in mathematics.

Gian-Carlo Rota wrote in his preface to the monograph [1] that the theory of partitions is one of the very few branches of mathematics that can be appreciated by anyone who is endowed with little more than a lively interest in the subject. Its applications are found wherever discrete objects are to be counted or classified, whether in the molecular and the atomic studies of matter, in the theory of numbers, or in combinatorial problems from all sources.

Partitions are investigated in combinatorics and in the theory of numbers; classical combinatorial problems concern counting partitions, and in the theory of numbers, problems on additive representations of numbers are being solved under arithmetical constraints imposed on the parts (for example, the well-known Goldbach and Waring problems). Serious difficulties may arise while solving problems on partitions, though; so a great body of special methods in the theory of partitions have been elaborated

(see, e.g., [1]). Historically, the first method which has since then become the most common in the whole theory of partitions is the method of generating functions. It was developed by Euler and has found application both in the theory of numbers and in combinatorics; it has been evolved into very delicate but universal tools utilising the Dirichlet generating functions, the trigonometric sums, the characteristic functions.

Euler indeed laid the foundations of the theory of partitions. Many of the other great mathematicians—Cayley, Gauss, Hardy, Jacobi, Lagrange, Legendre, Littlewood, Rademacher, Ramanujan, Schur, and Sylvester—have contributed to the development of the theory.

If one considers the applications of partitions in various branches of mathematics, one is struck by the interplay of combinatorial and asymptotic methods.

We consider the problem on the number of partitions of a positive integer n into s positive integer parts which do not exceed a given integer r (those partitions which differ in the order of parts only are counted as one). Let $C_{n,s,r}$ stand for the number of these partitions. A series of exact and asymptotic formulas have been known for this number (see, e.g., [1]).

It turns out that there is an easy way to arrive at a compact asymptotic expression for the number of these partitions with the use of probabilistic reasoning (see, e.g., [2, 3]). It is not difficult to see, indeed, that the partition of a number is described by the classical scheme of equiprobable allocation of n particles into s cells under the condition that each cell contains not more than r particles.

Let $\xi_1, \xi_2, \ldots, \xi_s$ be independent random variables which take values $1, 2, \ldots, r$ with equal probabilities, which can be considered as the contents of the corresponding cells in the above allocation scheme. It is easily seen that

$$\begin{aligned}\mathbf{P}\{\xi_1 + \xi_2 + \cdots + \xi_s = n\} &= \sum_{\substack{k_1+k_2+\cdots+k_s=n \\ k_1,k_2,\ldots,k_s \le r}} \mathbf{P}\{\xi_1 = k_1, \xi_2 = k_2, \ldots, \xi_s = k_s\} \\ &= C_{n,s,r}\frac{1}{r^s}. \end{aligned} \tag{1}$$

In addition, the equalities

$$\mathsf{E}\xi_1 = \frac{r+1}{2} = m,$$
$$\mathsf{Var}\,\xi_1 = \frac{r^2-1}{12} = \sigma^2$$

hold true. So, in order to investigate the asymptotic behaviour of $C_{n,s,r}$ one is able to utilise the well-developed apparatus of local limit theorems of probability theory (see, e.g., [4, 5, 6]).

First, let us analyse the behaviour of the number of partitions in the so-called 'central' domain of variation of the parameters.

It is obvious that

$$\mathsf{P}\{\xi_1+\xi_2+\cdots+\xi_s = n\} = \mathsf{P}\left\{\frac{\xi_1+\xi_2+\cdots+\xi_s - sm}{\sigma} = \frac{n-sm}{\sigma}\right\}.$$

Let

$$x = \frac{n-sm}{\sigma}.$$

Using the above relations for the mathematical expectation m and the variance σ^2, we find that

$$x = \frac{2n - s(r+1)}{\sqrt{(r^2-1)/3}}.$$

For the sake of simplicity, let r be fixed. From relation (1), with the use of the local convergence to the standard normal law we see that for fixed r, while the parameters n and s tend to infinity in such a way that the ratio of n to s lies in some finite interval, the relation

$$\frac{1}{r^s}C_{n,s,r} = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}(1+o(1))$$

holds true uniformly with respect to all x in an arbitrary fixed finite interval. Thus, the following assertion is valid.

Theorem *Let r be a fixed integer. If the parameters n and s tend to infinity in such a manner that the ratio of n to s remains inside some*

finite interval, then the number $C_{n,s,r}$ of partitions of a positive integer n into s positive integer parts not exceeding r obeys the equality

$$C_{n,s,r} = \frac{r^s}{\sqrt{2\pi}} e^{-x^2/2}(1 + o(1)),$$

which holds true uniformly in all

$$x = \frac{2n - s(r+1)}{\sqrt{(r^2-1)/3}}$$

which fall into an arbitrary fixed finite interval.

The author believes that the analysis of behaviour of this number of partitions in other domains of variation of the parameters n, s, r certainly deserves a separate presentation.

In conclusion, it is worth noticing that the reverse approach which consists of studying asymptotic properties of characteristics of a classical allocation scheme with the use of apparatus of the theory of partitions of a number seems to be very fruitful as well.

References

[1] G. E. Andrews, *The Theory of Partitions*. Addison–Wesley, Reading, Mass., 1976.

[2] J. Spencer, P. Erdős, *Probabilistic Methods in Combinatorics*. Akadémiai Kiadó, Budapest, 1974.

[3] N. Alon, J. Spencer, *The Probabilistic Method*. Wiley, New York, 1992.

[4] B. V. Gnedenko, A. N. Kolmogorov, *Limit Distributions for Sums of Independent Random Variables*. Addison–Wesley, Cambridge, 1954.

[5] V. V. Petrov, *Limit Theorems of Probability Theory. Sequences of Independent Random Variables*. Clarendon Press, Oxford, 1995.

[6] V. F. Kolchin, *Asymptotic Methods in Probability Theory*. MIEE, Moscow, 1988 (in Russian).

On a Scheme of Allocation of Distinguishable Particles into Indistinguishable Cells

Andrei V. Kolchin, Valentin F. Kolchin, and Natalia Yu. Enatskaya

Steklov Institute of Mathematics, Moscow, Russia

Abstract

In the framework of a scheme of allocating distinguishable particles into indistinguishable cells, upon defining an easy-to-use representation form of an outcome, we explicitly enumerate the outcomes; solve the direct and reverse problems of enumeration of the outcomes; find the probability distribution for the outcomes; give a recurrence relation for the number of outcomes of the scheme and their probabilities under the condition that the number of nonempty cells is fixed; derive an explicit formula for the total number of the outcomes of the scheme; analyse the distribution of the statistics of empty cells; present a numerical method to find their number; and suggest various methods to simulate the outcomes of the scheme which allow us to carry out an approximate calculation of the number of outcomes of the scheme.

We discuss combinatorial and probabilistic problems in the framework of the scheme of allocating distinguishable particles into indistinguishable cells and similar schemes with certain constraints imposed on the distribution of particles.

By the combinatorial problems are primarily meant those where one has to find the number of outcomes in the basic scheme under consideration and schemes with constraints, as well as to give a visual representation of all outcomes in an effort to simplify the analysis.

By the probabilistic problems we mean to find probability distributions of outcomes or groups of outcomes.

The combinatorial problems are solved in three forms: to find exact analytical formulas for the characteristics of the scheme we are interested in; to construct numerical (algorithmic) methods to evaluate them under given parameters of the scheme; to suggest approximate methods to solve combinatorial problems by means of stochastic simulation.

The basis of the attempts of exact (analytical and numerical alike) calculation of the number of outcomes of a scheme is the visual representation (enumeration) of the outcomes; we suggest two convenient forms to represent the outcomes of the scheme keeping in mind that they are determined by the contents of the cells only with no account for their arrangement because the cells are indistinguishable.

In order to enumerate the outcomes of the scheme we construct a random process of successive one-by-one equiprobable allocation of r particles to n cells in the context of the scheme under consideration. The step of the process consists of allocating the next particle. Upon introducing a certain numeration of the outcomes at each allocation step, at the last rth step we enumerate all outcomes of the scheme. We thus establish a one-to-one correspondence between the outcomes and the labels they get assigned under our numeration; so we obtain an explicit formula for the number of outcomes of the scheme and are able to implement their fast simulation. In order to analyse this random process we draw its state transition graph.

The problem to find the probability distribution of the outcomes of the scheme is numerically solved for any fixed values of the scheme parameters by enumerating all outcomes with the use of the above graph.

We suggest three ways to simulate the outcomes of the scheme:

1. we discard the excessive outcomes while simulating as in [1] the more general scheme of allocation of distinguishable particles into distinguishable cells (with repetitions allowed);

2. with the help of the labelling method described in [1], knowing the probability distribution of the outcomes we randomly draw the outcome label and use the correspondence between the outcomes and their labels;

3. we directly simulate the outcomes by drawing at random the minimum label assigned to the elements in a cell.

Step-by-step algorithms can be given for each of these ways.

The procedure of approximate calculation of the number of outcomes S_2 of the scheme is based on rejecting the excessive outcomes in the more general scheme with S_1 outcomes for which a fast and efficient simulation algorithm is known. As a more general scheme we suggest the scheme of allocation of r distinguishable particles to n distinguishable cells with repetitions allowed, where any cell can hold all particles. It is known that the number of outcomes of this scheme is $S_1 = n^r$, while a quite good algorithm to simulate its outcomes is given in [1].

References

[1] Enatskaya N. Yu. and Khakimullin E. R., *Stochastic Simulation.* Moscow Institute of Electronics and Mathematics, 2012.

On Relation of Linear Diophantine Equation Systems with Commutative Grammars

Dmitry G. Korzun

Petrozavodsk State University, Petrozavodsk, Russia

Let $\mathbb{Z}$ and $\mathbb{Z}_+$ be the set of integers and nonnegative integers, respectively. A homogenous linear Diophantine equation system consists of n equations in m nonnegative unknowns,

$$Ax = \mathbb{O}, \quad \text{where } A \in \mathbb{Z}^{n \times m},\ x \in \mathbb{Z}_+^m. \tag{1}$$

A is the coefficient matrix and x is the column of unknowns.

A solution to (1) is irreducible if it is not a sum of two non-zero solutions to the same system. The set $\mathcal{H}$ of all irreducible solutions to (1) is called Hilbert basis, which is unique and finite. The general solution to (1) is

$$x = \sum_{h \in \mathcal{H}} c_h h \ \text{ for some } \ c_h \in \mathbb{Z}_+.$$

Moving terms with negative coefficients in each equation to another side, we rewrite (1) with nonnegative matrices A' and A'' (i.e., $A = A' - A''$, $A', A'' \geq \mathbb{O}$):

$$\sum_{j=1}^{m} a'_{ij} x_j = \sum_{j=1}^{m} a''_{ij} x_j, \quad i = 1, 2, \ldots, n, \tag{2}$$

where $\min\{a'_{ij}, a''_{ij}\} = 0$ for any i, j.

In our previous work [1] (and references therein), we studied restrictions to A'' that leads to a mapping between systems (1) and commutative

context-free grammars [2]. In this case, (1) can be used as a tool for modeling the topology of computer networks [3, 4]. In particular, Hilbert basis provides a compact description of the routes structure, although the number of possible network paths can be essentially large.

In this talk we consider (2) with no additional restriction to A''. We present formal construction of a commutative grammar for arbitrary system (2). In contrast with the previously studied cased, such a grammar is context-sensitive in general case. Grammatical derivations map to solutions to (2).

This mapping can be described in terms of a directed hypergraph where vertices are grammar nonterminals (n equations) and hyperarcs are grammar rules (m unknowns). In such a graph, a basis solution to (2) represents a cyclic-like structure with certain minimality properties.

References

[1] D. Korzun. Syntactic methods in solving linear Diophantine equations. *Annual Finnish Data Processing Week at Petrozavodsk State University (FDPW 2004): Advances in Methods of Modern Information Technology.* Vol. 6, 151–156.

[2] J. Esparza. Petri nets, commutative context-free grammars, and basic parallel processes. *Fundamenta Informaticae*, Vol. 30 (1997),23–41.

[3] D. Korzun and A. Gurtov. A Diophantine model of routes in structured P2P overlays. *ACM SIGMETRICS Performance Evaluation Review.* Vol. 35, Issue 4 (2008), 52–61.

[4] Ю.А. Богоявленский, К.А. Кулаков, Д.Ж. Корзун. Линейные диофантовы модели восстановления соединений в сетях MPLS. Информационные технологии, №3 (2011), 7–13.

Minimum Number of Input Clues in an Associative Memory

Ville Junnila, Tero Laihonen

Department of Mathematics and Statistics, University of Turku
Turku, Finland

In a recent article by Yaakobi and Bruck (2012), the question how stored information can be efficiently retrieved from associative memories was studied. An associative memory is modeled by a graph $G = (V, E)$. The vertices represent the stored information units and the edges between them define the associations of information units to each other. Two distinct vertices $x, y \in V$ are *t-associated* if the graphic distance $d(x, y) \leq t$. Let us denote by $B_t(x)$ the closed t-neighborhood of x. For any $T \subseteq V$, denote

$$S_t(T) = \bigcap_{c \in T} B_t(c).$$

Let m be a positive integer. A *reference set* is a subset $C \subseteq V$. We retrieve an information unit $x \in V$ from the associative memory in the following manner. A set of m distinct elements $T = \{c_1, \ldots, c_m\} \subseteq C$ t-associated with x, called *input clues*, are given. As an output set we receive $S_t(T)$. The smaller the output set is, the more precisely we know the sought information unit. The maximum size of an output set over any $T \subseteq C$ with $|T| = m$ is called the *uncertainty*. Naturally, we wish to set an upper bound, say N, on the uncertainty. We also need to make sure that we have access to every information unit $x \in V$. This requires that $|I_t(x)| \geq m$ where

$$I_t(x) = B_t(x) \cap C.$$

This gives rise to the following definition.

Definition 1 *We say that a pair (G, C) is a (t, m, N)-*associative memory with the reference set C *if*

(i) $|I_t(x)| \geq m$ *for any* $x \in V$ *and*

(ii) $|S_t(T)| \leq N$ *for any subset* $T \subseteq C$ *of size* m.

In this paper, we focus on unambiguous output by setting $N = 1$. We give bounds on the minimum number of input clues needed and optimal constructions attaining the bounds. Moreover, we discuss the maximum number of memory entries when m and $|C|$ are given. We also investigate the problem of forced vertices, i.e., those vertices which must belong to C.

Forest Fire Models on Configuration Random Graphs*

Marina M. Leri, Yuri L. Pavlov

Institute of Applied Mathematical Research,
Karelian Research Centre RAS, Petrozavodsk, Russia

We consider two types of configuration models [1, 2] of random graphs. Let graph nodes be numbered from 1 to N and node degrees $\xi_1, \xi_2, \ldots, \xi_N$ be drawn independently from one of the two given distributions: power-law (1) or Poisson (2).

$$\mathbf{P}\{\xi \geq k\} = k^{-\tau}, \qquad k = 1, 2, \ldots, \ \tau > 1, \tag{1}$$

$$\mathbf{P}\{\xi = k\} = \frac{\lambda^k}{k!} e^{-\lambda}, \qquad k = 1, 2, \ldots, \ \lambda > 0. \tag{2}$$

For each graph node the number of stubs (or half edges) is defined by a given distribution. To form a graph all the stubs are joined one to another with equal probability to form links. If the sum of node degrees is odd one stub is added to a random node in order to form a lacking link.

One of the important trends in random graphs' field has been the study of graphs robustness to different types of breakdowns [1, 3, 4]. In this work we consider random graphs resilience to random and to targeted node destructions from a viewpoint of node saving. This approach has appeared as modelling of forest fires [5, 6] as well as banking system defaults so that to minimize their negative effects [7].

Let view graph nodes as trees on a confined area of a real forest, placing them in the nodes of a square lattice sized 100×100. Nodes are connected in a closest neighbour manner in a way that a link existence means the fire propagation between neighbouring nodes. A relation between an average node degree m and the parameters of node degree distributions is the

*The work was supported by the Russian Foundation for Basic Research, grant 13-01-00009.

following: $m = \zeta(\tau) = \lambda$, where $\zeta(x)$ is the Riemann zeta function. Thus, we consider random graphs sized $3000 \leq N \leq 10000$ and $m \leq 8$ (as a fully packed lattice gives every inner node 8 adjacent neighbours).

Fire starts either from an equiprobably chosen node (random fire start) or from a node with the highest degree (target fire start) spreading onto neighbouring nodes with a probability $0 < p \leq 1$. The study aims at finding the best topology of configuration random graph that ensures maximum survival of nodes in case of a fire.

By computer simulations we considered forest fire models on both graph types (power-law and Poisson) in two fire start cases: random (when the fire starts from an equiprobably chosen node) and target (when fire starts from a node with the highest degree). In each situation we found the optimal values of node degree distribution parameters (τ for power-law and λ for Poisson) that ensure maximum survival of graph nodes. The results showed that both power-law and Poisson random graphs are more robust in case of a random fire start than at targeted attack on a node with the highest degree. We also compared considered graphs (power-law and Poisson) from a viewpoint of the number of survivor nodes under the same initial conditions: the values of N and p. Thus, when a fire starts "randomly" more nodes will survive if node degrees follow the power-law distribution rather than Poisson distribution. However, if a fire starts from a node with the highest degree the topology that will give the highest node survival depends on both the fire transition probability p and the initial graph size N. If the following condition is true:

$$N \geq 2088.1 + \frac{894.2}{p}, \tag{3}$$

the power-law topology will ensure more nodes to survive in case of a target fire start, otherwise a Poisson node degree distribution will give better results.

References

[1] R. Durrett, *Random Graph Dynamics.* Cambridge: Cambridge Univ. Press. 2007.

[2] R. Hofstad, *Random Graphs and Complex Networks.* Eindhoven University of Technology. 2011.

[3] R. Cohen, K. Erez, D. Ben-Avraham, S. Havlin, Resilience of the Internet to Random Breakdowns. *Phys. Rev. Lett.* Vol. 85 (2000), 4626–4628.

[4] I. Norros, H. Reittu, Attack resistance of power-law random graphs in the finite mean, infinite variance region. *Internet Mathematics.* Vol. 5, N 3 (2008), 251–266.

[5] J. Bertoin, Fires on trees. *Annales de l'Institut Henri Poincare Probabilites et Statistiques.* Vol. 48, N 4 (2012), 909–921.

[6] B. Drossel, F. Schwabl, Self-organized critical forest-fire model.*Phys. Rev. Lett.* Vol. 69 (1992), 1629–1632.

[7] N. Arinaminparty, S. Kapadia, R. May, Size and complexity model financial systems. *Proceedings of the National Academy of Sciences of the USA.* Vol. 109 (2012), 18338–18343.

Subgroup Nash Equilibrium and Communication for S5n-Knowledge

Takashi Matsuhisa

Department of Natural Science, Ibaraki National College of Technology
Ibaraki, Japan

Recently, researchers in economics, AI, and computer science become entertained lively concerns about relationships between knowledge and actions. At what point does an economic agent sufficiently know to stop gathering information and make decisions? There are also concerns about cooperation and knowledge. What is the role of sharing knowledge to making cooperation among agents?

Considering a coalition among agents, we tacitly understand that each agents in the coalition share their individual information and so they commonly know each other. In mathematical point of view yet a little is known what structure they have to know commonly. The aim of this paper is to fill the gap. Our point is that in a coalition, the members does not necessary have common-knowledge to each others but they communicate his/her own beliefs on the others to each other through messages.

The purposes of this paper are to introduce the concept of Subgroup Nash equilibrium of a strategic game, and to show that a communication among the players in a coalition leads to the equilibrium through messages. A Subgroup Nash equilibrium for a strategic game consists of (1) a subset S of players, (2) independent mixed strategies for each member of S, (3) the conjecture of the actions for the other players not in S with the condition that each member of S maximizes his/her expected payoff according to the product of all mixed strategies for S and the other players' conjecture.

This paper analyses the solution concept from the Bayesian point of view: The players start with the same prior distribution on a state-space. In addition they have private information which is given by a partition of

the state space. Each player in a coalition S predicts the other players' actions as the posterior of the others' actions given his/her information. He/she communicates privately their beliefs about the other players' actions through messages among all members in S according to the communication network in S, which message is information about his/her individual conjecture about the others' actions. The recipients update their belief by the messages. Precisely, at every stage each player communicates privately not only his/her belief about the others' actions but also his/her rationality as messages according to a protocol and then the recipient updates their private information and revises her/his prediction.

In this circumstance, we shall show that

Main theorem. *Suppose that the players in a strategic form game have the knowledge structure associated a partitional information with a common prior distribution. In a communication process of the game according to a protocol with revisions of their beliefs about the other players' actions, the profile of their future predictions converges to a Subgroup Nash equilibrium of the game in the long run.*

Generating Functions and Cooperation in Communication Networks*

Vladimir V. Mazalov

Institute of Applied Mathematical Research,
Karelian Research Centre RAS, Petrozavodsk, Russia

Ludmila I. Truhina

Chita Branch of Baikal Institute of Economics and Low, Chita, Russia

We study a cooperative game determined on an undirected graph. The characteristic function is determined by special way taking into account the number of connections and the path lengths between nodes in the graph. An allocation imputation procedure is proposed and it is proven that it coincides with the Myerson value. We introduce the algorithm based in generating functions for computing the Myerson value in communication networks. Examples of applying the allocation rule for analysis of the transportation networks are provided.

*The research was supported by the grant of the Russian Fund for Basic Research (project 13-01-91158) and the Division of Mathematical Sciences of the Russian Academy of Sciences.

Undecidability for Integer Weighted Büchi Automata and Robot Games with States

Vesa Halava, Tero Harju, Reino Niskanen

Department of Mathematics and Statistics, University of Turku
Turku, Finland

Igor Potapov

Department of Computer Science, University of Liverpool, UK

1. Introduction

In the universality problem we are given an automaton $\mathcal{A}$ accepting words over an alphabet A, and it is to determine whether or not $\mathcal{A}$ accepts all possible inputs over A. In this paper we consider the universality problem for integer weighted finite automata accepting infinite words. The model of automata to be considered has a finite number of states, and acceptance is defined by existence of a path (or a computation) that visits a final state infinitely often. The requirement of the acceptance is that in *Büchi automata*; see [1]. The universality problem is known to be decidable for the Büchi automata. Indeed, this follows from the facts that the family of languages accepted by Büchi automata is closed under the complementation and the emptiness problem is decidable for Büchi automata; see, e.g., Thomas [7].

We shall prove that the universality problem is undecidable for Büchi automata having weights on the transitions from the additive group of integers. The weight of a path is the sum of weights of its transitions. We shall prove the undecidability result even for automata having merely three states. In our automata each transition of the underlying automaton may have several copies with different weights.

We shall use the undecidability of the universality problem to prove that deciding whether a Robot Game with states has a winning strategy is also undecidable.

We prove that the universality problem is undecidable for integer weighted Büchi automata by reducing the instances of the *infinite Post Correspondence Problem*, or the ω PCP, to the universality problem. The ω PCP is a natural extension of the Post Correspondence Problem. In ω PCP morphisms g, h are given and the solution is an infinite word w such that for every finite prefix p of w either $h(p) < g(p)$ or $g(p) < h(p)$. The ω PCP was shown to be undecidable for instances of size 9 in [4]. The proof uses a reduction from the termination problem of the semi-Thue systems proved to be undecidable for the 3-rule semi-Thue systems from [5]. Later in [2] the ω PCP was proved to be undecidable for instances of size 8, using the same ideas but also with an encoding that helped to decrease the number of letters. In both articles, the possible solution is of a particular form and this form has a significant role in our proof. In particular if w is not a solution, then the first position where $h(w)$ and $g(w)$ differ (called the *error*) is reached in $h(w)$ at least one letter (of w) earlier than it is reached in $g(w)$.

2. The weighted Büchi automaton

For the proof of undecidability of the universality problem for weighted finite automata, for each instance (h, g) of the ω PCP, we need to construct an integer weighted Büchi automaton $\mathcal{A}^\gamma$ such that its language $L(\mathcal{A}^\gamma) \neq A^\omega$ if and only if the instance (h, g) has an infinite solution. This is done by constructing the automaton in such way that word w has zero weight if and only if there is an error in w. This is done by having special edges for error guessing and error verifying.

Theorem 1 *It is undecidable whether or not $L(\mathcal{A}^\gamma) = A^\omega$ holds for 3-state integer weighted Büchi automata $\mathcal{A}^\gamma$ over its alphabet A.*

We can modify the automaton of previous Theorem such that all of its states are final.

Corollary 1 *It is undecidable whether or not $L(\mathcal{B}^\gamma) = A^\omega$ holds for 3-state integer weighted Büchi automata $\mathcal{B}^\gamma$, where each state is final, over its alphabet A.*

3. Application to Robot Games with states

We use Theorem 1 to prove undecidability of finding a winning strategy in Robot Games with states. The idea is that one player gives input letters one by one and the other player has to match them according to the automaton. The first player has a way to check whether the second player chose a correct letter and he wins if that did not happen. On the other hand, if the first player checks and the letter is properly matched, the second player wins.

A *Robot Game with states* consists of two players, *Attacker* and *Defender* having sets of vectors $U_1, \ldots, U_r \subseteq \mathbb{Z}^n$ and $V_1, \ldots, V_s \subseteq \mathbb{Z}^n$, finite automata $\mathcal{A}$ and $\mathcal{B}$, respectively. An *initial vector* $\mathbf{x}_0$ of the game is given. For each automaton, there is a bijective mapping from transitions to vectors in vector sets. That is, the vectors players can play are represented by transitions in the underlying automaton.

The game goes on in the following way. Starting from $\mathbf{x}_0$ players add a vector from a respective set, determined by the underlying finite automaton, to the current position of the game in turns. Attacker tries to reach the origin while Defender tries to keep Attacker from reaching the origin. Note that it is possible that $\mathbf{x}_0 = (0, \ldots, 0)$, in this case Attacker does not trivially win the game.

The task is to determine whether or not there exists a winning strategy for Attacker, i.e., can Attacker reach the origin regardless of the vectors Defender chooses during his turns.

We construct a 2-dimensional Robot Game with states such that deciding whether Attacker has a winning strategy is undecidable. The initial vector of the game is $\mathbf{x}_0 = (0, 0)$. Attacker's automaton is a modification of automaton of Corollary 1. The special checking state is added which allows Attacker to win if Defender makes a mistake. The first component of his vectors is used to simulate $\mathcal{B}^\gamma$ while with the second component he has to match vectors played by Defender. Defender's automaton does not alter the first component.

It can be proved that Attacker has to match the second component of the vector $(0, -i)$ played by Defender. On the other hand, Defender has to keep playing vectors $(0, -i)$ if Attacker has matched them. Thus the first component of the configuration gives us the following undecidability result.

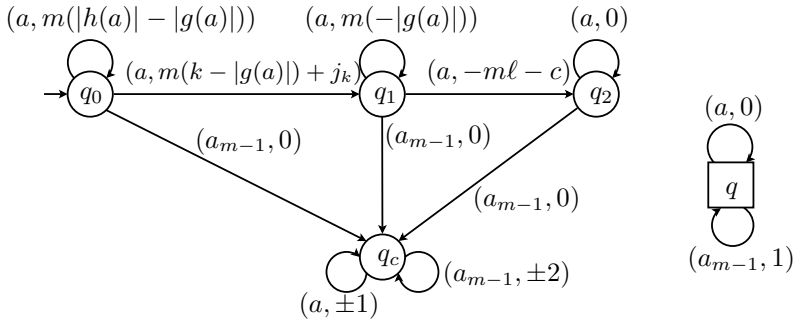

Figure 4: The picture of the weighted automata $\mathcal{A}$ and $\mathcal{B}$. In the figure $a \in A$ and $|A| = m - 1$, where A is the alphabet used in the proof of Theorem 1.

Theorem 2 *It is undecidable whether Attacker has a winning strategy in 2-dimensional Robot Game with states.*

References

[1] J.R. Büchi. *On a decision method in restricted second order arithmetic.* In Proc. International Congress on Logic, Method, and Philosophy of Science. 1–12, 1960.

[2] J. Dong, Q. Liu. Undecidability of Infinite Post Correspondence Problem for Instances of Size 8. *RAIRO–Theor. Inf. Appl.* **46** (2012), 451-457.

[3] V. Halava, T. Harju. Undecidability in Integer Weighted Finite Automata. *Fund. Inf* 34 (1999),189-200.

[4] V. Halava, T. Harju. Undecidability of Infinite Post Correspondence Problem for Instances of Size 9. *RAIRO–Theor. Inf. Appl.* **40** (2006), 551-557.

[5] Y. Matiyasevich, G. Sénizergues. Decision problems for semi-Thue systems with a few rules. *Theor. Comput. Sci.* 330(1) (2005),145-169.

[6] E. Post. A variant of recursively unsolvable problem. *Bull. of Amer. Math. Soc.* 52 (1946), 264-268.

[7] W. Thomas. *Automata on infinite objects.* in Van Leeuwen, Ed., *Handbook of Theoretical Computer Science*, pp. 133-164, Elsevier, 1990.

About Vertices of Degree 6 of C_3-critical Minimal 6-connected Graph

Alexei V. Pastor

St.Petersburg Department of V.A.Steklov Institute of Mathematics of the Russian Academy of Sciences
St.Petersburg, Russia

The notion of C_k-critical n-connected graph was introduced by W. Mader in [5]. This notion is a natural generalization of the notion of contraction-critical n-connected graph: it means that for any $\ell \leq k$ if some ℓ vertices is a clique, than there is a n- vertices cutset contains all this ℓ vertices. In a partial case of $k = 3$ it means, that contraction of any edge reduce connectivity of graph by 1 and contraction of any triangle reduce the connectivity by 2.

The problem of bounding the number of vertices of degree n in a minimal and contraction-critical n-connected graph is well-known. First time this question was asked by R. Halin in [2]. For $n = 4$ it was found by N. Martinov [6] and, independently, by M. Fontet [3] that for any contraction-critical 4-connected graph all of its vertices has degree 4. For $n = 5$ the best known bound was found by Ando at al. [1] and, independently, by S. A. Obraztsova and A. V. Pastor (RuFiDim-2012): at least 2/3 of vertices of contraction-critical minimal 5-connected graph has degree 5. For $6 \leq n \leq 10$ it was proved by S. A. Obraztsova and A. V. Pastor in [7, 8, 9], that at least 1/2 of vertices of such graph has degree n. For $n > 10$ the best known bound is the result of W. Mader [4]: any minimal n-connected graph contains at least $\frac{n-1}{2n-1}$ vertices of degree n.

The problem of bounding the number of vertices of degree n in a C_3-critical minimal n-connected graph is a natural generalization of the previous problem. It was proved in [5], that any C_3-critical graph is 6-connected. So the case $n < 6$ is trivial. In this paper we research the

structure of C_3-critical minimal 6-connected graph G and its subgraph G_6 induced on the vertices of degree 6. We prove that any connectivity component of G contains a cycle, and by a corollary we prove that any C_3-critical minimal 6-connected graph contains at least 5/9 vertices of degree 6.

References

[1] K. Ando, T. Iwase. The number of vertices of degree 5 in a contraction-critically 5-connected graph, *Discrete Mathematics*, **311** (2011), N. 17, 1925-1939.

[2] R. Halin. On the structure of n-connected graphs, In:*Recent Progress in Combinatorics (ed: W. T. Tutte)*, Academic Press, London – New York, 1969, p. 91-102.

[3] M. Fontet. *Graphes 4-essentiels*, C. R. Acad. Se. Paris, **287**, serie A (1978), 289-290.

[4] W. Mader. Ecken Vom Gard n in minimalen n-fach zusammenhangenden Graphen. (German), *Arch.Math. (Basel)*, **23** (1972), 219-224.

[5] W. Mader. Generalization of critical connectivity of graphs. *Discrete Mathematics*, **72** (1988), 267-283.

[6] N. Martinov. A recursive characterization of the 4-connected graphs, *Discrete Mathematics* **84** (1990), 105-108.

[7] S. A. Obraztsova. Local structure of 5 and 6-connected graphs, *Journal of Mathematical Sciences*, **179** (2011), N. 5, 621-625.

[8] S. A. Obraztsova, A. Pastor. Local structure of 7- and 8-connected graphs. *Journal of Mathematical Sciences*, **179** (2011), N. 5, 626-633.

[9] S. A. Obraztsova. Local structure of 9 and 10-connected graphs *Journal of Mathematical Sciences*, **184** (2012), N. 5, 634-654.

Coset Closure of a Circulant S-ring and Schurity Problem*

Ilya Ponomarenko

St. Petersburg Department of Steklov Institute of Mathematics
St. Petersburg, Russia

A *Schur ring* or *S-ring* over a finite group G can be defined as a subring of the group ring $\mathbb{Z}G$ that is a free $\mathbb{Z}$-module spanned by a partition of G closed under taking inverse and containing $\{1_G\}$ as a class. It is well known that there is a Galois correspondence † between the permutation groups on G that contain the regular group G_{right}, and the S-rings over G:

$$\{\Gamma \le \mathrm{Sym}(G) :\ \Gamma \ge G_{right}\} \ \rightleftarrows\ \{\mathcal{A} \le \mathbb{Z}G :\ \mathcal{A} \text{ is an S-ring over } G\}. \quad (1)$$

More precisely, the "$\to$" mapping is given by taking the partition of G into the orbits of the stablizer of 1_G in Γ, whereas the "$\leftarrow$" mapping is given by taking the automorphism group of the colored Cayley graph corresponding to the partition of G associated with $\mathcal{A}$. The Galois closed objects are called 2-closed groups and *schurian* S-rings, respectively. The schurity problem consists in finding an inner characterization of schurian S-rings.

The theory of S-rings was initiated by I. Schur (1933) and later developed by H. Wielandt and his followers. The starting point for Schur was the Burnside theorem stating that any primitive permutation group containing a regular cyclic p-group of composite order, is 2-transitive. Using the S-ring method introduced by him, Schur generalized this theorem to an arbitrary finite cyclic group G. To some extent this explains the fact that "Schur had conjectured for a long time that every S-ring over G is

*based on joint work with Sergei Evdokimov

†We recall that a Galois correspondence between two posets consists of two mappings reversing the orders such that both superpositions are closure operators.

determined by a suitable permutation group" [8, p.54]. This statement had been known as the Schur-Klin conjecture up to 2001, when the first examples of circulant (i.e. over a cyclic group) S-rings were constructed in [1] by the authors. A recent result in [5] shows that schurian circulant S-rings are relatively rare. In this paper we provide a solution to the schurity problem for circulant S-rings.

The non-schurian examples of S-rings were constructed using the operation of *generalized wreath product* introduced in [1] (and independently in [6] under the name "wedge product"). This is not suprising due to the seminal Leung-Man theorem according to which any circulant S-ring can be constructed from S-rings of rank 2 and cyclotomic S-rings by means of two operations: tensor product and generalized wreath product [6]. Here under a *cyclotomic* S-ring $\mathcal{A}$ we mean the ring of all K-invariant elements of $\mathbb{Z}G$ where K is a subgroup of $\mathrm{Aut}(G)$:

$$\mathcal{A} = (\mathbb{Z}G)^K. \tag{2}$$

The Leung-Man theorrm reduces the schurity problem for circulant S-rings to finding a criterion for the schurity of the generalized wreath product. Such a criterion, based on a generalization of the Leung-Man theory (see [2]), was obtained in paper [4] where the generalized wreath product of permutation groups was introduced and studied. All these results form a background to prove the main results of the paper.

Let $\mathcal{A}$ be a circulant S-ring. Suppose that among the "bricks" in the Leung-Man decomposition of $\mathcal{A}$, there is a non-cyclotomic S-ring. Then this ring is of rank 2, its underlying group has composite order and it is Cayley isomorphic to the restriction of $\mathcal{A}$ to one of its sections. Moreover, as it was proved in [4] the S-ring $\mathcal{A}$ has a quite a rigid structure that enables us to control the schurity of $\mathcal{A}$. This provides a reduction of the schurity problem to the case when $\mathcal{A}$ has no rank 2 section of composite order. The S-rings satisfying the latter property are *quasidence* in sense of paper [5]. Thus without loss of generality we concentrate on the schurity problem for quasidence S-rings.

Our first step is to represent the schurian closure $\mathrm{Sch}(\mathcal{A})$ of a quasidense circulant S-ring $\mathcal{A}$ in a regular form (Theorem 1). The idea here is to replace the ring $\mathcal{A}$ by a simpler one keeping the structure of its Leung-Man decomposition. The simplification is achieved by changing

each "brick" for a group ring. This leads to the class of *coset* S-rings, i.e. ones for which any class of the corresponding partition of the group G is a coset of a subgroup in G. It appears that this class is closed under restriction to a section, tensor and generalized wreath products, and consists of schurian quasidense S-rings. The regular form of $\mathrm{Sch}(\mathcal{A})$we want to come, will be defined by means of the following concept.

Definition 1 *The coset closure of a quasidense circulant S-ring $\mathcal{A}$ is the intersection $\mathcal{A}_0$ of all coset S-rings over G that contain $\mathcal{A}$.*

The coset closure of any quasidense circulant S-ring is a coset S-ring. Now, to clarify how to represent the schuran closure of $\mathcal{A}$ via its coset closure, suppose that the group G is of prime order. In this case it is well known that the S-ring $\mathcal{A}$ is of the form (2), and, moreover, $\mathcal{A}_0 = \mathbb{Z}G$. In particular, $\mathcal{A}$ is schurian and any automorphism of G induces a similarity of $\mathcal{A}_0$.[‡] Furthermore, if the automorphism belongs to the group K, the similarity is identical on $\mathcal{A}$. Thus

$$\mathcal{A} = (\mathbb{Z}G)^K = (\mathcal{A}_0)^{\Phi_0}$$

where $\Phi_0 = \Phi_0(\mathcal{A})$ is the group of all similarities of $\mathcal{A}_0$ that are identical on $\mathcal{A}$. It appears that this idea works for any quasidense S-ring $\mathcal{A}$.

Theorem 1 *Let $\mathcal{A}$ be a quasidense circulant S-ring. Then*

$$\mathrm{Sch}(\mathcal{A}) = (\mathcal{A}_0)^{\Phi_0}.$$

In particular, $\mathcal{A}$ is schurian if and only if $\mathcal{A} = (\mathcal{A}_0)^{\Phi_0}$.

Theorem 1 gives a necessary and sufficient condition for an S-ring to be schurian. This condition being a satisfactory from the theoretical point of view, is hardly an inner characterization. To obtain the latter, we prove Theorem 2 below. Let us discuss briefly the idea behind it.

One of the key properties of coset S-rings that is used in the proof of Theorem 1, is that every similarity of any such ring is induced by isomorphism. This fact also shows that in the schurian case the set of

[‡]Under a similarity of an S-ring $\mathcal{A}$ we mean a ring isomorphism of it that respects the partition of G corresponding to $\mathcal{A}$.

all isomorphisms of $\mathcal{A}_0$ that induce similarities belonging to Φ_0, forms a permutation group the associated S-ring of which coincides with $\mathcal{A}$. In general, this is not true. A rough reason for this can be explained as follows. Set

$$\mathfrak{S}_0 = \{S \in \mathfrak{S}(\mathcal{A}_0) : \ (\mathcal{A}_0)_S = \mathbb{Z}S\} \tag{3}$$

where $\mathfrak{S}(\mathcal{A}_0)$ is the set of all $\mathcal{A}_0$-sections and $(\mathcal{A}_0)_S$ is the restriction of $\mathcal{A}_0$ to S. Then in the schurian case every S-ring $\mathcal{A}_S$ with $S \in \mathfrak{S}_0$, must be cyclotomic, whereas in general this condition does not necessarily hold. However, if even all the S-rings $\mathcal{A}_S$ are cyclotomic, one still might find a section S for which $\mathcal{A}_S \neq \mathrm{Sch}(\mathcal{A})_S$. These two reasons are controlled respectively by conditions (1) and (2) of Theorem 2.

It should be mentioned that the proof of the fact that the circulant S-rings constructed in [1] are non-schurian, was based on studying the relationship between their cyclotomic sections. More careful analysis can be found in [3] where the isomorphism problem for circulant graphs was solved. In that paper the authors introduce and study the notion of projective equivalence on the sections of a circulant S-ring (this notion is similar to one used in the lattice theory). It appears that the class $\mathfrak{S}_0$ defined in (3) is closed under the projective equivalence and taking subsections. Moreover,

$$\mathfrak{S}(\mathcal{A}_0) = \mathfrak{S}(\mathcal{A}).$$

To formulate Theorem 2 we need additional notation. For $S \in \mathfrak{S}(\mathcal{A})$ denote by $\mathrm{Aut}_{\mathcal{A}}(S)$ the subgroup of $\mathrm{Aut}(S)$ that consists of all Cayley automorphisms of the S-ring $\mathcal{A}_S$. A family

$$\Sigma = \{\sigma_S\}_{S \in \mathfrak{S}_0}$$

is called a *multiplier* of $\mathcal{A}$ if for any sections $S, T \in \mathfrak{S}_0$ such that T is projectively equivalent to a subsection of S the automorphisms $\sigma_T \in \mathrm{Aut}(T)$ and $\sigma_S \in \mathrm{Aut}(S)$ are induced by raising to the same power[§]. The set of all multipliers of $\mathcal{A}$ forms a subgroup of the direct product $\prod_{S \in \mathfrak{S}_0} \mathrm{Aut}_{\mathcal{A}}(S)$ that is denoted by $\mathrm{Mult}(\mathcal{A})$.

Theorem 2 *A quasidense circulant S-ring $\mathcal{A}$ is schurian if and only if the following two conditions are satisfied for all $S \in \mathfrak{S}_0$:*

[§]We recall that any automorphism of a finite cyclic group is induced by raising to a power coprime to the order of this group.

1. *1 the S-ring $\mathcal{A}_S$ is cyclotomic,*
2. *2 the restriction homomorphism from* $\mathrm{Mult}(\mathcal{A})$ *to* $\mathrm{Aut}_{\mathcal{A}}(S)$ *is surjective.*

We prove that the class $\mathfrak{S}_0$ consists of all $\mathcal{A}$-sections S such that each Sylow subgroup of S (treated as a section of G) is projectively equivalent to a subsection of a principal $\mathcal{A}$-section. Thus in contrast to Theorem 1, Theorem 2 gives a necessary and sufficient condition for an S-ring $\mathcal{A}$ to be schurian in terms of $\mathcal{A}$ itself rather than of its coset closure $\mathcal{A}_0$. It should be remarked that in general the class $\mathfrak{S}_0$ may contain non-cyclotomic sections. However, we do not know whether condition (1) in Theorem 2 is implied by condition (2).

We would like to reformulate Theorem 2 in the number theoretical language. In what follows we assume that condition (1) of that theorem is satisfied. To make condition (2) more clear let us fix a section $S_0 \in \mathfrak{S}_0$ and an integer b coprime to $n_{S_0} = |S_0|$ for which the mapping $s \mapsto s^b$, $s \in S_0$, belongs to $\mathrm{Aut}_{\mathcal{A}}(S_0)$. Let us consider the following system of linear equations in integer variables x_S, $S \in \mathfrak{S}_0$:

$$\begin{cases} x_S \equiv x_T \pmod{n_T}, \\ x_{S_0} \equiv b \pmod{n_{S_0}} \end{cases} \tag{4}$$

where S and T run over $\mathfrak{S}_0$ and the section T is projectively equivalent to a subsection of S. We are interested only in the solutions of this system that satisfy the additional condition

$$(x_S, n_S) = 1 \quad \text{for all} \quad S \in \mathfrak{S}_0. \tag{5}$$

Every such solution produces the family $\Sigma = \{\sigma_S\}$ where σ_S is the automorphism of the group S taking s to s^{x_S}. Moreover, the equations in the first line of (4) guarantee that if a section T is projectively equivalent to a subsection of S, then the automorphisms $\sigma_T \in \mathrm{Aut}(T)$ and $\sigma_S \in \mathrm{Aut}(S)$ are induced by raising to the same power. Therefore,

$$\Sigma \in \mathrm{Mult}(\mathcal{A}).$$

Conversely, it is easily seen that given $S_0 \in \mathfrak{S}$ every multiplier of $\mathcal{A}$ produces a solution of system (4) for the corresponding b. Finally, the

consistency of this system for all S_0 and all possible b is equivalent to the surjectivity of the restriction homomorphism from $\mathrm{Mult}(\mathcal{A})$ to $\mathrm{Aut}_{\mathcal{A}}(S_0)$ for all S_0. Thus we come to the following corollary of Theorem 2.

Corollary 1 *Let $\mathcal{A}$ be a quasidense circulant S-ring such that for any section $S \in \mathfrak{S}_0$, the S-ring $\mathcal{A}_S$ is cyclotomic. Then $\mathcal{A}$ is schurian if and only if system* (4) *has a solution satisfying* (5) *for all possible S_0 and b.*

Corollary 1 reduces the schurity problem for circulant S-rings to solving modular linear system (4) under restriction (5). One possible way to solve this system is to represent the group $\prod_S \mathrm{Aut}_{\mathcal{A}}(S)$ as a permutation group on the disjoint union of the sections S. Then every equation in the first line of (4) defines a subgroup of that group the index of which is at most n^2. Therefore the set of solutions can be found by a standard permutation group technique, see [7, p. 144] for details.

References

[1] S. Evdokimov, I. Ponomarenko, *On a family of Schur rings over a finite cyclic group*, Algebra and Analysis, **13** (2001), 3, 139–154.

[2] S. Evdokimov, I. Ponomarenko, *Characterization of cyclotomic schemes and normal Schur rings over a cyclic group*, Algebra and Analysis, **14** (2002), 2, 11–55.

[3] S. Evdokimov, I. Ponomarenko, *Recognizing and isomorphism testing circulant graphs in polynomial time*, Algebra and Analysis, **15** (2003), 6, 1–34).

[4] S. Evdokimov, I. Ponomarenko, *Schurity of S-rings over a cyclic group and generalized wreath product of permutation groups*, Algebra and Analysis, **24** (2012), 3, 84–127.

[5] S. Evdokimov, I. Kovács, I. Ponomarenko, *Characterization of cyclic Schur groups*, Algebra and Analysis, **25** (2013), 5, Algebra and Analysis, 61–85.

[6] K. H. Leung, S. H. Man, *On Schur Rings over Cyclic Groups, II*, J. Algebra, **183** (1996), 273–285.

[7] E. M. Luks, *Permutation groups and polynomial-time computation*, American Mathematical Society. DIMACS, Ser. Discrete Math. Theor. Comput. Sci. 11, Providence, RI: American Mathematical Society, **1993**, 139–175.

[8] H. Wielandt, *Permutation groups through invariant relations and invariant functions*, Lect. Notes Dept. Math. Ohio St. Univ., Columbus, 1969.

Asymmetry in Discrete-Time Bioresource Management Problem*

Anna N. Rettieva

Institute of Applied Mathematical Research,
Karelian Research Centre RAS, Petrozavodsk, Russia

Discrete-time game-theoretic models related to a bioresource management problem (fish catching) are investigated. The players are countries or fishing firms that harvest the fish stock. Players differ in their discount factors and planning horizons. Two variants are considered: fixed and random harvesting times.

The main goal here is to construct the value function for the cooperative solution and to distribute the joint payoff among the players. We propose to use Nash bargaining solution in order to determine cooperative behavior.

1. Fixed harvesting times

Let two players (countries or fishing firms) exploit the fish stock. The dynamics of the fishery is described by the equation

$$x_{t+1} = (\varepsilon x_t - u_{1t} - u_{2t})^{\alpha}\,, \;\; x_0 = x\,, \tag{1}$$

where $x_t \geq 0$ – the size of population at a time t, $\varepsilon \in (0,1)$ – natural survival rate, $\alpha \in (0,1)$ – natural birth rate, $u_{it} \geq 0$ – the catch of player i, $i = 1,2$.

The first player extracts the stock n_1 time moments, and the second – n_2. Let $n_1 < n_2$. So, we have the situation when on time interval $[0, n_1]$ players cooperate and we need to determine their strategies. After n_1 till n_2 the second player acts individually. So the players' profits have forms

$$J_1 = \sum_{t=0}^{n_1} \delta_1^t \ln(u_{1t}^c)\,, \;\; J_2 = \sum_{t=0}^{n_1} \delta_2^t \ln(u_{2t}^c) + \sum_{t=n_1+1}^{n_2} \delta_2^t \ln(u_{2t}^a)\,, \tag{2}$$

*The research was supported by the Russian Fund for Basic Research, grants 13-01-91158-GFEN_a and 13-01-00033a.

where u_i^c, $i = 1, 2$ are the cooperative strategies, u_2^a is the second's player strategy when she extracts the stock alone.

The main question arising here is how to construct the value function for cooperative solution in the case when players have different discount factors and planning horizons. We propose to use recursive Nash bargaining solution to determine cooperative behavior.

We construct cooperative strategies and the joint payoff maximizing the Nash product for the whole game, so we solve the following problem

$$(V_1^c(x,\delta_1)[0,n_1] - V_1^N(x,\delta_1)[0,n_1]) \cdot (V_2^c(x,\delta_2)[0,n_1] + \\ + V_2^{ac}(x^{cn_1},\delta_2)[n_1,n_2] - V_2^N(x,\delta_2)[0,n_1] - V_2^{aN}(x^{Nn_1},\delta_2)[n_1,n_2]) \to \max, \quad (3)$$

where $V_i^N(x,\delta_i)[0,n_1]$ are the non-cooperative gains, $V_2^{ac}(x^{cn_1},\delta_2)[n_1,n_2]$ – the second player's gain when acting individually after n_1 periods of cooperation,
$V_2^{aN}(x^{Nn_1},\delta_2)[n_1,n_2]$ – the second player's gain when acting individually after n_1 periods of noncooperation.

2. Random harvesting times

Here we consider an extension of the previous model where the stochastic nature of real processes is captured. Let n_1 be a random variable with range $\{1,\dots,n\}$ and corresponding probabilities $\{\theta_1,\dots,\theta_n\}$. n_2 is random variable with the same range and probabilities $\{\omega_1,\dots,\omega_n\}$.

First, we determine the Nash equilibrium as we use it as a status-quo point for the Nash bargaining solution.

Second, we construct cooperative strategies and the payoff maximizing the Nash product for the whole game, so we solve the next problem

$$(V_1^c(1,x) - V_1^N(1,x))(V_2^c(1,x) - V_2^N(1,x)) = \\ = (\sum_{n_1=1}^{n} \theta_{n_1} \Big[\sum_{n_2=n_1}^{n} \omega_{n_2} \sum_{t=1}^{n_1} \delta_1^t \ln(u_{1t}^c) +$$

$$+\sum_{n_2=1}^{n_1-1}\omega_{n_2}\Big(\sum_{t=1}^{n_2}\delta_1^t\ln(u_{1t}^c)+\sum_{t=n_2+1}^{n_1}\delta_1^t\ln(u_{1t}^a)\Big)\Big]-V_1^N(1,x))\cdot$$

$$\cdot\Big(\sum_{n_2=1}^{n}\omega_{n_2}\Big[\sum_{n_1=n_2}^{n}\theta_{n_1}\sum_{t=1}^{n_2}\delta_2^t\ln(u_{2t}^c)+$$

$$+\sum_{n_1=1}^{n_2-1}\theta_{n_1}\Big(\sum_{t=1}^{n_1}\delta_2^t\ln(u_{2t}^c)+\sum_{t=n_1+1}^{n_2}\delta_2^t\ln(u_{2t}^a)\Big)\Big]-V_2^N(1,x)\Big)\to\max\,, \quad (4)$$

where $V_i^N(1,x)=A_i^N\ln x+B_i^N$, $i=1,2$ are the non-cooperative gains.

The results of numerical modelling, in the stochastic case using Monte-Carlo method, are presented.

References

[1] M. Breton, M.Y. Keoula, A great fish war model with asymmetric players, *Cahiers du GERAD* G-2010-73, December 2010.

[2] D. Levhari , L.J. Mirman, The great fish war: an example using a dynamic Cournot-Nash solution, *The Bell Journal of Economic*. V. 11. N 1 (1980), 322–334.

[3] V.V. Mazalov, A.N. Rettieva, Fish wars and cooperation maintenance,*Ecological Modelling*. V. 221 (2010), 1545–1553.

[4] A.N. Rettieva, Discrete-time bioresource management problem with asymmetric players, *Matematicheskaya Teoriya Igr i Prilozheniya*. V. 5. N. 3 (2013), 72–87. (in Russian).

Equivalence Relations Defined by Numbers of Occurrences of Factors*

Aleksi Saarela

Department of Mathematics and Statistics
University of Turku, Turku, Finland

We study the question of what can be said about a word based on the numbers of occurrences of certain factors in it. As a simple example, suppose that we do not know the word $u \in \{0,1\}^*$, but we know its length $|u|$ and the number of 0's $|u|_0$. Then we can of course deduce the number of 1's: $|u|_1 = |u| - |u|_0$. As another example, suppose that we do not know the word $u \in \{0,1\}^+$, but we know its length $|u|$, first letter $\mathrm{pref}_1(u)$, last letter $\mathrm{suff}_1(u)$, and the number of 01's $|u|_{01}$. Then we can deduce the number of 10's: $|u|_{10} = |u|_{01} + \mathrm{pref}_1(u) - \mathrm{suff}_1(u)$.

To formally present some questions and results, we define an equivalence relation: For an alphabet Σ, positive integer k, and set $S \subseteq \Sigma^{\leq k}$, words $u, v \in \Sigma^*$ are called (k,S)-*equivalent* if $|u| = |v|$, $|u|_s = |v|_s$ for all $s \in S$, $\mathrm{pref}_{k-1}(u) = \mathrm{pref}_{k-1}(v)$, and $\mathrm{suff}_{k-1}(u) = \mathrm{suff}_{k-1}(v)$ (if $|u| < k-1$, we define $\mathrm{pref}_{k-1}(u) = \mathrm{suff}_{k-1}(u) = u$).

On one extreme, there is $(1,\varnothing)$-equivalence, which takes into account only the length of a word. On the other extreme, there is $(k,\Sigma^{\leq k})$-equivalence, which is known as *k-abelian equivalence*. Many sets S can lead to the same equivalence. For example, it is known that Σ^k or $\Sigma^{\leq k} \setminus a\Sigma^* \setminus \Sigma^* a$ (where a is an arbitrary letter) could be used instead of $\Sigma^{\leq k}$ in the definition of k-abelian equivalence. One of our motivations is finding a characterization for the sets S such that (k,S)-equivalence is the same as k-abelian equivalence.

The first specific question we study is the following: Given k and S, for which $t \in \Sigma^{\leq k}$ can we deduce $|u|_t$ based on the (k,S)-equivalence class of u? The set of all such t does not depend on k and it is denoted by $\overline{S}$.

*Supported by the Academy of Finland under grant 257857.

The second question is the following: For which S_1, S_2 is (k, S_1)-equivalence the same as (k, S_2)-equivalence, or in other words, $\overline{S}_1 = \overline{S}_2$. These two questions are of course closely related and we can answer both of them by using linear algebra.

The third question is the number of (k, S)-equivalence classes of words of length n. It is $\Theta(n^m)$, where m is the size of the smallest set R such that $\overline{R} = \overline{S}$.

It is known that every infinite aperiodic word w has factors of length n in at least $\min(2k, n+1)$ k-abelian equivalence classes. Moreover, w is Sturmian if and only if it has factors of length n in exactly $\min(2k, n+1)$ k-abelian equivalence classes for all n. Analyzing the proof of this result reveals that it can be generalized for (k, S)-equivalence whenever $\Sigma \subseteq \overline{S}$. If $\Sigma \not\subseteq \overline{S}$, then such a generalization does not necessarily exist.

On the Multiplicative Complexity of Some Boolean Functions*

Svetlana N. Selezneva

Moscow State University, Moscow, Russia

In this paper, we study the multiplicative complexity of Boolean functions. The multiplicative complexity $\mu(f)$ of a Boolean function f is the minimal number of &-gates (binary multiplications) in circuits in the basis $\{x\&y, x \oplus y, 1\}$ that compute the function f. Some results of the multiplicative complexity of Boolean functions are in [1–6].

Introduce some definitions. A Boolean function f of n variables is a mapping $f : B^n \to B$, where $B = \{0, 1\}$, $n = 0, 1, \ldots$. Each Boolean function $f(x_1, \ldots, x_n)$ can uniquely be represented by a Zhegalkin polynomial, i.e., by an EXOR sum of monomials (of positive products of variables). The degree $\deg(f)$ of a Boolean function $f(x_1, \ldots, x_n)$ is the maximal number of variables in monomials of the Zhegalkin polynomial for the function f. In [1], it was proved that $\mu(f) \geq \deg(f) - 1$ holds for an arbitrary Boolean function f. This is the best known lower bound of the multiplicative complextity for explicitly defined Boolean functions. It is clear that, for example, this bound is exact for the conjunction of n variables, i.e., for the function $x_1 x_2 \ldots x_n$.

A Boolean function g is called affine iff $\deg(g) \leq 1$. A Boolean function f is called multi-affine iff f can be represented in the form $\prod_{i=1}^{m} g_i$ where $g_1, \ldots, g_m$ are affine functions. In [1], it was shown that $\mu(f) = \deg(f) - 1$ holds for an arbitrary multi-affine Boolean function f. In this paper, we consider functions that can be represented as an EXOR sums of two multi-affine functions, and we prove the following theorem.

*This work is supported by RFBR, grant 13-01-00684-a.

Theorem 1 *If $n \geq 2$, and a Boolean functions $f(x_1, \ldots, x_n)$ can be represented in the form $f_1(x_1, \ldots, x_n) \oplus f_2(x_1, \ldots, x_n)$, where f_1, f_2 are multiaffine functions, then*

1) $\mu(f) = n - 2$ in the case of $\deg(f_1) = \deg(f_2) = n$;
2) $\mu(f) = n - 1$ in the case of $\deg(f_1) = n$, $\deg(f_2) < n$;
3) $\mu(f) \leq n - 1$ in the case of $\deg(f_1) < n$, $\deg(f_2) < n$.

A Boolean function $f(x_1, \ldots, x_n)$ is called quadratic iff $\deg(f) = 2$. In [1, 2], it was shown that $\mu(f) \leq \lfloor n/2 \rfloor$ holds for an arbitrary quadratic Boolean function $f(x_1, \ldots, x_n)$, and, moreover, quadratic functions $f(x_1, \ldots, x_n)$ with $\mu(f) = \lfloor n/2 \rfloor$ were described. In this paper, we extend this result, and we prove the following theorem.

Theorem 2 *If $n \geq 3$, and a Boolean function $f(x_1, \ldots, x_n)$ can be represented in the form $x_1 \ldots x_n \oplus q(x_1, \ldots, x_n)$, where q is a quadratic function, then $\mu(f) = n - 1$.*

The multiplicative complexity of Boolean functions in the worst case was studied. In [3], it was obtained that for an arbitrary Boolean function $f(x_1, \ldots, x_n)$, $\mu(f) \leq 2 \cdot 2^{n/2} - O(n)$ holds, if n is even, and $\mu(f) \leq (3/\sqrt{2}) \cdot 2^{n/2} - O(n)$ holds, if n is odd. In this paper, we improve these upper bounds, and we prove the following theorem.

Theorem 3 *For an arbitrary Boolean function $f(x_1, \ldots, x_n)$, $\mu(f) \leq (3/2) \cdot 2^{n/2} + o(2^{n/2})$ holds, if n is even, and $\mu(f) \leq \sqrt{2} \cdot 2^{n/2} + o(2^{n/2})$ holds, if n is odd.*

References

[1] C.P. Schnorr, The multiplicative complexity of Boolean functions, *Proc. 1st Internat. Joint Conf of ISSAC '88 and AAECC-6*, Rome (1988). Lecture Notes in Computer Science. **357** (1989), 45–58.

[2] R. Mirwald, C.P. Schnorr, The multiplicative complexity of quadratic boolean forms, *Theoretical Computer Science*. **102** (1992), 307–328.

[3] J. Boyar, R. Peralta, D. Pochuev, On the Multiplicative Complexity of Boolean Functions over the Basis $\{\wedge, \oplus, 1\}$, *Theor. Comp. Sci.* **235** (2000), 43–57.

[4] A. Kojevnikov, A.S. Kulikov Circuit Complexity and Multiplicative Complexity of Boolean Functions. *Lecture Notes in Computer Science*. **6158** (2010), 239–245.

[5] T.I. Krasnova On the conjunction complexity of circuits in the Zhegalkin basis for one sequence of Boolean functions, *Proc. of XI International Workshop "Discrete Mathematics and its Applications"*. Moscow. 2012. P. 138–141 (in Russian).

[6] I.S. Sergeev, A Relation between Additive and Multiplicative Complexity of Boolean Functions, arXiv:1303.4177. 2013.

On the Number of Distinct Subpalindromes in Words

Mikhail Rubinchik, Arseny M. Shur

Ural Federal University
Ekaterinburg, Russia

Palindromes are among the most important and actively studied repetitions in words. Recall that a word $w = a_1 \cdots a_n$ is a palindrome if $a_1 \cdots a_n = a_n \cdots a_1$. In particular, all letters are palindromes; the empty word is also considered as a palindrome, but below we do not count it. A group of combinatorial problems concerns the possible number of distinct palindromic factors, or subpalindromes, in a word. We call this number *palindromic richness*.

Clearly, for the words containing k different letters the lower bound for their palindromic richness is k. If $k > 2$, then this bound is sharp, since the infinite word $(a_1 \cdots a_k)^\omega$, where $a_1, \ldots, a_k$ are different letters, has no subpalindromes except letters. For $k = 2$ the situation is less obvious: the minimum richness of an infinite word is 8, and the minimum richness of an *aperiodic* infinite word is 10 [2]. On the other hand, the maximum richness of an n-letter word over any alphabet is n, as was first observed in [1]. Such "rich" words are objects of intensive study (see, e.g., [3]).

However, the extremums mentioned above give no clue about the generic case.

Observation *Any number between 8 and N in the binary case, and between k and N in the k-ary case with $k > 2$ is the palindromic richness of a word of length N.*

So, the following question is quite natural:

what is the expected palindromic richness of a random word of length N?

We studied this question using both theory and numerical experiments. Our main theoretic result is the following

Theorem 1 *For any fixed alphabet Σ, the expected palindromic richness of a random word of length N over Σ is $\Theta(\sqrt{N})$.*

Note that the expected total number of nontrivial subpalindromes in a random word is $\Theta(N)$, but the constant drops quickly as the alphabet grows. In a contrast with that, the constant in the Θ-expression from Theorem 1 even grows with $k = |\Sigma|$. More precisely, we proved that this constant $C(k)$

- tends to an absolute constant as $k \to \infty$ if N is close to an even power of k;
- grows as $\sqrt{k}$ if N is close to an odd power of k;
- stays in between the above extremal values in the remaining cases.

We also performed some computational experiments based on the linear-time algorithm for counting distinct subpalindromes in a word [4]. By averaging the data obtained for groups of random words we derive the following estimations for $C(k)$:

k	$C(k)$ for even powers	$C(k)$ for odd powers
2	6.129 for $N = 2^{16}$	6.164 for $N = 2^{17}$
3	4.393 for $N = 3^{12}$	4.408 for $N = 3^{13}$
10	3.023 for $N = 10^6$	3.388 for $N = 10^7$
50	2.702 for $N = 50^4$	5.038 for $N = 50^3$

The figures in the central column decrease but seem to have a limit about 2.5. The figures in the rightmost column demonstrate an initial decrement but then grow back to follow the theoretical bound of $\Theta(\sqrt{k})$. All these figures are nicely predicted by a very simple probabilistic model in which all events of the form "to contain a fixed subpalindrome u of length p" are assumed independent and equiprobable. This is worth noting because these events in fact are dependent and have different probabilities.

References

[1] X. Droubay, J. Justin, and G. Pirillo. Episturmian words and some constructions of de Luca and Rauzy. *Theoret. Comput. Sci.*, 255 (2001), 539–553.

[2] G. Fici and L.Q. Zamboni. On the least number of palindromes contained in an infinite word. *Theoret. Comput. Sci.*, 481 (2013), 1–8.

[3] A. Glen, J. Justin, S. Widmer, and L.Q. Zamboni. Palindromic richness. *European J. Combinatorics*, 30(2) (2009), 510–531.

[4] D. Kosolobov, M. Rubinchik, and A. M. Shur. Finding distinct subpalindromes online. In *Proc. Prague Stringology Conference. PSC 2013*, pages 63–69. Czech Technical University in Prague, 2013.

Short communications

Some Models of Representation of Two Parallel FIFO-queues and Their Optimal Control*

Eugene A. Barkovsky

Institute of Applied Mathematical Research,
Karelian Research Centre RAS, Petrozavodsk, Russia

In many applications, such as the development of various network devices and embedded operating systems it is required to work with multiple FIFO-queues, located in the shared memory space. Mechanism of paged virtual memory is not used here, and the entire operation occurs in multiple memory pools. The number of queues in such devices can reach several hundreds and thousands, and in the future, according to experts, may reach several million [1]. To represent FIFO-queues different software or hardware solutions are used [2, 3, 4].

Suppose that in the memory size of m we work with two parallel FIFO-queues. Operations with queues are performed by the following scheme: on the odd step occurs operation of insertion in one of the queues, on the even step — deletion from any of the queues, where some probabilities of operations performed with queues are known. Assume that p_1 and p_2 are probabilities of insertion in the first and second queue respectively; p_{12} — probability of simultaneous insertion in both queues. q_1 and q_2 — probabilities of deletion from the first and second queue respectively; q_{12} — probability of simultaneous deletion from both queues. Since Markov

*This research work was supported by Russian Foundation for Basic Research, grant 12-01-00253-a and by program of the strategic development of PetrSU as a part of the complex of measures for the development of research activities.

chain, built on the basis of such formulation of the problem, will not be regular and uniform, two consecutive steps are combined in one and also we introduce the following probabilities of operations that do not change the length of the queue (for example, reading): r_1 — on the odd step and r_2 — on the even step, where $r_1 \neq 0$, $r_2 \neq 0$. Accordingly, $p_1 + p_2 + p_{12} + r_1 = 1$, $q_1 + q_2 + q_{12} + r_2 = 1$.

Here we propose mathematical and simulation models of the process of working with two parallel FIFO-queues and solve the problem of optimal partitioning of shared memory in following cases:

I. Queues are sequential and cyclic.

II. Queues move in a circle one after another.

In both cases the criterion of optimality is the minimum average portion of lost elements on an infinite time. The forms of the matrixes of transition probabilities, corresponding to Markov chain, were established and algorithms of their generation were created.

We work with problems of nonlinear discrete programming where the criterion of optimality is defined algorithmically. To solve said problems we use apparatus of controlled random walks, Markov chains [5] and system Intel Math Kernel Library PARDISO.

References

[1] A. Nikologiannis, M. Katevenis, Multi Queue Management for Advanced QoS in High-Speed Communication Systems Computer Architecture and VLSI Systems Lab, Institute of Computer Science (ICS) Head, `http://archvlsi.ics.forth.gr/muqpro/queueMgt.html`

[2] R. Sedgwick, *Algorithms in C++.* Third Edition. Parts 1-4. Addison Wesley Longman, 1999.

[3] V. Bollapragada, C. Murphy, R. White, *Inside Cisco IOS.* Software Arcitecture. Cisco Press, 2000.

[4] D.E. Knuth, *The Art of Computer Programming.* Vol. 1. Addison-Wesley, Reading, MA, 2001.

[5] J.G. Kemeny, J.L. Snell, *Finite Markov Chains.* Van Nostrand, Princeton, New Jersey, 1960.

On Property B of Hypergraphs

Danila D. Cherkashin

Saint-Petersburg State University, Mathematics and Mechanics Faculty
Saint-Petersburg, Russia

I am going to speak about a classical quantity $m(n)$ introduced by Erdős and Hajnal in 1961 (see [1]).

A hypergraph $H = (V, E)$ is said to have *property* B, if there is a 2-coloring of V with no monochromatic edges. Denote by $m(n)$ the minimum number of edges in a hypergraph that does not have property B.

The best known bounds for $m(n)$ are as follows:

$$c\sqrt{\frac{n}{\ln n}}2^n < m(n) < c'n^2 2^n.$$

The lower bound is due to Radhakrishnan and Srinivasan (see [2]), and the upper bound was given by Erdős.

I want to present a new simple proof of the lower bound (based on ideas by A. Pluhár from [3]) and a new lower bound for a quantity $m(n, r)$ that generalizes $m(n)$ onto the case of r colors.

This is my joint work with J. Kozik.

References

[1] P. Erdős, A. Hajnal, On a property of families of sets, *Acta Mathematica of the Academy of Sciences*, **12** (1-2) (1961), 87–123.

[2] J.Radhakrishnan, A.Srinivasan, Improved bounds and algorithms for hypergraph two-coloring, *Random Structures and Algorithms*, **16**(1) (2000), 4–32.

[3] A. Pluhár, Greedy colorings for uniform hypergraphs, *Random Structures and Algorithms*, **35**(2) (2009), 216–221.

Paged Representation of Stacks in Single-Level Memory*

Andrew V. Drac

Institute of Applied Mathematical Research
Karelian Research Centre RAS
Petrozavodsk State University
Petrozavodsk, Russia

Consider a continuous block of m cells, which we use to implement n LIFO-stacks. Assume that the time is discrete and one of the following operations can happen during each time step:

- insertion of the element into i-th stack with the probability p_i,
- deletion of the element from i-th stack with the probability q_i $(1 \leq i \leq n)$,
- access the element with the probability r (stacks don't change their lengths).

The time to absorbtion and the final sizes of stacks are random variables whose distributions depend on m and the probabilities of insertion and deletion. There are several methods of representation of stacks in single-level memory. In [2],[3] we considered consecutive and linked representations of stacks.

In the case of paged representation all the memory is split into parts with equal sizes. Let k is the size of page. Then $\lfloor m/k \rfloor$ is the total number of pages. Each page contains a link to previous page and may contain $k-1$ elements of one of stacks.

In the case of insertion of element if memory in page is exausted it will be

*This research work was supported by the Russian Foundation for Basic Research, grant 12-01-00253-a and program of the strategic development of PetrSU as a part of the complex of measures for the development of research activities.

put into one of empty pages. If there is no one then the overflow occurs. In the case of deletion of element if it is the only element in page then the page becomes empty. At the begining of work all stacks are empty and there is no shutdown in the case of deletion of element from empty stack. The problem is to find the average time T of working before memory overflow.
As the mathematical model we used the apparatus of absorbing Markov chains. In this paper we calculated average time of working with stacks in the case of paged representation and compared it with consequtive and linked representations.

References

[1] D. E. Knuth. *The art of computer programming.* Vol. 1, Addison-Wesley, Reading, MA, 2001.

[2] A. V. Sokolov, A. V. Drac. Allocation of n Stacks and/or Queues in Single-Level Memory. *Stochastic Optimization in Informatics*, Vol. 4 (2008), 72-89. (in Russian)

[3] A. V. Sokolov, A. V. Drac. The linked list representation of n LIFO-stacks and/or FIFO-queues in the single-level memory. *Information Processing Letters*, Vol. 13, Iss. 19-21 (2013), 832-835.

[4] J. Riordan. *Introduction to Combinatorial Analysis*, Dover Publications, 1958.

[5] J. G. Kemeny, J. L. Snell *Finite Markov Chains*, Van Nostrand, Princeton, New Jersey, 1960.

Desctiptive Complexity of Additive Shift of Regular Language

Denis D. Dublennykh

Insitiute of Mathematics and Computer Science
Ural Federal University
Ekaterinburg, Russia

Abstract

In this article we show that if we consider binary strings as binary representations of integer numbers, then for any deterministic finite automaton A and for any integer k there is such deterministic automaton B that language of automaton B is language of all integers from language of automaton A decreased by k, and also we present upper bound on the size of automaton B.

Let Λ be a set of all deterministic finite automatons with alphabet $\Sigma = \{0, 1\}$ and $\mathbb{N}$ be a set of all positive integers. Let $f : \Lambda \times \mathbb{N} \to \{0, 1\}$ be a function that takes an automaton $A = \langle \Sigma, Q, q_0, \delta, T \rangle$ and a positive integer x as its input and returns 0 and 1 as its output depending on whether automaton A accepts standard binary representation of number x or not. More strictly: let $x = a_l \cdot 2^l + a_{l-1} \cdot 2^{l-1} + \cdots + a_0$, where $a_i \in \{0, 1\}$, $a_l = 1$, $q_t = \delta(q_0, a_l a_{l-1} \dots a_0)$. Then $f(A, x) = 1$ if $q_t \in T$, and $f(A, x) = 0$ otherwise.

Theorem 1 *For any automaton $A \in \Lambda$ with n states and for any integer $k \in \mathbb{N}$ there is such automaton $B \in \Lambda$ with no more than $\lceil \frac{k}{2} \rceil n^2 + 2kn + 2k$ states that for any $x \in \mathbb{N}$ $f(B, x) = f(A, x + k)$.*

Proof. The proof will be divided into three parts. First of them will contain explicit construction that produce required automaton for $k = 1$. Second part will show how to generalize this construction for any k, but

with resulting automatons having $2kn^2 + 2k$ states. And the last part will present more detailed analysis of produced automatons and a proof of the fact that any of them have equivalent one with required number of states.

Later on we will be using the following utility functions:

- $1_Y : X \to \{0, 1\}$ — indicator function of subset Y of set X ($1_Y(x) = 1$ if $x \in Y$, $1_Y(x) = 0$ otherwise).
- $g : \Lambda \to \mathbb{S}$, where $\mathbb{S}$ is a union of all states of all automatons in Λ. g is a function that takes an automaton $A \in \Lambda$ and an integer $x \in \mathbb{N}$ and returns the final state of automaton A after processing binary representation of number x as its output.
- $bin_l : \{0, \ldots, 2^l - 1\} \to \{0, 1\}^l$ — function takes an integer as its intput and returns its binary representation padded with zeros to the length of l. That is if $x = 2^{l-1}a_{l-1} + 2^{l-2}a_{l-2} + \cdots + a_0$, where $a_i \in 0, 1$, then $bin_l(x) = a_{l-1}a_{l-2} \ldots a_0$.
- $\mathrm{mod}_m : \mathbb{N} \to \{0, \ldots, m - 1\}$ — function that returns remainder of interer division by m, i.e. $\mathrm{mod}_m x = x - m \left\lfloor \frac{x}{m} \right\rfloor$

In particular, for any automaton $A = \langle \Sigma, Q, q_0, \delta, T \rangle$ $f(A, x) = 1_T(g(A, x))$.

Part 1. For the beginning let us examine a change in the binary representation of a number after we add 1 to it. Consider some positive integer x. Let $a_l a_{l-1} \ldots a_1 a_0$ be its binary representation, i.e. $x = 2^l a_l + 2^{l-1}a_{l-1} + \cdots + 2a_1 + a_0$. If $a_0 = 0$, then $x + 1 = 2^l a_l + 2^{l-1}a_{l-1} + \cdots + 2a_1 + a_0 + 1 = 2^l a_l + 2^{l-1}a_{l-1} + \cdots + 2a_1 + 1$, that is binary representation of such number is simply $a_l a_{l-1} \ldots a_1 1$. If we denote $x' = \left\lfloor \frac{x}{2} \right\rfloor$, i.e., in substance, number x without last binary digit, then representation of number $x + 1$ when $a_0 = 0$ is representation of number x' with appended digit 1. If $a_0 = 1$, then $x + 1 = 2^l a_l + 2^{l-1}a_{l-1} + \cdots + 2a_1 + a_0 + 1 = 2^l a_l + 2^{l-1}a_{l-1} + \cdots + 2a_1 + 2 = 2((2^{l-1}a_l + 2^{l-2}a_{l-1} + \cdots + a_1) + 1) + 0 = 2(x' + 1) + 0$, i.e. binary representation of $x + 1$ is a binary representation if $x' + 1$ with appended digit 0. So, binary representation of number $x + 1$ is either binary representation of number x' with single digit appended or binary representation of number $x' + 1$ with single digit appended. As a followup, we may conclude that in order to know the final state of

an automaton after processing binary representations of numbers x and $x+1$ we don't need to know number x itself, it's sufficient to know last digit of binary nsotation of x and final states of this automaton after processing binary representations of numbers x' and $x'+1$. So, for any automaton $A \in Lambda$ we may build new automaton $B \in Lambda$ such that its set of states is a set of pairs of states of automaton A and such that for any positive integer x if automaton A resulted in being in states a and b after processing binary representations of numbers x and $x+1$ respectively, then automaton B results in being in state (a,b) after processing binary representation of number x. If we choose a set of all pairs (a,b) such that b is terminal state in automaton A as a set of terminal states of automaton B, then automaton B will correspond to required equality $f(B,x) = f(A,x+1)$. Let us explicitly write down a structure of automaton B. Let $A = \langle \Sigma, Q, q_0, \delta, T\rangle$, $B = \langle \Sigma, Q', q_0', \delta', T'\rangle$. Then:

$$
\begin{aligned}
&Q' = Q \times Q\\
&q_0' = (q_0, \delta(q_0,1))\\
&\delta'((a,b),0) = (\delta(a,0), \delta(a,1))\\
&\delta'((a,b),1) = (\delta(a,1), \delta(b,0))\\
&T' = Q \times T
\end{aligned}
$$

Let us assure that automaton B indeed corresponds to the stated condition, i.e. for any $x \in \mathbb{N}$ $g(B,x) = (g(A,x), g(A,x+1))$. Denote by $len(x)$ length of binary representation of number x and carry out the proof with mathematical induction on $len(x)$. Basis. $len(x) = 1$. There is only one such integer — $x = 1$. For it: $g(B,1) = \delta'(q_0',1) = \delta'((q_0,\delta(q_0,1)),1) = (\delta(q_0,1), \delta(\delta(q_0,1),0)) = (\delta(q_0,1), \delta(q_0,10)) = (g(A,1), g(A,2)) = (g(A,x), g(A,x+1))$. Inductive step. Let $x' = \left\lfloor \frac{x}{2} \right\rfloor$. Then $len(x') = len(x) - 1$ and, according to induction hypothesis, $g(B,x') = (g(A,x'), g(A,x'+1))$. If last digit of x is 0, i.e. $x = 2x'$, then $g(B,x) = \delta'(g(B,x'),0) = \delta'((g(A,x'), g(A,x'+1)),0) = (\delta(g(A,x'),0), \delta(g(A,x'),1)) = (g(A,2x'), g(A,2x'+1)) = (g(A,x), g(A,x+1))$. Otherwise $x = 2x'+1$ and $g(B,x) = \delta'(g(B,x'),1) = \delta'((g(A,x'), g(A,x'+1)),1) = (\delta(g(A,x'),1), \delta(g(A,x'+1),0)) = (g(A,2x'+1), g(A,2(x'+1))) = (g(A,x), g(A,x+1))$, q.e.d. The last thing to note is that resulting automaton has exactly n^2 states, which is less than required $\left\lceil \frac{k}{2} \right\rceil n^2 + 2nk + 2k = n^2 + 2n + 2$. So, proof for the case $k = 1$ is complete.

Part 2. Let us select some integer $k > 1$. Denote $l = \lceil \log_2 k \rceil$, $m = 2^l$. Again examine changes in binary representation of some integer x after adding k to it. For convinience we will consider only case $x \geq m$, since cases $x < m$ will be worked around separetely anyway. Let $x' = \lfloor \frac{x}{m} \rfloor$, $x'' = \mathrm{mod}_m x$. Then binary representation of number x itself is binary representation of number x' with $bin_l(x'')$ appended to it. If $x'' + k < m$, then $x + k = mx' + (x'' + k)$ and binary representation of $x + k$ is simply binary representation of x' with $bin_l(x'' + k)$ appended to it. Otherwise $x+k = mx'+x''+k = m(x'+1)+(x''+k-m)$. Since $x'' < m$ and $k \leq m$, then $0 \leq x'' + k - m < m$, and binary representation of number $x + k$ is binary representation if $x' + 1$ with $bin_l(x'' + k - m)$ appended to it. So, binary representation of number $x+k$ is either binary representation of x' or binary representation of $x'+1$ with some l symbols appended, and that l symbols depend only on number x'', i.e., in substance, on last l symbols of binary representation of number x itself. So, in order to determine final state of some automaton after processing binary representation of number $x+k$ it is sufficient to know its final states after processing binary representations of numbers x' and $x' + 1$, and also last l digits of binary representation of number x.

These all gives us a possibility to build for any automaton $A \in Lambda$ new automaton that will be working essentially the same way as an automaton described in previous part, only processing representation of a number with delay of l symbols. More definitely: let $A = \langle \Sigma, Q, q_0, \delta, T \rangle$. Than we will choose a set $Q \times Q \times \{0 \ldots m-1\} \cup \{0 \ldots m-1\}$ as a set of states of new automaton B, $q'_0 = 0$ as its initial state. Let us write down transition function δ' of automaton B and prove that it conforms to our conditions.

$$\delta'(q,a) = \begin{cases} 2q + a & \text{if } 2q + a < m \\ (\delta(q_0, 1), \delta(q_0, 10), (2q + a - m)) & \text{if } 2q + a \geqslant m \end{cases}$$

$$\delta'((p_1, p_2, q), a) = \begin{cases} (\delta(p_1, 0), \delta(p_1, 1), (2q + a)) & \text{if } 2q + a < m \\ (\delta(p_1, 1), \delta(p_2, 0), (2q + a - m)) & \text{if } 2q + a \geqslant m \end{cases}$$

Lemma 1 *For any number $x \in \mathbb{N}$ $g(B, x) = x$ if $x < m$ and $g(B, x) = (g(A, \lfloor \frac{x}{m} \rfloor), g(A, \lfloor \frac{x}{m} \rfloor + 1), \mathrm{mod}_m x)$ otherwise.*

Proof. Proof will be held with mathematical induction on $len(x)$. Basis. $len(x) = 1$. There is only one such integer — $x = 1$. For it: $g(B, x) =$

$\delta'(q'_0, 1) = \delta'(0, 1) = 2 \cdot 0 + 1 = 1$, because $1 < m$. Inductive step. Let $len(x) = t$, binary representation of x be $a_{t-1}a_{t-2}\dots a_0$. There are three possible cases:

- $t \le l$. It means that $x < m$. Then $g(B, x) = \delta'(q'_0, a_{t-1}a_{t-2}\dots a_0) = \delta'(\delta'(q'_0, a_{t-1}a_{t-2}\dots a_1), a_0) = \delta'(g(B, \lfloor \frac{x}{2} \rfloor), a_0) = \delta'(\lfloor \frac{x}{2} \rfloor, a_0) = 2\lfloor \frac{x}{2} \rfloor + a_0 = x$, since $a_{t-1}a_{t-2}\dots a_1$ is binary representation of $\lfloor \frac{x}{2} \rfloor$, $len(\lfloor \frac{x}{2} \rfloor) = t - 1$.

- $t = l + 1$. It means that $m \le x < 2m$, correspondingly, $\lfloor \frac{x}{2} \rfloor < m$, $\lfloor \frac{x}{m} \rfloor = 1$. Then
  $g(B, x) = \delta'(q'_0, a_{t-1}a_{t-2}\dots a_0) = \delta'(\delta'(q'_0, a_{t-1}a_{t-2}\dots a_1), a_0) = \delta'(g(B, \lfloor \frac{x}{2} \rfloor), a_0) = \delta'(\lfloor \frac{x}{2} \rfloor, a_0) = (\delta(q'_0, 1), \delta(q'_0, 10), (2\lfloor \frac{x}{2} \rfloor + a_0 - m)) = (g(A, 1), g(A, 2), x - m) = (g(A, \lfloor \frac{x}{m} \rfloor), g(A, \lfloor \frac{x}{m} \rfloor + 1), x - m\lfloor \frac{x}{m} \rfloor) = (g(A, \lfloor \frac{x}{m} \rfloor), g(A, \lfloor \frac{x}{m} \rfloor + 1), \mathrm{mod}_m x)$.

- $t > l + 1$. It means that $x \ge m$, $\lfloor \frac{x}{2} \rfloor \ge m$. Then
  $g(B, x) = \delta'(q'_0, a_{t-1}a_{t-2}\dots a_0) = \delta'(\delta'(q'_0, a_{t-1}a_{t-2}\dots a_1), a_0) = \delta'(g(B, \lfloor \frac{x}{2} \rfloor), a_0) = \delta'((g(A, \left\lfloor \frac{\lfloor \frac{x}{2} \rfloor}{m} \right\rfloor), g(A, \left\lfloor \frac{\lfloor \frac{x}{2} \rfloor}{m} \right\rfloor + 1), \mathrm{mod}_m \lfloor \frac{x}{2} \rfloor), a_0)$.
  If we recall that $x = 2^{t-1}a_{t-1} + \dots + a_0$ and $m = 2^l$, then we may deduce $\left\lfloor \frac{\lfloor \frac{x}{2} \rfloor}{m} \right\rfloor = 2^{t-1-(l+1)}a_{t-1} + 2^{t-2-(l+1)}a_{t-2} + \dots + a_{l+1} = \lfloor \frac{x}{2m} \rfloor$, $\mathrm{mod}_m(\lfloor \frac{x}{2} \rfloor) = 2^{l-1}a_l + \dots + 2a_2 + a_1$. Then condition $2q + a < m$ from definition of transition function δ' becomes equivalent to condition $a_l = 0$. We may note that since a_l is the last digit of binary representation of number $\lfloor \frac{x}{m} \rfloor$, then $\lfloor \frac{x}{m} \rfloor = 2\lfloor \frac{x}{2m} \rfloor + a_l$. Also we may note that $2\mathrm{mod}_m \lfloor \frac{x}{2} \rfloor - a_l m + a_0 = 2(2^{l-1}a_l + 2^{l-2}a_{l-1} \cdots + a_1) - 2^l a_l + a_0 = 2^{l-1}a_{l-1} + \dots + 2a_1 + a_0 = \mathrm{mod}_m x$. So, if we consider both cases when this condition is true and when it is false, we get:

  - if $2\mathrm{mod}_m \lfloor \frac{x}{2} \rfloor + a_0 < m$ ($a_l = 0$), then: $\delta'((g(A, \left\lfloor \frac{\lfloor \frac{x}{2} \rfloor}{m} \right\rfloor), g(A, \left\lfloor \frac{\lfloor \frac{x}{2} \rfloor}{m} \right\rfloor + 1), \mathrm{mod}_m \lfloor \frac{x}{2} \rfloor), a_0) = (\delta(g(A, \left\lfloor \frac{\lfloor \frac{x}{2} \rfloor}{m} \right\rfloor), 0), \delta(g(A, \left\lfloor \frac{\lfloor \frac{x}{2} \rfloor}{m} \right\rfloor), 1), 2\mathrm{mod}_m \lfloor \frac{x}{2} \rfloor + a_0) = (g(A, 2\left\lfloor \frac{\lfloor \frac{x}{2} \rfloor}{m} \right\rfloor), g(A, 2\left\lfloor \frac{\lfloor \frac{x}{2} \rfloor}{m} \right\rfloor + 1), 2\mathrm{mod}_m \lfloor \frac{x}{2} \rfloor +$

$a_0) = (g(A, 2\left\lfloor\frac{x}{2m}\right\rfloor + a_l), g(A, 2\left\lfloor\frac{x}{2m}\right\rfloor + a_l + 1), 2\mathrm{mod}_m\left\lfloor\frac{x}{2}\right\rfloor + a_0 - a_l m) = (g(A, \left\lfloor\frac{x}{m}\right\rfloor), g(A, \left\lfloor\frac{x}{m}\right\rfloor + 1), \mathrm{mod}_m x)$

– if $2\mathrm{mod}_m\left\lfloor\frac{x}{2}\right\rfloor + a_0 \geq m$ ($a_l = 1$), then: $\delta'((g(A, \left\lfloor\frac{\left\lfloor\frac{x}{2}\right\rfloor}{m}\right\rfloor), g(A, \left\lfloor\frac{\left\lfloor\frac{x}{2}\right\rfloor}{m}\right\rfloor + 1), \mathrm{mod}_m\left\lfloor\frac{x}{2}\right\rfloor), a_0) = (\delta(g(A, \left\lfloor\frac{\left\lfloor\frac{x}{2}\right\rfloor}{m}\right\rfloor), 1), \delta(g(A, \left\lfloor\frac{\left\lfloor\frac{x}{2}\right\rfloor}{m}\right\rfloor + 1), 0), 2\mathrm{mod}_m\left\lfloor\frac{x}{2}\right\rfloor + a_0 - m) = (g(A, 2\left\lfloor\frac{\left\lfloor\frac{x}{2}\right\rfloor}{m}\right\rfloor + 1), g(A, 2(\left\lfloor\frac{\left\lfloor\frac{x}{2}\right\rfloor}{m}\right\rfloor + 1)), 2\mathrm{mod}_m\left\lfloor\frac{x}{2}\right\rfloor + a_0 - m) = (g(A, 2\left\lfloor\frac{x}{2m}\right\rfloor + a_l), g(A, 2(\left\lfloor\frac{x}{2m}\right\rfloor + a_l)), 2\mathrm{mod}_m\left\lfloor\frac{x}{2}\right\rfloor + a_0 - a_l m) = (g(A, \left\lfloor\frac{x}{m}\right\rfloor), g(A, \left\lfloor\frac{x}{m}\right\rfloor + 1), \mathrm{mod}_m x)$

□

Now we have to select such set T' of terminal states of automaton B that equality $f(B, x) = f(A, x + k)$ holds for all $x \in \mathbb{N}$. Concerning states $q \in 0 \ldots m - 1$ everything is quite simple — according to lemma, there is only one number that leads to this state, so we mark as terminal all of them that satisy the condition $f(A, q + k) = 1$. For states of the form (a, b, q) everything is quite more difficult. From lemma we know that if $g(B, x) = (a, b, q)$, then $g(A, \left\lfloor\frac{x}{m}\right\rfloor) = a$, $g(A, \left\lfloor\frac{x}{m}\right\rfloor + 1) = b$, $\mathrm{mod}_m x = q$. If we recall the way binary representation of number changes after adding k to it, we may find out that there are two cases:

- $q + k < m$. It means that binary representation of number $x + k$ is binary representation of number $\left\lfloor\frac{x}{m}\right\rfloor$ with $bin_l(q+k)$ appended to it. For such states we may select $(a, b, q) \in T' \iff \delta(a, bin_l(q + k)) \in T$.

- $q + k \geq m$. It means that binary representation of number $x + k$ is binary representation of number $\left\lfloor\frac{x}{m}\right\rfloor + 1$ with $bin_l(q + k - m)$ appended to it. For such states we may select $(a, b, q) \in T' \iff \delta(b, bin_l(q + k - m)) \in T$.

□

Part 3. Consider some automaton $A = \langle\Sigma, Q, q_0, \delta, T\rangle$ and some integer $k > 1$. Let $B = \langle\Sigma', Q', q'_0, \delta', T'\rangle$ be an automaton, built using

procedure described in previous part. Let us show that some of its states will be merged during standard minimization process.

Let $(p_1, p_2, q) \in Q'$ and $(p_1, p_2', q) \in Q'$. Let $q + k < m$. Since $m = 2^{\lceil \log_2 k \rceil}$, then $m < 2k$. So, $q < m - k < m - \frac{m}{2} = \frac{m}{2}$. Since q and $\frac{m}{2}$ are integers, then $q \leq \frac{m}{2} - 1$, so, $2q + 1 < m$. Then by construction $\delta'((p_1, p_2, q), a) = (\delta(p_1, 0), \delta(p_1, 1), 2q + a)$ for any $a \in \{0, 1\}$. Similarly, $\delta'((p_1, p_2', q), a) = (\delta(p_1, 0), \delta(p_1, 1), 2q + a)$. Also $(p_1, p_2, q) \in T' \iff \delta(a, bin_l(q + k)) \in T$ and $(p_1, p_2', q) \in T' \iff \delta(a, bin_l(q + k)) \in T$, i.e. for states (p_1, p_2, q) and (p_1, p_2', q) transition functions coincide and either both such states are terminal or both are non-terminal. So, during standard minimization process this states will not be separated and stay merged.

Let $(p_1, p_2, q) \in Q'$ and $(p_1, p_2', q) \in Q'$. Let $q + k \geqslant m$, $q < \lfloor \frac{m}{2} \rfloor$. Similarly to previous case $\delta'((p_1, p_2, q), a) = \delta'((p_1, p_2', q), a)$ for any $a \in \{0, 1\}$. So, if we consider set of all states of the form (p_1, t, q), where $t \in Q$, then during standard minimization process this set will be split into two subsets — subset of terminal states and subset of nonterminal states. Since for each of these subsets transition function is defined the same way on all states in this subset and states in this subset are either all terminal or all nonterminal, there is no way standard minimization process may distinguish states inside one subset, so they stay all merged after minimization process finishes.

Let $(p_1, p_2, q) \in Q'$ and $(p_1, p_2', q) \in Q'$. Let $q \geqslant \lfloor \frac{m}{2} \rfloor$, $2q + k + 1 < 2m$. Then $\delta'((p_1, p_2, q), a) = (\delta(p_1, 1), \delta(p_2, 0), 2q + a - m)$, $\delta'((p_1, p_2', q), a) = (\delta(p_1, 1), \delta(p_2', 0), 2q + a - m)$ for any $a \in \{0, 1\}$. Since $2q + k + 1 < 2m$, then $(2q + a - m) + k \leqslant (2q + 1 - m) + k = (2q + k + 1) - m < m$. According to first considered case, states $(\delta(p_1, 1), \delta(p_2, 0), 2q + a - m)$ and $(\delta(p_1, 1), \delta(p_2', 0), 2q + a - m)$ will stay merged during whole minimization process. Then, similarly to previous case we may note that since transition from states (p_1, p_2, q) and (p_1, p_2', q) leads to the states that stay merged during minimization process, then whole set of states (p_1, t, q) will be splited into only two subsets — subset of terminal states and subset of nonterminal states.

So, if $2q + k + 1 < 2m$, i.e. $q < m - \frac{k+1}{2} \leqslant m - \lfloor \frac{k+1}{2} \rfloor = m - \lceil \frac{k}{2} \rceil$, then set of states $\{(p_1, t, q) | t \in Q\}$ will be merged into no more than two states during standard minimization process. Since total amount of such sets is $n(m - \lceil \frac{k}{2} \rceil) \leqslant nk$, then after minimization they will be merged into

no more than $2nk$ states. Besides that automaton have sets of the form (p_1, p_2, q), where $q \geqslant m - \lceil \frac{k}{2} \rceil$ — there are no more than $\lceil \frac{k}{2} \rceil n^2$ of them in total, and also states from set $\{0 \dots m-1\}$ — there are m of them in total, that is not greater than $2k$. So, total number of states of resulting automaton after minimization is not greater than $\lceil \frac{k}{2} \rceil n^2 + 2nk + 2k$, q.e.d.

On the Behaviour of an Edge Number in a Power-Law Random Graph Near a Critical Point*

Elena V. Feklistova, Yuri L. Pavlov

Institute of Applied Mathematical Research
Karelian Research Centre RAS, Petrozavodsk, Russia

We consider power-law random graphs with N vertices. The degrees of vertices $1, \ldots, N$ are independent identically distributed random variables $\xi_1, \xi_2, \ldots, \xi_N$ drawn from the following law:

$$\mathbf{P}\{\xi_i \geq k\} = k^{-\tau}, \tag{1}$$

where $i = 1, \ldots, N, k = 1, 2, \ldots, \tau > 0$. This distribution determines the number of stubs for each vertex, i.e. the number of edges coming out of the vertex for which the connected vertices are not yet known. Since the sum of vertex degrees has to be even, one stub is added to a random vertex if the sum is odd. All stubs of vertices are numbered in an arbitrary order. The graph is constructed by joining each stub to another equiprobably to form edges. Such graphs are called power-law random graphs. Studies carried out in the past decades showed that power-law random graphs are deemed to be a good models of complex networks, e.g. Internet.

Let ζ_N stand for the total number of stubs, $\zeta_N = \xi_1 + \cdots + \xi_N$. It is clear that the number of edges in a graph is equal to $\zeta_N/2$. In [1] local limit theorems were proved for ζ_N as $N \to \infty$ and any fixed $\tau > 0$. This parameter has two critical points: $\tau = 1$ and $\tau = 2$. The structure and properties of a graph change significantly when $\tau = \tau(N)$ passes these points. We proved local limit theorems for ζ_N as $N \to \infty$ and $\tau(N) \to 2$. In particular, the next result are valid.

*The study was carried out with financial support from the Russian Foundation for Basic Research, grant 13-01-00009.

Theorem *Let* $N \to \infty,\ \tau(N) = 2 + y_N,\ y_N \to 0,\ y_N \ln N \to 0.$ *Then*

$$P\left\{\frac{\zeta_N - N\zeta(\tau(N))}{\sqrt{N \ln N}} < x\right\} \to \frac{1}{\sqrt{2\pi}} \int\limits_{-\infty}^{x} e^{-y^2/2} dy,$$

where $\zeta(\tau(N)) = \sum\limits_{k=1}^{\infty} k^{-\tau(N)}$.

References

[1] Y. L. Pavlov, E. V. Feklistova. Limit distributions of the edge number in random configuration graph. *European researcher*, 48(5-1) (2013), 1097-1100.

Principal (Left) Ideal Languages, Constants and Synchronizing Automata

Marina Maslennikova

Institute of Mathematics and Computer Science
Ural Federal University, Ekaterinburg, Russia

Emanuele Rodaro

Centro de Matemática, Faculdade de Ciências
Universidade do Porto, Porto, Portugal

Let $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ be a *deterministic finite automaton* (DFA), where Q is the *state set*, Σ stands for the *input alphabet*, and $\delta : Q \times \Sigma \to Q$ is the totally defined *transition function* defining the action of the letters in Σ on Q. The function δ is extended uniquely to a function $Q \times \Sigma^* \to Q$, where Σ^* stands for the free monoid over Σ. The latter function is still denoted by δ. In the theory of formal languages the definition of a DFA usually includes the *initial state* $q_0 \in Q$ and the set $F \subseteq Q$ of *terminal states*. We will use this definition when dealing with automata as devices for recognizing languages. A language $L \subseteq \Sigma^*$ is *recognized* (or *accepted*) by an automaton $\mathscr{A} = \langle Q, \Sigma, \delta, q_0, F \rangle$ if $L = \{w \in \Sigma^* \mid \delta(q_0, w) \in F\}$. We denote by $L[\mathscr{A}]$ the language accepted by the automaton $\mathscr{A}$.

A DFA $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ is called *synchronizing* if there exists a word $w \in \Sigma^*$ whose action leaves the automaton in one particular state no matter at which state in Q it is applied, i.e., $\delta(q, w) = \delta(q', w)$ for all $q, q' \in Q$. Any word w with this property is said to be *reset* for the DFA $\mathscr{A}$. For the last 50 years synchronizing automata received a great deal of attention. In 1964 Černý conjectured that every synchronizing automaton with n states possesses a reset word of length at most $(n-1)^2$. Despite

intensive efforts of researchers this conjecture still remains open. For a brief introduction to the theory of synchronizing automata we refer the reader to the survey [5].

Recently, in a series of papers [2, 4, 9, 6] a language theoretic approach to the study of synchronizing automata has been developed. In this abstract, we summarize some known facts from the above papers and present some new results. We denote by $\mathrm{Syn}(\mathscr{A})$ the language of synchronizing words for a given synchronizing automaton $\mathscr{A}$. It is well known that $\mathrm{Syn}(\mathscr{A})$ is regular [5]. Furthermore, it is a *two-sided ideal* (or simply an ideal) in Σ^*, i.e., $\mathrm{Syn}(\mathscr{A}) = \Sigma^* \mathrm{Syn}(\mathscr{A})\Sigma^*$. On the other hand, every two-sided regular ideal language L serves as a language of synchronizing words for some automaton. For instance, the minimal automaton of the language L is synchronized by L [2]. Thus synchronizing automata can be considered as a special representation of ideal languages. The complexity of such a representation is measured by the *reset complexity* $rc(L)$ which is the minimal possible number of states in a synchronizing automaton $\mathscr{A}$ such that $\mathrm{Syn}(\mathscr{A}) = L$. Every such automaton $\mathscr{A}$ is called *minimal synchronizing automaton* (for brevity, MSA). Let $sc(L)$ be the *state complexity* of L, i.e. the number of states in the minimal automaton recognizing L.

For every ideal language L we have $rc(L) \leq sc(L)$ (since the minimal automaton is synchronized by L). Moreover, there are languages L_n for every $n \geq 3$ such that $rc(L_n) = n$ and $sc(L_n) = 2^n - n$, see [2]. Thus representation of an ideal language by means of one of its MSA can be exponentially smaller than its "traditional" representation via minimal automaton. However, no reasonable algorithm is known for computing an MSA of a given language. One of the obstacles is that MSA is not uniquely defined. Furthermore, the problem of checking, whether a given synchronizing automaton with at least 5 letters is an MSA for a given ideal language, has recently been shown to be **PSPACE**-complete[6].

Another source of motivation for studying representations of ideal languages by means of synchronizing automata comes from the aforementioned Černý conjecture [2]. We can restate the Černý conjecture in terms of reset complexity as follows. If $||L||$ is the minimal length of words in an ideal language L then $rc(L) \geq \sqrt{||L||} + 1$. Actually, even a lower bound $rc(L) \geq \sqrt{||L||}/c$, for some constant $c > 0$, would be a major breakthrough for this conjecture. Thus a deeper understanding of reset

complexity may help to shed light on this longstanding conjecture. In this language theoretic approach to Černý conjecture, strongly connected synchronizing automata play an important role. Since the Černý conjecture holds true whenever it holds true for strongly connected automata, an important issue, risen in [4], is the problem of finding strongly connected synchronizing automaton whose set of reset words is equal to a given ideal language L. Indeed, while the minimal automaton recognizing an ideal language L is always a synchronizing automaton with a unique *sink* state (i.e. a state fixed by all letters), finding example of strongly connected synchronizing automata $\mathscr{A}$ with $\mathrm{Syn}(\mathscr{A}) = L$ is non-trivial task. In [9] it is proved that such an automaton always exists. The construction itself is non-trivial and rather technical. Furthermore, the upper bound on the number of states of the associated strongly connected automaton is very big.

Theorem 1 *Let I be an ideal language such that I^R (the ideal obtained applying the reversal operator) has state complexity n. Then there is a strongly connected synchronizing automata $\mathscr{B}$ with N states and* $\mathrm{Syn}(\mathscr{B}) = I$ *such that:*

$$N \leq m^{k2^n} \left(\sum_{t=2}^{n} m^{\binom{n}{t}} \right)^{2^n}$$

where $k = |\Sigma|$ and $m = \left(\frac{n^2+n}{2} + 1 \right)$.

The approach of [9] has the extra advantage of detaching Černý conjecture from the automata point of view. This is achieved by introducing a purely language theoretic notion of *reset left regular decomposition* of an ideal. We refer the reader to that paper for the definition of such decompositions and the details of the connection between decompositions of an ideal language and the Černý conjecture. Here we just note that the cardinality of the smallest reset left regular decomposition of an ideal L is equal to the size of the smallest strongly connected synchronizing automaton having L as the language of reset words. Furthermore, if we denote this common value by $\mathrm{rdc}(L)$, then $rc(L) \leq \mathrm{rdc}(L)$, and Černý conjecture holds if and only if $\mathrm{rdc}(L) \geq \sqrt{||L||} + 1$. Therefore, it is clear how important it is to study some issues like finding more effective constructions of these decompositions (or equivalently their associated automata), or to find more precise upper and lower bounds of $\mathrm{rdc}(L)$. The first attempts to approach

these questions has been made in [4], where it is considered the particular case of a principal ideal languages, i.e. a languages of the form $\Sigma^* w \Sigma^*$, for some word $w \in \Sigma^*$. Namely, the following theorem holds.

Theorem 2 *For the language $\Sigma^* w \Sigma^*$ there is a strongly connected automaton $\mathscr{B}$ with $|w| + 1$ states, such that $\mathrm{Syn}(\mathscr{B}) = \Sigma^* w \Sigma^*$. Such an automaton can be constructed in $O(|w|^2)$ time.*

Here, we provide a new result stating that the automaton $\mathscr{B}$ from the above theorem is in fact an MSA. More precisely, we have the following theorem.

Theorem 3 *Let $I = \Sigma^* w \Sigma^*$ be a principal ideal language, then $\mathrm{rdc}(I) = rc(I) = |w| + 1$.*

We also study principal left ideals, i.e. ideals of the form $\Sigma^* w$ for some word w. Such left ideals seem to play an important role in the Černý conjecture and the theory of synchronizing automata. Indeed, we characterize strongly connected automata via homomorphic images of automata belonging to a particular class of automata recognizing languages of the form $w^{-1}\Sigma^* w = \{u \mid wu \in \Sigma^* w\}$ for some $w \in \Sigma^*$. Namely, consider the class $\mathcal{L}(\Sigma)$ of all trim automata $\mathscr{A} = \langle Q, \Sigma, \delta, q_0, \{q_0\}\rangle$ such that $L[\mathscr{A}] = w^{-1}\Sigma^* w$ for some word $w \in \Sigma^*$. We recall that a DFA $\mathscr{A} = \langle Q, \Sigma, \delta, q_0, \{q_0\}\rangle$ is called *trim* whenever each state $q \in Q$ is accessible from q_0 and q_0 is accessible from each state $q \in Q$. Note that the following fact holds.

Lemma 1 *Let $\mathscr{A} \in \mathcal{L}(\Sigma)$ with $L[\mathscr{A}] = w^{-1}\Sigma^* w$. Then $\mathscr{A}$ is a strongly connected synchronizing automaton with $w \in \mathrm{Syn}(\mathscr{A})$.*

We recall that a homomorphism $\varphi : \mathscr{A} \to \mathscr{B}$ of automata is a map between the sets of states preserving the action of the two automata. Similarly, a congruence is an equivalence relation on the set of states which is compatible with the action of the letters. We have the following theorem.

Theorem 4 *Let $\mathscr{A} = \langle Q, \Sigma, \delta\rangle$ be a strongly connected synchronizing automaton. For any synchronizing word $w \in \mathrm{Syn}(\mathscr{A})$ of minimal length there is a DFA $\mathscr{B} \in \mathcal{L}(\Sigma)$ with $L[\mathscr{B}] = w^{-1}\Sigma^* w$ and*

$$\Sigma^* w \Sigma^* \subseteq \mathrm{Syn}(\mathscr{B}) \subseteq \mathrm{Syn}(\mathscr{A})$$

such that $\mathscr{A}$ is a homomorphic image of $\mathscr{B}$.

From which we derive the following

Corollary 1 *The class of strongly connected synchronizing automata are all and only all the homomorphic images of the class $\mathcal{L}(\Sigma)$ formed by the trim automata $\mathscr{A} = \langle Q, \Sigma, \delta, \{q_0\}, q_0\rangle$ such that $L[\mathscr{A}] = w^{-1}\Sigma^* w$ for some word $w \in \Sigma^*$.*

By $\mathrm{Cong}_k(\mathscr{B})$ we mean the (maybe empty) set of all congruences of automaton $\mathscr{B}$ of index k. Using Theorem 4 we can give another reformulation of the Černý conjecture using the automata from the class $\mathcal{L}(\Sigma)$.

Theorem 5 *Cerny's conjecture holds if and only if for any $\mathscr{B} \in \mathcal{L}(\Sigma)$ and $\rho \in \mathrm{Cong}_k(\mathscr{B})$ for all $k < \sqrt{\| \mathrm{Syn}(\mathscr{B})\|} + 1$ we have*

$$\| \mathrm{Syn}(\mathscr{B}/\rho)\| < \| \mathrm{Syn}(\mathscr{B})\|$$

In this regard we initiate the study of automata recognizing languages of the form $w^{-1}\Sigma^* w$ for some $w \in \Sigma^*$. In what follows we will assume that $|\Sigma| > 1$. For a given word w of length $|w| = n$ one may construct an n-state DFA $\mathscr{A}_w = \langle P(w), \Sigma, \delta, q_n, \{q_n\}\rangle$, where $P(w) = \{q_0, \ldots, q_n\}$ is the set of all prefixes of w including the empty word and the whole w. Let us assume that, for all i, $|q_i| = i$. The transition function δ is defined in such way that $\delta(q_i, a) = q_j$ for some i, j if and only if q_j is the maximal suffix of $q_i a$ that appears in w as a prefix. We prove the following proposition.

Proposition 1 *$\mathscr{A}_w$ is the minimal DFA recognizing the language*

$$L[\mathscr{A}_w] = w^{-1}\Sigma^* w \tag{1}$$

It it turns out that $\mathscr{A}_w$ is a finitely generated synchronizing automaton, i.e. $\mathrm{Syn}(\mathscr{A}_w) = \Sigma^* U \Sigma^*$ for some finite set of words U (for more details on finitely generated synchronizing automata see [8]). Furthermore, it can be easily seen that $w \in \mathrm{Syn}(\mathscr{A}_w)$. Now, in this context, we have that a word of the language recognized by the automaton is also a synchronizing word. Thus it is quite natural to ask in which cases the minimal automaton recognizing a given regular language L is synchronized by some word from L. Here we answer this question. Moreover, we prove a criterion for

the minimal automaton recognizing L to be synchronized by some word from L. We state this criterion in terms of the notion of a constant of L introduced by Schützenberger. Let $L \subseteq \Sigma^*$ be a regular language. A word $w \in \Sigma^*$ is a *constant* for L if the implication

$$u_1wu_2 \in L, u_3wu_4 \in L \Rightarrow u_1wu_4 \in L$$

holds for all $u_1, u_2, u_3, u_4 \in \Sigma^*$. We denote the set of all constants of L by $C(L)$. As it is mentioned in [10], the set $C(L)$ contains the ideal $Z(L) = \{w \mid \Sigma^* w \Sigma^* \cap L = \emptyset\}$. The notion of a constant is widely studied and finds applications in bioinformatics and coding theory [1, 5].

Constant words of a regular language L satisfy the property contained in Lemma 2 which has also been reported in [10].

Lemma 2 *Let $L \subseteq \Sigma^*$ be a regular language and let $\mathscr{A}$ be its minimal automaton with the state set Q and transition function δ. A word $w \in \Sigma^*$ is a constant for L if and only if $|\delta(Q, w)| \leq 1$.*

By this lemma it follows that if the automaton $\mathscr{A}$ is complete, then $w \in \Sigma^*$ is a constant of L if and only if w is a reset word for $\mathscr{A}$. Denote by $\overline{L} = \overline{L} = \Sigma^* \setminus L$. Recall that a language $L \subseteq \Sigma^*$ is a *right ideal* if it is non-empty and $L\Sigma^* \subseteq L$. We have the following characterization for the minimal automaton recognizing the language L to be synchronizing by some word of L.

Theorem 6 *The minimal automaton $\mathscr{A}$ recognizing a language L is synchronizing and $L \cap \mathrm{Syn}(\mathscr{A}) \neq \emptyset$ if and only if the following properties hold:*

(i) $C(L) \neq \emptyset$;

(ii) $\overline{L}$ does not contain right ideals.

Since the problem of checking whether or not $\overline{L}$ does not contain right ideals is polynomial time task, to understand the cost of checking the conditions of Theorem6 we need to study the following **CONSTANT** problem:

–*Input:* a regular language L over Σ, its minimal automaton $\mathscr{A}$.

–*Question:* is $C(L) \neq \emptyset$?

By Lemma 2 it is clear that the complex case is when $\mathscr{A}$ is a partial automaton, or equivalently, $\mathscr{A}$ contains a non-accepting sink state s. In

this case it remains to check whether or not $\mathscr{A}$ is synchronizing in the following sense: there exists a word w mapping the state set Q to two-element subset $\{s, q_w\}$ for some $q_w \neq s$. We show that this task can be solved in polynomial time. Whence, we get the following result.

Theorem 1 ***CONSTANT*** *can be solved in polynomial of n time.*

Now it is quite natural to ask how hard it is to check that there exists a constant of L of length at most ℓ, for some positive integer number ℓ. Again it is sufficient to consider only the case where $\mathscr{A}$ contains a non-accepting sink state s. We state formally the following **SHORT-CONSTANT** problem:

–Input: a DFA with a unique sink state s, positive integer number ℓ.

–Question: does there exist a word w of length at most ℓ bringing $\mathscr{A}$ to some subset of size at most two?

Proposition 2 ***SHORT-CONSTANT*** *is* ***NP****-complete.*

The proof of this last statement follows by a similar construction used in [3] to prove the **NP**-completeness of the problem of checking whether a given DFA is synchronized by some word of length at most ℓ.

Acknowledgements

The first author acknowledges support from the Presidential Programm for young researchers, grant MK-3160.2014.1.
The last author acknowledges support from the European Regional Development Fund through the programme COMPETE and by the Portuguese Government through the FCT – Fundação para a Ciência e a Tecnologia under the project PEst-C/MAT/UI0144/2013 as well as support from the FCT project SFRH/BPD/65428/2009.

References

[1] P. Bonizzoni, C. De Felice, R. Zizza. The structure of reflexive regular splicing languages via Schützenberger constants, *Theor. Comp. Sci.*, vol. 334 (2005), 71–98.

[2] J. Černý. Poznámka k homogénnym eksperimentom s konečnými automatami, *Mat.-Fyz. Cas. Slovensk. Akad. Vied.* V. **14** (1964), 208–216.

[3] D. Eppstein. Reset sequences for monotonic automata, *SIAM J. Comput., V.* **19** (1990), 500–510.

[4] V.V.Gusev, M.I.Maslennikova, E.V.Pribavkina. Principal Ideal languages and synchronizing automata. In *V.Halava, J.Karhumäki, Yu. Matiyasevich: the Special Issue of the RuFiDiM 2012, Fundamenta Informaticae*, DOI 10.3233/FI-2014-2015, V. 132 (2014), 95–108.

[5] A. de Luca, D. Perrin, A. Restivo and S. Termini. Synchronization and symplification, *Discrete Math.*, vol. 27 (1979), 297–308.

[6] M. Maslennikova. *Complexity of checking whether two automata are synchronized by the same language.* arXiv: 1405.3576v1, May 2014.

[7] M.I. Maslennikova. Reset Complexity of Ideal Languages. 2014. arXiv: 1404.2816 (published in M. Bieliková (eds.) *Int. Conf. SOFSEM 2012, Proc.* V. II, Institute of Computer Science Academy of Sciences of the Czech Republic, 33–44, 2012.)

[8] E. Pribavkina, E. Rodaro. Synchronizing automata with finitely many minimal synchronizing words, *Information and Computation* 209(3) (2011), 568-579.

[9] E. Rodaro, R. Reis. Regular ideal languages and synchronizing automata. In *J. Karhumäki, A. Lepistö, L. Zamboni (eds.) Proc. WORDS 2013.* LNCS, V. 8079, P.205–216, Springer, Heidelberg, 2013.

[10] M.P.Schützenberger. Sur certains opérations de fermeture dans les languages rationnels. *Sympos. Math.*, V.15, 245–253 [in French].

[11] M. V. Volkov. Synchronizing automata and the Černý conjecture. In: *C. Martín-Vide, F. Otto, H. Fernau (eds.), Languages and Automata: Theory and Applications.* LATA 2008. Lect. Notes Comp. Sci., Berlin, Springer. V.5196 (2008), 11–27.

Complexity of Checking whether Two Automata are Synchronized by the Same Language*

Marina Maslennikova

Institute of Mathematics and Computer Science
Ural Federal University, Ekaterinburg, Russia

Let $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ be a *deterministic finite automaton* (DFA), where Q is the *state set*, Σ stands for the *input alphabet*, and $\delta : Q \times \Sigma \to Q$ is the totally defined *transition function* defining the action of the letters in Σ on Q. The function δ is extended uniquely to a function $Q \times \Sigma^* \to Q$, where Σ^* stands for the free monoid over Σ. The latter function is still denoted by δ. In the theory of formal languages the definition of a DFA usually includes the *initial state* $q_0 \in Q$ and the set $F \subseteq Q$ of *terminal states*. We will use this definition when dealing with automata as devices for recognizing languages. A language $L \subseteq \Sigma^*$ is *recognized* (or *accepted*) by an automaton $\mathscr{A} = \langle Q, \Sigma, \delta, q_0, F \rangle$ if $L = \{w \in \Sigma^* \mid \delta(q_0, w) \in F\}$.

A DFA $\mathscr{A} = \langle Q, \Sigma, \delta \rangle$ is called *synchronizing* if there exists a word $w \in \Sigma^*$ whose action leaves the automaton in one particular state no matter at which state in Q it is applied: $\delta(q, w) = \delta(q', w)$ for all $q, q' \in Q$. Any word w with this property is said to be *reset* for the DFA $\mathscr{A}$. For the last 50 years synchronizing automata received a great deal of attention. In 1964 Černý conjectured that every synchronizing automaton with n states possesses a reset word of length at most $(n-1)^2$. Despite intensive efforts of researchers this conjecture still remains open. For a brief introduction to the theory of synchronizing automata we refer the reader to the recent surveys [5, 4].

In the present paper we focus on some complexity aspects of the theory of synchronizing automata. We denote by $\mathrm{Syn}(\mathscr{A})$ the language of reset

*The author acknowledges support from the Presidential Programm for young researchers, grant MK-3160.2014.1.

words for a given automaton $\mathscr{A}$. It is well known that $\mathrm{Syn}(\mathscr{A})$ is regular [5]. Furthermore, it is an *ideal* in Σ^*, i.e. $\mathrm{Syn}(\mathscr{A}) = \Sigma^* \mathrm{Syn}(\mathscr{A})\Sigma^*$. On the other hand, every regular ideal language L serves as the language of reset words for some automaton. For instance, the minimal automaton recognizing L is synchronized exactly by L [2]. Thus synchronizing automata can be considered as a special representation of an ideal language. Effectiveness of such a representation was addressed in [2]. The *reset complexity* $rc(L)$ of an ideal language L is the minimal possible number of states in a synchronizing automaton $\mathscr{A}$ such that $\mathrm{Syn}(\mathscr{A}) = L$. Every such automaton $\mathscr{A}$ is called a *minimal synchronizing automaton* (for brevity, MSA). Let $sc(L)$ be the number of states in the minimal automaton recognizing L. For every ideal language L we have $rc(L) \leq sc(L)$ [2]. Moreover, there are languages L_n for every $n \geq 3$ such that $rc(L_n) = n$ and $sc(L_n) = 2^n - n$ [2]. Thus the representation of an ideal language by means of a synchronizing automaton can be exponentially more succinct than the "traditional" representation via the minimal automaton. However, no reasonable algorithm is known for computing an MSA of a given language. One of the obstacles is that an MSA is not uniquely defined. For instance, there is a language with at least two different MSAs [2].

Let L be an ideal regular language over Σ with $rc(L) = n$. The latter equality means that there exists some n-state DFA $\mathscr{B}$ such that $\mathrm{Syn}(\mathscr{B}) = L$, and $\mathscr{B}$ is an MSA for L. Now it is quite natural to ask the following question: how hard is it to verify the condition $\mathrm{Syn}(\mathscr{B}) = L$? It is well known that the equality of the languages accepted by two given DFAs can be checked in polynomial of the size of automata time. However, the problem of checking the equality of the languages of reset words of two synchronizing DFAs turns out to be hard. Moreover, it is hard to check whether one particular ideal language serves as the language of reset words for a given synchronizing automaton. We state formally the SYN-EQUALITY problem:

–*Input:* synchronizing automata $\mathscr{A}$ and $\mathscr{B}$.

–*Question:* is $\mathrm{Syn}(\mathscr{A}) = \mathrm{Syn}(\mathscr{B})$?

One may notice now that the problem SYN-EQUALITY can be solved by the following naive algorithm. Indeed, we construct the power automata $\mathcal{P}(\mathscr{A})$ and $\mathcal{P}(\mathscr{B})$ for DFAs $\mathscr{A}$ and $\mathscr{B}$. Now it remains to verify that automata $\mathcal{P}(\mathscr{B})$ and $\mathcal{P}(\mathscr{A})$ accept the same language. However, the automaton $\mathcal{P}(\mathscr{A})$ has $2^n - n$ states, where n is the number of states in

the DFA $\mathscr{A}$. So we cannot afford to construct directly the corresponding power automata. Now we state formally the SYN-INCLUSION problem. We show that SYN-INCLUSION is in **PSPACE**. For more information about different complexity classes and classical computational problems we refer the reader to the source [3].

SYN-INCLUSION

–*Input:* synchronizing automata $\mathscr{A}$ and $\mathscr{B}$.

–*Question:* is $\mathrm{Syn}(\mathscr{A}) \subseteq \mathrm{Syn}(\mathscr{B})$?

Theorem 1 *SYN-INCLUSION is in* ***PSPACE****.*

Since SYN-INCLUSION belongs to the class **PSPACE**, we obtain that SYN-EQUALITY is in **PSPACE** as well. We prove that SYN-EQUALITY is a **PSPACE**-complete problem. Actually, we prove a stronger result, that it is a **PSPACE**-complete problem to check whether the language $\mathrm{Syn}(\mathscr{A})$ for a given automaton $\mathscr{A}$ coincides with the language $\mathrm{Syn}(\mathscr{B})$ for some particular automaton $\mathscr{B}$. The automaton $\mathscr{B}$ possesses just three states. Furthermore, it is an MSA for the language $\mathrm{Syn}(\mathscr{B})$. So we have the following theorem.

Theorem 2 *SYN-EQUALITY is* ***PSPACE****-complete.*

To prove that SYN-EQUALITY is a **PSPACE**-complete problem we reduce the following classical **PSPACE**-complete problem to the complement of SYN-EQUALITY. This problem deals with checking emptiness of the intersection of languages accepted by DFAs from a given collection [1].

FINITE AUTOMATA INTERSECTION

–*Input:* given n DFAs $M_i = \langle Q_i, \Sigma, \delta_i, q_i, F_i \rangle$, for $i = 1, \ldots, n$.

–*Question:* is $\bigcap_i L[M_i] \neq \emptyset$?

Let us notice that the Theorem 2 holds for automata over at least binary alphabet. Checking the equality of languages of reset words of two synchronizing automata over unary alphabet can be done in polynomial time. Now it is interesting to consider the SYN-STRICT-INCLUSION problem:

–*Input:* synchronizing automata $\mathscr{A}$ and $\mathscr{B}$.

–*Question:* is $\mathrm{Syn}(\mathscr{A}) \subsetneq \mathrm{Syn}(\mathscr{B})$?

Theorem 3 *SYN-STRICT-INCLUSION is* ***PSPACE****-complete.*

So the problem of constructing an MSA for a given ideal language is unlikely to be an easy task. Also we prove that the problem of checking the inequality $rc(L) \leq \ell$, for a given positive integer number ℓ, is in **PSPACE**. Here an ideal language L is presented by a DFA, for which L serves as the language of reset words. Let us note that checking the equality $rc(L) = 1$ or $rc(L) = 2$ is trivial. However, the problem of checking the inequality $rc(L) \leq 3$ turns out to be hard.

Theorem 4 *Let L be an ideal language and $\mathscr{A}$ a synchronizing DFA with at least 5 letters such that* $\mathrm{Syn}(\mathscr{A}) = L$*. The problem of checking the inequality $rc(L) \leq 3$ is* ***PSPACE****-complete.*

References

[1] D. Kozen *Lower bounds for natural proof systems.* In: Proc. of the 18th FOCS. 1977 P. 254–266.

[2] M.I. Maslennikova *Reset Complexity of Ideal Languages.* 2014. arXiv: 1404.2816 (published in M. Bieliková (eds.) Int. Conf. SOFSEM 2012, Proc. V. II, Institute of Computer Science Academy of Sciences of the Czech Republic. 2012 P. 33–44.)

[3] C.H. Papadimitriou *Computational complexity.* Reading–Menlo Park–N.Y.: Addison-Wesley, 1994.

[4] S. Sandberg *Homing and synchronizing sequences.*// In: M. Broy et al (eds.) Model-Based Testing of Reactive Systems, Lect. Notes Comput. Sci, Springer-Verlag, Berlin-Heidelberg-New York. 2005. V.3472. P.5–33.

[5] M. V. Volkov *Synchronizing automata and the Černý conjecture.* In: C. Martín-Vide, F. Otto, H. Fernau (eds.), Languages and Automata: Theory and Applications. LATA 2008. Lect. Notes Comp. Sci., Berlin, Springer. 2008. V.5196. P.11–27.

On Heawood-Type Problems for Maps with Tangencies

Gleb Nenashev

Chebyshev Laboratory, St. Petersburg State University
Saint Petersburg, Russia

Introduction

Questions about connection between the chromatic number of a graph and the existence of its drawing on some surface are among oldest problems of graph theory. For example, the 4 Color Conjecture ([1]-[3]), which is one of the most famous problems. P. J. Heawood [6] has found an upper bound on the chromatic number of a graph, that can be drawn without intersections on a surface of genus g ($g \geq 1$): it was shown that the chromatic number of such graph does not exceed $\frac{7+\sqrt{1+48g}}{2}$. Moreover, it was also shown [8], that this bound is tight.

Another well-known formulation of this problem is one for the dual graph, where the vertices are faces of the initial map (regions) and to faces are adjacent if and only if they have common part of boundary. We consider proper colorings of faces, i.e. such coloring that adjacent faces have different colors.

An equivalent formulation of this problem is as follows. Let the map on a surface is such that no 4 faces have a common point and we color faces such that faces having a common point must have different colors. (Clearly, two faces can have a common only on their common boundary).

We generalize this problem and consider the condition "any k faces have no common point" (or, what is the same, "no k maps are tangent to each other at same point") instead of "any 4 faces have no common point" in classic one.

Definition 1 *Let k and g be nonnegative integers. Let us denote by $\mathcal{B}_{k,g}$ the class of all maps on the surface of genus g such that any $k+1$ faces have no common point.*

Definition 2 *Let k be a nonnegative integer. We say that a graph is k-planar, if it can be drawn on the plane such that any edge intersects at most k other edges.*

In [4] O. V. Borodin found tight bound on the chromatic number for 1-planar graphs. The chromatic number of such graphs does not exceed 6, and, clearly, this bound is attained (for exampple, the complete graph K_6 is 1-planar).

Definition 3 *Let k and g be nonnegative integers. Let us denote by $\mathcal{A}_{k,g}$ the class of all graphs without loops and multiple edges which can be drawn on a surface of genus g, such that any edge intersects not more than k other edges.*

We denote by $\chi(\mathcal{A}_{k,g})$ and $\chi(\mathcal{B}_{k,g})$ the maximal chromatic number of a graph from the corresponding class.

In [5] the following bound was proved:

$$\chi(\mathcal{A}_{1,g}) \leq \frac{9+\sqrt{17+64g}}{2}.$$

Now we improve this bound and prove that $\chi(\mathcal{A}_{1,g}) = \chi(\mathcal{B}_{4,g})$.

Moreover, we construct nontrivial examples confirming that our bounds are tight. For this purpose we use the ideas of ribborn graphs and Kirchhoff's graphs.

Moreover, we prove that $\chi(\mathcal{B}_{k,g}) \leq \frac{2k+1+\sqrt{4k^2-12k+16gk+1}}{2}$ and $\chi(\mathcal{B}_{5,g}) \leq \chi(\mathcal{A}_{2,g}) \leq \frac{11+\sqrt{41+80g}}{2}$.

References

[1] K. Appel, W. Haken. Every Planar Map Is Four Colorable, *A.M.S. Contemp. Math.* 98 (1989).

[2] K. Appel, W. Haken. Every map is four colourable, Part I: Discharging. *Illinois Journal of Mathematics* 21 (1977), 429–490.

[3] K. Appel, W. Haken. Every map is four colourable, Part II: Reducibility. *Illinois Journal of Mathematics* 21 (1977), 491–567.

[4] O. V. Borodin Solution of Ringel's problems on vertex-face coloring of plane graphs and coloring of 1-planar graphs. *Met. Diskret. Anal.* 41 (1984), 12–26.

[5] G. V. Nenashev On a bound on the chromatic number of almost planar graph, *Combinatorics and graph theory.* Part V, Zap. Nauchn. Sem. POMI, 406, St. Petersburg, (2012), 95-106

[6] P. J. Heawood. Map colour theorem. *Quart. J. Math.* 24 (1890), 332–338.

[7] J. Pach, G. Tóth. Graphs drawn with few crossing per edge. *Combinatorica* 17 (1997), no. 3,427–439.

[8] G. Ringel, J. W. T. Youngs. Solution of the Heawood map-coloring problem *Proc. Nat. Acad. Sci. USA* 60 (1968), no. 2, 438–445.

[9] N. Robertson, D. Sanders, P. Seymour, R. Thomas. The Four-Colour Theorem. *J.Comb.Theory,* Series B **70** (1997), p.2–44.

Several Necessary Conditions For Uniformity of Finite Systems of Many-valued Logic

Pavel Tarasov

Lomonosov Moscow State University, Moscow, Russia

We use the following defenitions:
$E_k = \{0, \ldots, k-1\}$, $E_k^n = \underbrace{E_k \times E_k \ldots \times \ldots E_k}_{\text{n times}}$.

Denote by P_k the set of all functions $f : E_k^n \to E_k$ and by $P_{k,2}$ the set of all functions of the form $f : E_k^n \to E_2$.

For a formula Φ over a finite system of functions from P_k we define two measures: by $L(\Phi)$ we denote the number of occurrences of variables symbols in Φ (the complexity of the formula Φ) , and by $l(\Phi)$ we denote the depth of the formula Φ . We can define $l(\Phi)$ inductively:
1) if Φ consists of a single symbol (of variable or of constant), then $l(\Phi) = 0$;
2) if $\Phi = f(\Phi_1, \ldots, \Phi_n)$, then $l(\Phi) = (\max_{i=1,\ldots,n} l(\Phi_i)) + 1$.

Let A be a finite system of functions from $P_{k,2}$. For the function $f \in [A]$, we define

$$l_A(f) = \min(l(\Phi)) \quad , \quad L_A(f) = \min(L(\Phi)),$$

the minimum is taken over all formulas Φ, Φ' over A, realizing f.

The system of functions A will be called a uniform system if constants c and d exist such that for any function $f \in [A]$ we have the inequality

$$l_A(f) \leq c \log_2 L_A(f) + d,$$

(see definitions in [3]).

Khrapchenko has shown (see [4,5]) that all complete systems of boolean functions are uniform, the same result has been obtained in [7]. Wegener [6] proved, that all finite systems generating the class of all monotone Boolean functions are uniform. Ugol'nikov (see [3,8]) proved that all finite systems of boolean functions are uniform. Also, examples of non-uniform systems of many-valued logic are provided in this work. The same results has been obtained in [9]. Safin has shown (see [10]) uniformity for some finite systems generating some closed classes of many valued logic. This results were generalized by author in [11]. Moreover, in this work some sufficient conditions of uniformity of finite systems of functions from $P_{k,2}$ were obtained.

Let $f(x_1, \ldots, x_n)$ be a function from $P_{k,2}$. A function $g(x_1, \ldots, x_n) \in P_2$, is called the "projection" of f if for all $\widetilde{\alpha} \in E_2^n$ we have the equality $f(\widetilde{\alpha}) = g(\widetilde{\alpha})$. We denote the projection of function f by $\mathrm{pr}(f)$. In the same way, if A is a system of function from $P_{k,2}$, denote $\mathrm{pr}(A) = \bigcup\limits_{f \in A} \mathrm{pr}(f)$ (see more definitions in [1]).

Let $f(x_1, \ldots, x_n) \in P_{k,2}$, $i \in \{1, \ldots, n\}$. Denote

$$M_f^{x_i} = \{\mathrm{pr}f(\alpha_1, \ldots, \alpha_{i-1}, x, \alpha_i, \ldots, \alpha_{n-1}) | \widetilde{\alpha} \in E_k^{n-1}\},$$

$$V_f^{x_i} = \{\widetilde{\alpha} | \widetilde{\alpha} \in E_k^{n-1}, \quad \mathrm{pr}f(\alpha_1, \ldots, \alpha_{i-1}, y, \alpha_i, \ldots, \alpha_{n-1}) = x\}.$$

Let A be a finite system of monotone functions from $P_{k,2}$. We will say than A has property $\#$, if $q \geq 3$ exists, such that for any function $f(x_1, \ldots, x_n) \in A$, any $i \in \{1, \ldots, n\}$ and any $\widetilde{\alpha} \in V_f^{x_i}$, there exists function $g(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n, y_1, \ldots, y_q) \in [A]$, such that for any $\widetilde{\beta} \in V_f^{x_i}$, we have $\mathrm{pr}g(\widetilde{\beta}, \widetilde{y}) \notin [\{0, 1\}]$ and

1. if $\{0, x\} \in M_f^{x_i}$, then $\mathrm{pr}g(\widetilde{\alpha}, \widetilde{y}) \in M_{01} \setminus O^\infty$;
2. if $\{1, x\} \in M_f^{x_i}$, then $\mathrm{pr}g(\widetilde{\alpha}, \widetilde{y}) \in M_{01} \setminus I^\infty$.

By $q(A)$ we will denote minimal such number q.

We denote a partial order relation on E_k as follows: $1 \geq 0$, and all other elements of E_k are incomparable. Futher, in this paper by monotone functions we will mean functions preserving this partial order relation. Note, that if $f \in P_{k,2}$ is a monotone function, $\mathrm{pr}f$ is a monotone boolean function.

The main results of this work are these statements:

Theorem 1. *A finite system of monotone functions from $P_{k,2}$ is uniform only if it has property #.*

Theorem 2. *Let A be a finite system of monotone functions from $P_{k,2}$, such that A has property #. Than constants c and d exists, such that for any function $f \in [A]$ we have $l_A(f) \leq c \log^2 L_A(f) + d$.*

Theorem 3. *Let A be a finite system of monotone functions from $P_{k,2}$, A has property # and functions from A depends on no more than n variables. Than $q(A) \leq n^{k^n}$.*

References

[1] Yablonsky S. V. Vvedenie v diskretnuyu matematiku [Introduction to discrete mathematics]. Moscow, Nauka, 1986. 340 p

[2] Lau D, Function Algebras on Finite Sets. Springer-Verlag Berlin Heidelberg 2006

[3] Ugolnikov A. B. Complexity and depth of formulas realizing functions from closed classes. Fundamentals of Computation Theory Lecture Notes in Computer Science Volume 278, 1987, pp 456-461

[4] S. V. Yablonskii and V. P. Kozyrev, "Mathematical issues of cybernetics," in: Information Materials [in Russian], No. 19a, Nauchnyi Sovet po Kompleksnoi Probleme "Kibernetika," AN SSSR, Moscow (1968), pp. 3-15.

[5] V. M. Khrapchenko, "On relationship between complexity and depth of formulas," in: Methods of Discrete Analysis in Control System Design [in Russian], No. 32, Inst. Mat. SO AN SSSR, Novosibirsk (1978), pp. 76-94.

[6] Spira P. M. On time-hardware complexity tradeoffs for Boolean functions // Proc. 4th Hawai Symposium on System Sciences, North Hollywood, 1971, Western Periodicals Company, P. 525-527.

[7] Wegener I. Relating Monotone Formula Size and Monotone Depth of Boolean Functions // Information Processing Letters, 16. 1983. P. 41-42.

[8] Ugolnikov A. B. Polynomial equivalence of formulas from closed classes of two-valued logic // VII All-Union Conference "Problems of Cybernetics in theory": Proc. Reports. Part 1. Irkutsk: Irkutsk State University. 1985. 194-195.

[9] Ragaz M. E. Parallelizable algebras. Archiv fur mathematische Logik und Grundlagenforschung 26 (1986/7). P. 77-99

[10] Safin R. F. On the relation between depth and complexity of formulas in precomplete classes of k-valued logic. (Russian) Mat. Vopr. Kibern. 13, 223-278 (2004).

[11] B. Tarasov, "Certain sufficient conditions of uniformity for systems of functions of many-valued logic," Vestn. Mosk. Univ., Matem. Mekhan., No. 5, 41-46 (2013).

On Connection between Permutation Complexity and Factor Complexity of Infinite Words

Alexandr Valyuzhenich

Sobolev Institute of Mathematics, Novosibirsk, Russia

The notion of an infinite permutation was introduced in [1], where the periodic properties and low complexity of permutations were investigated. The notion of a permutation generated by an infinite non-periodic word and the notion of the permutation complexity of infinite word was introduced in [2]. In [3] Makarov calculated the permutation complexity of a well-known family of Sturmian words. In [5] Widmer calculated the permutation complexity of the Thue-Morse word.

For a word $\omega = \omega_1\omega_2\omega_3\ldots$ over the alphabet $\Sigma = \{0, 1\}$ we define the binary real number $R_\omega(i) = 0, \omega_i\omega_{i+1}\ldots = \sum_{k\geq 0}\omega(i+k)2^{-(k+1)}$. Let ω be a right infinite nonperiodic word over the alphabet Σ. We define the *infinite permutation* generated by the word ω as follows: $\delta = \langle \mathbb{N}, <_\delta, < \rangle$, where $<_\delta$ and $<$ are linear orders on $\mathbb{N}$. The order $<_\delta$ is defined as follows: $i <_\delta j$ if and only if $R_\omega(i) < R_\omega(j)$, and $<$ is the natural order on $\mathbb{N}$. Since ω is a non-periodic word, all $R_\omega(i)$ are distinct, and the definition above is correct. We say that a permutation $\pi = \pi_1\ldots\pi_n$ of $\{1, 2, \ldots, n\}$ is a *subpermutation* of length n of an infinite permutation δ if there exist i such that the numbers $R_\omega(i+1), \ldots, R_\omega(i+n)$ form the linear order is equal to π. Now we define the *permutation complexity* $\lambda(n)$ of the word ω (or equivalently, the factor complexity of the permutation δ_ω) as the number of its distinct subpermutations of length n. Recall that the *factor complexity* $C(n)$ of word ω is the number of its distinct subwords of length n.

In [2] was proved that $\lambda(n) \geq C(n-1)$. For Sturmian words we have equality $\lambda(n) = C(n-1) = n$.

The main result of this paper is that equality $\lambda(n) = C(n-1)$ holds for uniformly recurrent word ω if and only if ω is a Sturmian word. Moreover we obtain an alternative way to prove that $\lambda(n) = n$ for Sturmian words.

Theorem 2 *Let ω be an infinite uniformly recurrent word. Then $\lambda(n) = C(n-1)$ if and only if ω is a Sturmian word.*

References

[1] D.G. Fon-Der-Flaass and A.E. Frid. On periodicity and low complexity of infinite permutations. European J. Combin., 28(8):2106-2114, 2007.

[2] M.A. Makarov. On permutations generated by infinite binary words. Sib. Elektron. Mat. Izv., 3:304-311, 2006. (in Russian).

[3] M.A. Makarov. On the permutations generated by the Sturmian words. Sib. Math. J., 50(3):674-680, 2009.

[4] A. Valyuzhenich. Permutation complexity of the fixed points of some uniform binary morphisms // EPTCS 63 (2011), Proceedings of WORDS 2011, p. 257-264.

[5] S. Widmer. Permutation complexity of the Thue-Morse word. Adv. in Appl. Math. 47(2), pp. 309-329, 2011.

Научное издание

Третий
Российско-Финский симпозиум
по дискретной математике

Расширенные тезисы докладов

Печатается по решению Ученого совета
Федерального государственного бюджетного учреждения науки
Института прикладных математических исследований
Карельского научного центра
Российской академии наук