T U M

INSTITUT FÜR INFORMATIK

36. Workshop über Komplexitätstheorie, Datenstrukturen und Effiziente Algorithmen

Anna Bernasconi, Stefan Bischof, Thomas Erlebach,
Tom Friedetzky, Volker Heun, Ernst W. Mayr, Michal Mnuk,
Martin Raab, Angelika Steger und Ulrich Voll

TUM-I9826
Oktober 98

TECHNISCHE UNIVERSITÄT MÜNCHEN

TUM-INFO-10-I9826-0/1.-FI

Druck: Institut für Informatik der
Technischen Universität München

36. Workshop über Komplexitätstheorie, Datenstrukturen und Effiziente Algorithmen

27. Oktober 1998, TU München, Raum 0602

09:30 *Rolf Klein (Hagen), K. Kriegel, C. Icking, F. Hoffmann*
Exploring a Simple Polygon

10:00 *Stefan Droste, Thomas Jansen, Ingo Wegener (Dortmund)*
Theoretische Analyse evolutionärer Algorithmen

10:30 PAUSE

10:50 *Petra Berenbrink, Friedhelm Meyer auf der Heide, Klaus Schröder (Paderborn)*
Allocating Weighted Jobs

11:20 *Andres Jakoby (Lübeck)*
The Average Time Complexity to Compute
Prefix Functions in Processor Networks

11:50 *Rudolf Fleischer (Saarbrücken)*
FUN with Implementing Algorithms

12:20 MITTAGSPAUSE

13:40 *Clemens Lautemann, Nicole Schweikardt, Thomas Schwentick (Mainz)*
Eine logische Charakterisierung von linearer Zeit
auf nichtdeterministischen Turing-Maschinen

14:10 *Clemens Lautemann (Mainz), Pierre McKenzie (Montreal), Thomas Schwentick (Mainz), Heribert Vollmer (Würzburg)*
The Descriptive Complexity Approach to LOGCFL

14:40 *Heinz Schmitz, Klaus W. Wagner (Würzburg)*
The Boolean Hierarchy over Level $1/2$ of the Straubing-Thérien Hierarchy

15:10 PAUSE

15:30 *Pierre McKenzie (Montreal), Klaus Reinhardt (Tübingen), V Vinay (Bangalore)*
Schaltkreise und kontextfreie Sprachen

16:00 *Stefan Edelkamp (Freiburg), Frank Reffel (Karlsruhe)*
BDDA*

16:30 *Oliver Kullmann (Frankfurt)*
Investigations on autark assignments

17:00 ENDE DES WORKSHOPS

Exploring a Simple Polygon

Rolf Klein

FernUniversität Hagen

Suppose a mobile robot is located in an unknown environment modelled by a simple polygon. The robot is point-shaped and equipped with a $360°$ on-board vision system. The robot's task is as follows. It starts from a given point, s, on the boundary, walks around in the polygon until each point has at least once been visible, and returns to s.

In this talk we present a competitive on-line strategy that ensures that the robot's path will always be shorter than 26.5 times the length of the optimum watchman tour (that could be computed off-line in time $O(n^2)$ if the polygon were known). Our analysis is based on a novel geometric structure. Let D be a connected subset of a simple polygon, P. Let A denote the set of all points inside P that can see two points of D at a right angle; we call A the *angle hull* of D with respect to P. Then the perimeter of A is at most twice the perimeter of D, and this bound is tight.

The talk is based on joint work with K. Kriegel, C. Icking, and F. Hoffmann.

Theoretische Analyse evolutionärer Algorithmen

Stefan Droste *Thomas Jansen* *Ingo Wegener*

Lehrstuhl Informatik II
Universität Dortmund
44221 Dortmund

Evolutionäre Algorithmen benutzen Grundprinzipien der natürlichen Evolution, wie Mutation, Rekombination und Selektion, um Optimierungsprobleme approximativ zu lösen. Durch eine Vielzahl von Experimenten und Anwendungen hat sich gezeigt, daß evolutionäre Algorithmen mit anderen Optimierverfahren konkurrieren können, weshalb sie immer öfter zur approximativen Optimierung benutzt werden. Jedoch ist die theoretische Analyse weit hinter den empirischen Resultaten zurückgeblieben.

Um theoretische Grundlagen zu legen, haben wir folgenden einfachen evolutionären Algorithmus (den sogenannten *(1+1) EA*) zur Maximierung einer Booleschen Zielfunktion $f : \{0,1\}^n \to \mathcal{R}$ untersucht:

1. Wähle $x \in \{0,1\}^n$ gleichverteilt zufällig aus.
2. Setze $y := x$.
3. Für alle $i \in \{1,\ldots,n\}$: Negiere y_i mit Wahrscheinlichkeit $1/n$.
4. Falls $f(y) \geq f(x)$, setze $x := y$.
5. Gehe zu Schritt 2.

Dieser Algorithmus verzichtet also auf Rekombination und stützt sich allein auf Mutation (Schritt 3) und Selektion (Schritt 4).

Schwerpunkt unserer bisherigen Untersuchungen ist die erwartete Laufzeit des (1+1) EA, bis das Maximum gefunden wird, auf zwei Klassen von Booleschen Zielfunktionen:

1. Lineare Funktionen, also Funktionen der Form $f(x_1,\ldots,x_n) = c + \sum_{i=1}^{n} w_i \cdot x_i$, wobei $c, w_1, \ldots, w_n \in \mathcal{R} \setminus \{0\}$.
2. Unimodale Funktionen, also Funktionen mit nur einem lokalen Maximum, d.h. nur einem Punkt x, so daß alle Punkte mit Hamming-Abstand Eins von x höchstens den gleichen Funktionswert wie x haben.

Dabei hat sich ergeben, daß der (1+1) EA auf linearen Funktionen eine erwartete Laufzeit von $\Theta(n\log(n))$ hat. Während die untere Schranke direkt aus der Betrachtung des "Coupon Collector's"-Problem folgt, muß für die obere Schranke der (1+1) EA durch einen geeigneten *Random Walk* modelliert werden, dessen Analyse das gewünschte Resultat liefert.

Für unimodale Funktionen war nicht klar, ob die erwartete Laufzeit des (1+1) EA ebenfalls $\Theta(n\log(n))$ ist. Wir konnten jedoch durch Betrachtung von sogenannten *Longpath*-Funktionen zeigen, daß die erwartete Laufzeit des (1+1) EA für unimodale Funktionen sogar exponentiell groß sein kann.

Weiterhin konnten wir zeigen, daß die Entscheidung, in Schritt 4 neue Punkte y mit gleichem Funktionswert (d.h. mit $f(y) = f(x)$) zu übernehmen, die erwartete Laufzeit gegenüber der Variante, die nur Verbesserungen akzeptiert, für manche Funktionen um Größenordnungen verringert.

Allocating Weighted Jobs

Petra Berenbrink *Friedhelm Meyer auf der Heide* *Klaus Schröder*

Heinz Nixdorf Institute and Dept. of Computer Science
University of Paderborn, D-33102 Paderborn, Germany
{pebe, fmadh}@uni-paderborn.de, ellern@hni.uni-paderborn.de

The classic balls-into-bins game considers the experiment of placing m balls independently and uniformly at random (i.u.r.) in n bins. For $m = n$, it is well known that the maximum load, i.e., the number of balls in the fullest bin is $\Theta(\log n / \log\log n)$, with high probability. It is also known (see [3]) that a maximum load of $\mathcal{O}\left(\frac{m}{n}\right)$ can be obtained for all $m \geq n$ if each ball is allocated in one (suitably chosen) of two (i.u.r.) bins. Stemann presents a distributed algorithm to find the "suitable" bin for each ball. The algorithm uses r communication rounds to achieve maximum load of max $\left\{\sqrt[r]{\log n}, \mathcal{O}\left(\frac{m}{n}\right)\right\}$, with high probability. Adler et al. [1] show that Stemann's protocol is optimal up to a constant factor for constant r.

We extend the above results in several directions: We generalize Stemann's upper bound to weighted balls: Let W^A (W^M) denote the average (maximum) weight of the balls. Furthermore, let $\Delta = W^A/W^M$. Then the optimal maximum load is $\Omega(m/n \cdot W^A + W^M)$. We present a protocol that achieves a maximum load of $\gamma \cdot \left(\frac{m}{n} \cdot W^A + W^M\right)$ using $\mathcal{O}\left(\frac{\log\log n}{\log(\gamma\cdot((m/n)\cdot\Delta+1))}\right)$ communication rounds. For uniform weights this matches the results of Stemann. In particular, we achieve a load of $\mathcal{O}\left(\frac{m}{n} \cdot W^A + W^M\right)$ using $\log\log n$ communication rounds, which is optimal up to a constant factor. An extension of the lower bound shows that our algorithm also reaches a load which is within a constant factor of the optimal load in the case of weighted balls.

The performance of our allocation protocol in the case of allocating m balls with a total weight of $W^A \cdot m$ is derived from the performance of Stemann's algorithm allocating a smaller set of $\delta \cdot m$ balls with uniform weight W^M. We show how to apply this analysis technique for other balls-into-bins games like the sequential process presented by [2], where each ball is allowed to choose d bins and is placed into the bin with the fewest load. Furthermore, we show that our technique can also be applied for balls-into-bins games for continuous generation processes where the number of balls is no longer fixed but new balls are generated (and consumed) as time passes. For all these models we get similar results expressed in terms of W^A and W^M.

All the balls-into-bins games model load balancing problems: The balls are jobs, the bins are resources, the task is to allocate the jobs to the resources in such a way that the maximum load is minimized. Our extensions to weighted balls allows us to extend previous bounds to models where resource requirements may vary. For example, if the jobs are computing tasks, their running time may vary. Applications of such load balancing problems occur, e.g. for client-server networks and for multimedia-servers using disk arrays.

References

[1] Micah Adler, Soumen Chakrabarti, Michael Mitzenmacher, Lars Rasmussen: Parallel Randomized Load Balancing. In Proceedings of the 27th Symposium on Theory of Computing, 1995.

[2] Y. Azar, A. Z. Broder, A. R. Karlin, E. Upfal: Balanced Allocations. In Proceedings of the 26th Symposium on Theory of Computing, 1994.

[3] Volker Stemann: Parallel Balanced Allocations. In Proceedings of the 8th Symposium on Parallel Algorithms and Architectures, 1996.

The Average Time Complexity to Compute Prefix Functions in Processor Networks[1]

Andreas Jakoby

Institut für Theoretische Informatik
Med. Universität zu Lübeck
jakoby@informatik.mu-luebeck.de

We analyze the average time complexity of evaluating all prefixes of an input vector over a given algebraic structure $\langle \Sigma, \otimes \rangle$. As a computational model networks of finite controls are used and a complexity measure for the average delay of such networks is introduced. Based on this notion, we then define the average case complexity of a computational problem for arbitrary strictly positive input distributions.

We give a complete characterization of the average complexity of prefix functions with respect to the underlying algebraic structure $\langle \Sigma, \otimes \rangle$ resp. the corresponding Moore-machine M. By considering a related reachability problem for finite automata it is shown that the complexity only depends on two properties of M, called *confluence* and *diffluence*. We prove optimal lower bounds for the average case complexity. Furthermore, a network design is presented that achieves the optimal delay for all prefix functions and all inputs of a given length while keeping the network size linear. It differs substantially from the known constructions for the worst case.

Our analysis yields that only three different cases can arise for the reachability question. We show that a prefix function can be computed either with a constant average delay, or with an average delay of order $\log \log n$, or in case of a nonconfluent function that no speedup is possible. This implies that an exponential speedup compared to the worst case is possible for several functions.

The processor networks investigated so far as it is generally the practice, are networks of bounded fanin and fanout. We expand our analysis to networks of unbounded fanout and obtain a corresponding characterization.

[1]supported by DFG Research Grant Re 672-2

FUN with Implementing Algorithms[1]

Rudolf Fleischer[2]

MPI für Informatik, Saarbrücken

rudolf@mpi-sb.mpg.de

In this paper, I would like to argue – based on my personal programming experience – that implementing algorithms is FUN. Moreover, it can lead to a better understanding of the problem at hand, it can help to avoid embarrassing mistakes, and animated algorithms can be used in class as a tool for teaching.

[1]This paper was presented at *FUN with algorithms*, Elba, June 18-20, 1998.

[2]The author was partially supported by the EU ESPRIT LTR Project No. 20244 (ALCOM-IT). He was also supported by a Habilitation Scholarship of the German Research Foundation (DFG).

Eine logische Charakterisierung von linearer Zeit auf nichtdeterministischen Turing Maschinen[1]

Clemens Lautemann *Nicole Schweikardt* *Thomas Schwentick*

{cl|nisch|tick}@informatik.uni-mainz.de

Eine komplexitätstheoretische Definition von *linearer Zeit* hängt (im Gegensatz zu z.B. polynomieller Zeit) stark vom jeweils betrachteten Maschinenmodell ab. Wir untersuchen die Klasse NTIME(n) aller in linearer Zeit auf nichtdeterministischen *Turing Maschinen* erkennbaren Stringmengen. Solange mehr als 1 Arbeitsband zur Verfügung steht, hängt diese Klasse weder von der genauen Anzahl der Arbeitsbänder ab, noch davon, ob $O(n)$ oder nur n Schritte erlaubt sind. Eine dem intuitiven algorithmischen Begriff von linearer Zeit vielleicht angemessenere Definition hat Grandjean mit der Klasse NLIN aller in linearer Zeit auf *Random-Access Maschinen* lösbaren Probleme gegeben.

Im Sinne der deskriptiven Komplexitätstheorie hat Grandjean u.a. folgende logische Charakterisierung von NLIN gefunden: Eine Stringmenge L liegt genau dann in NLIN, wenn es eine Formel der Form $\exists f_1 \cdots \exists f_k \, \exists R_1 \cdots \exists R_m \, \forall x \, \varphi$ gibt, die genau von den strings in L erfüllt wird. Dabei sind die f_i unäre Funktionssymbole, die R_i sind unäre Relationssymbole, und φ ist quantorenfrei.

Unserer Charakterisierung von NTIME(n) unterscheidet sich im wesentlichen von Grandjeans NLIN-Charakterisierung durch eine semantische Einschränkung der Funktionsquantoren.[2] Eine Funktion f nennen wir *decreasing*, falls $f(1)=1$ und $f(j)<j$ für alle $j>1$, und wir nennen f *non-crossing*, falls aus $f(j)<j' \leqslant j$ folgt, daß $f(j) \leqslant f(j')$, d.h., die Kanten im zu f gehörigen Graphen lassen sich überschneidungsfrei in die Halbebene oberhalb der Zahlengeraden einbetten. Wir erhalten folgende Charakterisierung: Eine Stringmenge L liegt genau dann in NTIME(n), wenn es eine Formel der Form $\exists f_1 \cdots \exists f_k \, \exists R_1 \cdots \exists R_m \, \forall x \, \varphi$ gibt, die genau von den strings in L erfüllt wird. Dabei ist die Quantifizierung über Funktionen eingeschränkt auf *decreasing non-crossing* Funktionen, und in φ sind keine Termgleichungen erlaubt, in denen unterschiedliche Funktionssymbole vorkommen.

Durch Beschränkung der Anzahl k von Funktionsvariablen ergibt sich eine strikte Hierarchie von $k{=}0$ bis $k{=}3$: Für $k{=}0$ erhält man gerade die regulären Sprachen, für $k{=}1$ die kontextfreien, und für $k \geqslant 3$ erhält man ganz NTIME(n). Die Trennung zwischen $k{=}2$ und $k{=}3$ ergibt sich u.a. aus Resultaten von Maas et al.

Unsere Logik erlaubt verschiedene Variationen, sowohl bzgl. der bei der Quantifizierung erlaubten Funktionen als auch bzgl. der Atome von φ. Eine dieser Varianten liefert ein Ehrenfeucht-Fraïssé-Spiel für NTIME(n), das für den *Duplicator* besonders einfach zu sein scheint. Dieses Spiel besteht aus nur drei Runden, in denen jeweils nur recht eingeschränkte Züge erlaubt sind.

[1] Eine Vollversion ist erhältlich unter `http://www.informatik.uni-mainz.de/~nisch/`

[2] Im Gegensatz zu Grandjean kodieren wir jedoch strings als Strukturen in der üblichen Weise.

The Descriptive Complexity Approach to LOGCFL

Clemens Lautemann
Institut für Informatik
Johannes-Gutenberg-Universität Mainz
55099 Mainz

Pierre McKenzie
Informatique et recherche opérationnelle
Université de Montréal
C.P. 6128, Succ. Centre-Ville
Montréal (Québec), H3C 3J7 Canada

Thomas Schwentick
Institut für Informatik
Johannes-Gutenberg-Universität Mainz
55099 Mainz

Heribert Vollmer
Theoretische Informatik
Universität Würzburg
Am Exerzierplatz 3
97072 Würzburg

Building upon the known generalized-quantifier-based first-order characterization of LOGCFL, we lay the groundwork for a deeper investigation. Specifically, we examine subclasses of LOGCFL arising from varying the arity and nesting of *groupoidal* quantifiers. Our work extends the elaborate theory relating *monoidal* quantifiers to NC^1 and its subclasses. In the absence of the BIT predicate, we resolve the main issues: we show in particular that no single outermost unary groupoidal quantifier with FO can capture all the context-free languages, and we obtain the surprising result that a variant of Greibach's "hardest context-free language" is LOGCFL-complete under quantifier-free BIT-free projections. We then prove that FO with unary groupoidal quantifiers is strictly more expressive with the BIT predicate than without. Considering a particular groupoidal quantifier, we prove that first-order logic with majority of pairs is strictly more expressive than first-order with majority of individuals. This answers an open question from [Barrington, Immerman, and Straubing, JCSS 41:274-396, 1990]. As a technical tool of independent interest, we define the notion of an aperiodic nondeterministic finite automaton and prove that FO translations are precisely the mappings computed by single-valued aperiodic nondeterministic finite transducers.

The Boolean Hierarchy over Level $1/2$ of the Straubing-Thérien Hierarchy

Heinz Schmitz *Klaus W. Wagner*

Theoretische Informatik
Universität Würzburg
Am Exerzierplatz 3
97072 Würzburg
{schmitz, wagner}@informatik.uni-wuerzburg.de

For some fixed alphabet A with $|A| \geq 2$, a language $L \subseteq A^*$ is in the class $\mathcal{L}_{1/2}$ of the Straubing-Thérien hierarchy if and only if it can be expressed as a finite union of languages $A^*a_1A^*a_2A^* \cdots A^*a_nA^*$, where $a_i \in A$ and $n \geq 0$. The class $\mathcal{L}_1$ is defined as the boolean closure of $\mathcal{L}_{1/2}$. It is known that the classes $\mathcal{L}_{1/2}$ and $\mathcal{L}_1$ are decidable. We give a membership criterion for the single classes of the boolean hierarchy over $\mathcal{L}_{1/2}$. From this criterion we can conclude that this boolean hierarchy is proper and that its classes are decidable. In finite model theory the latter implies the decidability of the classes of the boolean hierarchy over the class Σ_1 of the $\mathrm{FO}[<]$-logic. Moreover we prove a "forbidden-pattern" characterization of $\mathcal{L}_1$ of the type: $L \in \mathcal{L}_1$ if and only if a certain pattern does not appear in the transition graph of a deterministic finite automaton accepting L. We discuss complexity theoretical consequences of our results.

Classification: finite automata, concatenation hierarchies, boolean hierarchy, decidability, computational complexity

Schaltkreise und kontextfreie Sprachen

Klaus Reinhardt
Wilhelm-Schickhard Institut für Informatik
Eberhard-Karls-Universität Tübingen
Sand 13
72076 Tübingen
reinhard@informatik.uni-tuebingen.de

Pierre McKenzie
Departement d'informatique
University of Montreal
C.P. 6128 succursale Centre-ville
Montreal (Quebec) H3C 2J7, Canada
mckenzie@iro.umontreal.ca

V Vinay
Dept. of Computer Science and Automation
Indian Institute of Science
Bangalore-560 012 India
vinay@csa.iisc.ernet.in

Wir geben einen einfacheren Beweis für $SAC^1 = LOG(CFL)$ und $DAuxPDA-TIME(pol) = LOG(DCFL)$, welcher ohne die von Sudborough in [Sud78] verwendeten Mehrkopfautomaten [Iba73, Gal74] auskommt. Dazu geben wir Kellerautomaten (ohne zusätzliches Arbeitsband) an, welche einen Schaltkreis in angemessener Representation direkt auswerten können. Auch auf eine aufwendige Konstruktion, die den Schaltkreis auf logarithmische Tiefe balanciert (wie sie im Prinzip in [DR86, FLR96] vorkommt), kann verzichtet werden. Im deterministischen Fall verwenden wir hierzu die Multiplex-select-Gatter aus [FLR96]. Verlangt man von Schaltkreisen aus Multiplex-select-Gattern zusätzlich eine gewisse Selbstähnlichkeitseigenschaft, so benötigt deren Auswertung keinen Keller und wir erhalten damit eine Schaltkreischarakterisierung der Klasse L mit logarithmischer Tiefe. Durch eine Beschränkung der Breite von Schaltkreisen aus Multiplex-select-Gattern erhalten wir die Klasse NC^1.

Literatur

[DR86] P. W. Dymond and W. L. Ruzzo. Parallel RAMs with owned global memory and deterministic language recognition. In *Proc. 13th ICALP*, number 226 in LNCS, pages 95–104. Springer, 1986.

[FLR96] H. Fernau, K.-J. Lange, and K. Reinhardt. Advocating ownership. In V. Chandru, editor, *Proceedings of the 16th Conference on Foundations of Software Technology and Theoretical Computer Science*, volume 1180 of *LNCS*, pages 286–297. Springer, December 1996.

[Gal74] Z. Galil. Two–way deterministic pushdown automata and some open problems in the theory of computation. In *Proc. 15th IEEE Symp. on Switching and Automata Theory*, pages 170–177, 1974.

[Iba73] O. H. Ibarra. On two way multi-head automata. *Journal of Computer and System Sciences*, 7(1):28–36, 1973.

[Sud78] I. H. Sudborough. On the tape complexity of deterministic context-free languages. *Journal of the ACM*, 25:405–414, 1978.

BDDA*

Stefan Edelkamp[1]
Institut für Informatik
Albert-Ludwigs-Universität
Am Flughafen 17
D-79110 Freiburg
edelkamp@informatik.uni-freiburg.de

Frank Reffel[2]
Institut für Logik, Komplexität und Deduktionssysteme
Universität Karlsruhe
Am Fasanengarten 5, D-76128 Karlsruhe
reffel@ira.uka.de

The use of a lower bound estimate in the search has a tremendous impact on the size of the resulting search trees, whereas *OBDDs* can be used to efficiently describe sets of states based on their binary encoding.

We combine these two ideas into a new algorithm *BDDA**. It challenges both the breadth-first search using *OBDDs* and the traditional *A** algorithm. The problem with *A** is that in many application areas the set of states is too huge to be kept in main memory. In contrast, brute-force breadth-first search using *OBDDs* unnecessarily expands several nodes. In heuristic search the set of generated states is not far away from the states on the minimal solution path. However, a simple state enumeration does not take profit from this circumstance.

Therefore, we exhibit a new trade-off between time and space requirements and tackle the most important problem in heuristic search, to overcome space limitations while avoiding a strong penalty in time.

We study this effect in detail and estimate the worst-case complexity of the number of iterations within *BDDA**. Let f^* be the solution length to a given state space problem. Assuming that for every node the lower bound estimate does not exceed one plus the estimate of its successors we prove that the number of iterations is bounded by ${f^*}^2/3 + f^* + 1$. Since f^* is not known in advance, a good upper bound to f^* is necessary to predict the search efforts.

We modified a model checker to evaluate our approach in the $(n^2 - 1)$-Puzzle and within Sokoban.

References

[1] K.L. McMillan, *Symbolic Model Checking*, Kluwer Academic Press, 1993.

[2] P. E. Hart and N. J. Nilsson and B. Raphael, *A Formal Basis for Heuristic Determination of Minimum Path Cost*, IEEE Trans. on SSC, 1968(4), p. 100.

[1] Stefan Edelkamp is supported by DFG within graduate program on human and machine intelligence.
[2] Frank Reffel is supported by DFG within graduate program on controlability of complex systems.

Investigations on autark assignments

Oliver Kullmann

Johann Wolfgang Goethe-Universität, Fachbereich Mathematik
D-60054 Frankfurt am Main, Germany
kullmann@mi.informatik.uni-frankfurt.de
http://mi.informatik.uni-frankfurt.de/people/kullmann/kullmann.html

In [1] the structure of the monoid of autarkies of a conjunctive normal form (CNF) is investigated, and a sub-monoid of "linear autarkies" is introduced. A "maximal linear autarkie" can be computed in polynomial time by means of Linear Programming.

This leads to a canonical polynomial-time computable decomposition of a CNF into a "linearly independent" and a "linearly satisfiable" sub-clause-set: Linearly independent clause-sets" are a (poly-time decidable) generalization of minimally unsatisfiable clause-sets with interesting properties, while "linearly satisfiable" clause-sets are a new (poly-time decidable) class of satisfiable clause-sets.

In my talk I will present these theoretical results, and I will also discuss some (first) experimental results.

References

[1] Oliver Kullmann. Investigations on autarks assignments. Submitted to Discrete Applied Mathematics, October 1998.