# SALOMAA SYMPOSIUM

Technical sessions

May 26, 2009

University of Turku, Lecture room XXI

## Program

| | |
|---|---|
| 12.00 – 12.40 | **Avraham Trakhtman**<br>*The Problems around Cerny conjecture and Road Coloring* |
| 12.40 – 13.20 | **Hellis Tamm**<br>*On Minimality of Biseparable Automata* |
| 13.20 – 14.00 | **Manfred Kudlek**<br>*Concurrent Finite Automata, Multiset Pushdown Automata, and Related Language Classes* |
| 14.00 – 14.30 | ***Coffee*** |
| 14.30 – 15.10 | **Solomon Marcus**<br>*Correctness and meaning; Cooperation or conflict?* |
| 15.10 – 15.50 | **Kai Salomaa**<br>*Lower bounds for the state complexity of nested word automata* |
| 15.50 – 16.30 | **Erzsébet Csuhaj-Varjú**<br>*Competence or Efficiency?* |

Abstracts:

# The Problems around Cerny conjecture and Road Coloring

***Avraham Trakhtman***

Let *G* be directed strongly connected finite graph of uniform outdegree (constant outdegree of any vertex) and let some coloring of edges of *G* turn the graph into deterministic complete automaton. Let the word *s* be a word in the alphabet of colors (considered also as letters) on the edges of *G* and let *Gs* be a mapping of vertices *G*.

A coloring is called synchronizing if for some word *s* (called synchronizing word) $|G\, s| = 1$. The synchronizing coloring turns the graph into a deterministic finite automaton possessing a synchronizing word.

The recent positive solution of the road coloring problem instantly posed a lot of generalizations. Among them are the recoloring for to minimize the length of the synchronizing word (the generalized Cerny problem), the *k*-synchronizing coloring for arbitrary integer *k* (coloring for mapping on *k* states, different solitions are found), the class of the graphs such that any its coloring is synchronizing, and some other.

A polynomial time algorithm of the road coloring has been based on recent positive solution of this old problem. and is implemented. For today are known some such algorithms. One can use new visualization program (of T. Bauer, N. Cohen and myself) for demonstration of the algorithm as well as for visualization of the transition graph of any finite automaton. The visual image presents some structure properties of the transition graph. This help tool is linear in the size of the automaton.

# On Minimality of Biseparable Automata

***Hellis Tamm***

Residual finite state automata (RFSA) are subclass of nondeterministic finite automata (NFA) with the property that every state of an RFSA defines a residual language of the language accepted by the RFSA. Recently, a notion of biRFSA – an RFSA such that its reversal automaton is also an RFSA – was introduced by Latteux, Roos, and Terlutte, who also showed that a subclass of biRFSA called biseparable automata consists of unique state-minimal NFAs for their languages. This was an improvement of the same result for bideterministic automata since the class of biseparable automata includes bideterministic automata as a strict subfamily.

In this talk, we present two other minimality results about biseparable automata. Our first result is that biseparable automata form the subclass of NFAs for which the lower bound provided by much-cited work of Glaister and Shallit (Inform. Proc. Lett., v. 59, 1996, 75–77) is tight. And second, we show that any unambiguous reversible biseparable automaton is a transition-minimal $\epsilon$-NFA. This result is an improvement of the same result obtained earlier for bideterministic automata (Int. J. Found. Comp. Sci., v. 19, 2008, 677–690).

## Concurrent Finite Automata, Multiset Pushdown Automata, and Related Language Classes

### *Manfred Kudlek*

Concurrent finite automata are finite automata with control by a Petri net instead by itself. Classes of accepted word languages and their relation to the Chomsky hierarchy and Petri net languages are presented. Multiset pushdown automata are the analogon to word pushdown automata, working with multisets instead of words. Classes of accepted multiset languages and their relation to other classes of multiset languages, as given by multiset grammars or Parikh images of word languages, are presented too.

## Correctness and meaning; Cooperation or conflict?

### *Solomon Marcus*

To be announced.

## Lower bounds for the state complexity of nested word automata

### *Kai Salomaa*

Finite automata operating on nested words were introduced by Alur and Madhusudan (2006). A deterministic nested word automaton equivalent to a given nondeterministic automaton with *O(n)* states may need 2^{*n*^2} states. The fact that the state complexity blow-up is larger than in the case of ordinary finite automata, roughly speaking, causes that the known upper bounds for the state complexity of basic operations on regular languages need not hold for languages of nested words.

We discuss techniques to establish lower bounds for the number of states of nested word automata. We consider both deterministic and nondeterministic state complexity of basic operations on nested word languages and discuss open problems.

## Competence or Efficiency?

### *Erzsébet Csuhaj-Varjú*

In this talk, we present results on the generative power and the size complexity of context-free cooperating distributed grammar systems (CD grammar systems) with cooperation protocols based on the so-called competence or the so-called efficiency of the component grammars.

The original motivation for the introduction of these constructs was to model the blackboard type problem solving systems by grammatical means. In these systems, the grammars generate a common sentential form in turn according to a cooperation protocol (a derivation mode). The grammars represent problem solving agents, the sentential form describes the actual state of the problem solving process, and the derivation corresponds to the solving of the given problem.

Most of the derivation modes that have been studied so far, were based on the so-called competence of the cooperating grammar. The notion is defined as the number of different nonterminals in the current sentential form which can be rewritten by the rules of the component grammar. This concept reflects the idea that the nonterminals correspond to unsolved subproblems and their replacement to a step to their solutions.

The efficiency of the component grammar is formulated as follows: a grammar is more efficient in solving a subproblem than some other one if starting from the same sentential form it derives a string that contains a smaller number of different nonterminals than the string derived by the other grammar, i.e., it is able to reduce the number of different open subproblems more efficiently than the other one.

We demonstrate that context-free CD grammar systems with competence-based or with efficiency-based cooperation protocols determine large language classes (sometimes the recursively enumerable language class), and for this purpose systems with small size are sufficient. We then analyze the role of the two concepts, i.e. the competence and the efficiency, in the derivations, and derive conclusions regarding the background and motivating areas of CD grammar systems.