COMPLEXITY ABSTRACTS 1996. Vol VI

Abstract

This is a collection of one page abstracts of recent results of interest to the Structural Complexity community. The purpose of this document is to spread this information, not to judge the truth or interest of the results therein.

A list of the titles of all the abstracts

Pinpointing computation with modular queries in the Boolean Hierarchy
The Boolean Isomorphism Problem
A Nonadaptive Program Checker for Permutation Group Intersection
Circuit closures for PP and PL
On the Query Complexity of Sets
One Help Bit Doesn't
Ranks of Inductions Correspond to Ranks of Recursions
Equivalence of Measures of Complexity Classes
Resource-Bounded Kolmogorov Complexity Revisited
Lindström Quantifiers and Leaf Language Definability
Lectures on Proof Theory
Resolution of Hartmanis' Conjecture for NL-Hard Sparse Sets
Constant Depth Circuits and the Lutz Hypothesis
Uniformly Hard Languages
Towards Theoretical Foundations of Software Eng: The Kolg. approach
On the Message Complexity of Interactive Proof Systems
A Small Span Theorem within P
Towards the actual relationship between NP and Exponential Time
Quantum Cosmology: When Are Two Wave Functions Distinguishable?
Genericity and Randomness over Feasible Probability Measures
The Isomorphism Problem for One-Time-Only Branching Programs
Randomness and Complexity (Ph.D. Thesis)
Genericity, Randomness and Polynomial Time Approximations
Resource Bounded Randomness and Computational Complexity

Pinpointing computation with modular queries in the Boolean Hierarchy

Manindra Agrawal, Abteilung Theoretische Informatik, Universität Ulm, Oberer Eselsberg, 89069 Ulm, Germany. (manindra@informatik.uni-ulm.de)
Richard Beigel, Yale University, Dept. of Computer Science, P.O. Box 208285, New Haven, CT 06520-8285, USA (beigel-richard@cs.yale.edu)
Thomas Thierauf, Abteilung Theoretische Informatik, Universität Ulm, Oberer Eselsberg, 89069 Ulm, Germany. (thierauf@informatik.uni-ulm.de)

Abstract Number 96-1

A *modular query* consists of asking how many (modulo m) of k strings belong to a fixed NP language. Modular queries provide a form of restricted access to an NP oracle. For each k and m, we consider the class of languages accepted by NP machines that ask a single modular query. Han and Thierauf (at Structures '95) showed that these classes coincide with levels of the Boolean hierarchy when m is even or $k \leq 2m$, and they determined the exact levels. Until now, the remaining case — odd m and large k — looked quite difficult. We pinpoint the level in the Boolean hierarchy for the remaining case; thus, these classes coincide with levels of the Boolean hierarchy for every k and m.
In addition we characterize the classes obtained by using an NP(l) acceptor in place of an NP acceptor (NP(l) is the lth level of the Boolean hierarchy). As before, these all coincide with levels in the Boolean hierarchy.

A full paper is available as ECCC TR96-001 at http://www.eccc.uni-trier.de/eccc/ or by email to the authors.

The Boolean Isomorphism Problem

Manindra Agrawal, Abteilung Theoretische Informatik, Universität Ulm, Oberer Eselsberg, 89069 Ulm, Germany. (manindra@informatik.uni-ulm.de)
Thomas Thierauf, Abteilung Theoretische Informatik, Universität Ulm, Oberer Eselsberg, 89069 Ulm, Germany. (thierauf@informatik.uni-ulm.de)

Abstract Number 96-2

We investigate the computational complexity of the *Boolean Isomorphism problem (BI)*: on input of two Boolean formulas F and G decide whether there exists a permutation of the variables of G such that F and G become equivalent.
Our main result is a one-round interactive proof for for the complement of BI, where the Verifier has access to an NP oracle. To obtain this, we use a recent result from learning theory by Bshouty *et.al.* that Boolean formulas can be learned probabilistically with equivalence queries and access to an NP oracle. As a consequence, BI cannot be Σ_2^p complete unless the Polynomial Hierarchy collapses. This solves an open problem posed by Borchert *et.al.* Further properties of BI are shown: BI has And- and Or-functions, the counting version, #BI, can be computed in polynomial time relative to BI, and BI is self-reducible.

A full paper is available by email to the authors.

A Nonadaptive Program Checker for Permutation Group Intersection

V. Arvind, Institute of Mathematical Sciences, Madras 600113, INDIA, (arvind@imsc.ernet.in)
J. Torán, Abtelilung Theoretische Informatik, Universität Ulm, D-89069 Ulm, GERMANY. (toran@informatik.uni-ulm.de)

Abstract Number 96-3

Let *GroupInt* denote the problem of checking whether the intersection of two permutation groups presented by generator sets is non-trivial, and let *UGroupFact* denote the problem of checking whether a given permutation π can be uniquely factored as a product $\phi\psi$, where ϕ and ψ are elements of two given permutation groups).
In this note we show that solutions for the search problem for *GroupInt* can be obtained in polynomial time with non adaptive queries to the corresponding decision problem. We also show that *UGroupFact* is polynomial-time truth-table equivalent to *GroupInt*. For the proofs we use algorithmic properties of the tower decomposition of permutation groups and wreath products of permutation groups.
Using these results we design nonadaptive program checkers for *GroupInt* and *UGroupFact*.

A full paper is not yet available.

Circuit closures for PP and PL

Richard Beigel, Dept. of Computer Science, Yale University (rjb@cs.yale.edu)
Bin Fu, Dept. of Computer Science, Yale University (fu-bin@cs.yale.edu)

Abstract Number 96-4

Wilson's model of oracle gates provides a framework for considering reductions whose strength is intermediate between truth-table and Turing. Improving on a stream of results by Beigel, Reingold, Spielman, Fortnow, and Ogihara, we prove that PL and PP are closed under NC^1 reductions. Then we ask whether they are closed under Boolean formula reductions. This is a nontrivial question despite $\mathrm{NC}^1 = \mathrm{BF}$, because that equality is easily seen not to relativize. We show that $\mathrm{P}^{\mathrm{PP}[\log^2 n/\log\log n]} \subseteq \mathrm{BF(PP)}$ and similarly for PL, and therefore it is unlikely that PL or PP is closed under Boolean formula reductions.

A full paper is not yet available.

On the Query Complexity of Sets

Richard Beigel, Dept. of Computer Science, Yale University (rjb@cs.yale.edu)
William Gasarch, Dept. of Computer Science, Univ. of MD, (gasarch@cs.umd.edu)
Martin Kummer, Institut für Logik, Komplexität und Deduktionssysteme, Universität Karlsruhe, (kummer@ira.uka.de) *Georgia Martin*, Dept. of Computer Science, Univ. of MD, (gam@cs.umd.edu) *Timothy McNicholl* Dept. of Mathematics, Ottawa University, (mcnichol@iso.ott.edu) *Frank Stephan* Mathematisches Institut, Ruprecht-Karls-Universität, (fstephan@math.uni-heidelberg.de)

Abstract Number 96-5

We consider the following sets with regard to how many queries to A are needed to compute them.
$\mathrm{ODD}_n^A = \{(x_1, \ldots, x_n) \mid |A \cap \{x_1, \ldots, x_n\}| \text{ is odd}\}$ and
$\mathrm{WMOD}(m)_n^A = \{(x_1, \ldots, x_n) \mid |A \cap \{x_1, \ldots, x_n\}| \not\equiv 0 \pmod m\}$.
If $A = K$ or A is semirecursive, we obtain the following: (1) ODD_n^A can be computed with n parallel queries (obviously) but cannot be computed with any fewer; (2) ODD_n^A can be computed with $\lceil \log(n+1) \rceil$ serial queries but cannot be computed with any fewer; (3) $\mathrm{WMOD}(m)_n^A$ can be computed with $\frac{2n}{m}$ parallel queries but cannot be computed with any fewer. (4) $\mathrm{WMOD}(m)_n^A$ can be computed with $\lceil \log(\frac{2n}{m} + 1) \rceil$ queries but cannot be computed with any fewer.
The lower bounds hold for any nonrecursive A r.e. and are derived *from* the lower bounds for A semirecursive. We also show that every truth table degree contains a set B such that ODD_n^B can be decided with one query to B. Hence, for bounded query complexity, how information is packaged is more important than Turing degree.
We investigate when extra queries add power. We show that, for several nonrecursive sets A, the more queries you can ask, the more sets you can decide; however, there are sets for which more queries do not help at all.

The MFCS version of the paper is available from gasarch@cs.umd.edu

One Help Bit Doesn't

Richard Beigel, Dept. of Computer Science, Yale University (rjb@cs.yale.edu)
Tirza Hirst, Dept. of Computer Science, Yale University (hirst-tirzah@cs.yale.edu)

Abstract Number 96-6

Nisan, Rudich, and Saks (FOCS '94) determined the asymptotic number of help bits needed in order for a depth-d decision forest to evaluate a Boolean function f on k inputs. Still, nothing was known about fixed k. We prove a gap between 0 and $\lfloor \log k \rfloor + 1$ help bits: either f can be evaluated in depth d without any help bits, or else $\lfloor \log k \rfloor + 1$ help bits are required in order to evaluate k instances in depth d. In particular, one help bit is as good as none. There exist functions for which this gap is best possible.

A full paper is not yet available.

Ranks of Inductions Correspond to Ranks of Recursions

Stephen J. Bellantoni (sjb@cs.utoronto.ca)

Abstract Number 96-7

The derivations of primitive recursive functions can be assigned ranks, based on considerations of predicativity. The current work shows how to rank proofs containing inductions, such that the ranking of proofs corresponds exactly to this ranking of recursions. A function can be proven convergent by a rank k proof if and only if it is a rank k function. The ranking of proofs is entirely syntactic, without any obvious reference to function growth rate, computation time, etc. Furthermore, induction formulas are allowed to have arbitrary alternations of unbounded quantifiers. Yet the first level of the logic hierarchy proves convergence of exactly the polynomial time computable functions.

Compared to the bounded induction rules of earlier weak subsystems of arithmetic, the present definitions permit a radical reduction in the syntactic restrictions placed on the induction formula. The only remaining restriction is a relatively mild one concerning occurrences of 0. In exchange, the overall amount of induction in the proof is restricted, by measuring the rank of the proof.

The correspondence between ranks of proofs and ranks of function derivations is achieved by a careful semantic examination of the ramification that seems to be implicit in arithmetic induction. Ramification levels will be defined here using certain subuniverses of the "standard ramified model". The rank of a proof is, in a sense, a measure of how many ramification levels are required in order to satisfy the conclusion of the proof. At the same time, rank is a measure of the use of induction througout the proof. The correspondence with function complexity is that a certain amount of induction is required in order to analyse the recursive definitions appearing in the proof.

Rank is defined syntactically, but without using explicit unary formulas $K_0, K_1, K_2, \ldots$ corresponding to ramification levels. Neither are ramification levels explicit in the semantic definition, as they are for intuitionistic semantics.

Further information is available by email to sjb@cs.utoronto.ca

Equivalence of Measures of Complexity Classes

Josef M. Breutzmann, Department of Mathematics and Computer Science, Wartburg College Waverly, Iowa 50677 U.S.A.,
Jack H. Lutz, Department of Computer Science, Iowa State University, Ames, Iowa 50011, U.S.A.

Abstract Number 96-8

The resource-bounded measures of complexity classes are shown to be robust with respect to certain changes in the underlying probability measure. Specifically, for any real number $\delta > 0$, any uniformly polynomial-time computable sequence $\vec{\beta} = (\beta_0, \beta_1, \beta_2, \ldots)$ of real numbers (biases) $\beta_i \in [\delta, 1-\delta]$, and any complexity class $\mathcal{C}$ (such as P, NP, BPP, P/Poly, PH, PSPACE, etc.) that is closed under positive, polynomial-time, truth-table reductions with queries of at most linear length, it is shown that the following two conditions are equivalent.

(1) $\mathcal{C}$ has p-measure 0 (respectively, measure 0 in E, measure 0 in E_2) relative to the coin-toss probability measure given by the sequence $\vec{\beta}$.

(2) $\mathcal{C}$ has p-measure 0 (respectively, measure 0 in E, measure 0 in E_2) relative to the uniform probability measure.

The proof introduces three techniques that may be useful in other contexts, namely, (i) the transformation of an efficient martingale for one probability measure into an efficient martingale for a "nearby" probability measure; (ii) the construction of a *positive bias reduction*, a truth-table reduction that encodes a positive, efficient, approximate simulation of one bias sequence by another; and (iii) the use of such a reduction to *dilate* an efficient martingale for the simulated probability measure into an efficient martingale for the simulating probability measure.

A full paper is available; please send requests to `lutz@cs.iastate.edu`.

Resource-Bounded Kolmogorov Complexity Revisited

Harry Buhrman, CWI. PO Box 94079, 1090 GB Amsterdam, The Netherlands. E-mail: buhrman@cwi.nl
Lance Fortnow, Department of Computer Science, University of Chicago, 1100 E. 58th St., Chicago, IL 60637. Email: fortnow@cs.uchicago.edu.

Abstract Number 96-9

Originally designed to measure the randomness of strings, Kolmogorov complexity has become an important tool in computability and complexity theory. A simple lower bound showing that there exist random strings of every length has had several important applications.

Early in the history of computational complexity theory, many people naturally looked at resource-bounded versions of Kolmogorov complexity. This line of research was initially fruitful and led to some interesting results. In particular, Sipser invented a new variation of resource-bounded complexity, CD complexity, where one considers the size of the smallest program that accepts the given string and no others. Sipser used CD complexity for the first proof that BPP is contained in the polynomial-time hierarchy.

We use algebraic techniques to give a new upper bound lemma for CD complexity without the additional advice required of Sipser's lemma. With this lemma, we can approximately measure the size of a set using CD complexity.

We define CND complexity, a variation of CD complexity where we allow nondeterministic computation. We prove a lower bound for CND complexity where we show that there exists an infinite set A such that every string in A has high CND complexity even if we allow access to A as an oracle. We use this lemma to prove some negative result on nondeterministic search vs. deterministic decision. Once we have these tools in place, we use them to unify several important theorems in complexity theory. We give straightforward proofs that BPP is in Σ_2^p (first proven by Gács), a randomized way to isolate satisfying assignment (first proved by Valiant and Vazirani) and relativized worlds where assignments to SAT cannot be found with non adaptive queries to SAT (first proved by Buhrman and Thierauf), and where EXP = NEXP but there exists a NEXP machine whose accepting paths cannot be found in polynomial time (first proved by Impagliazzo and Tardos).

These results in their original form require a great deal of time to fully understand the proof because either the ideas and/or technical details are quite complex. We show that by understanding resource-bounded Kolmogorov complexity, one can see full and complete proofs of these results without much additional effort. We also look at when polynomial-time C, CD, and CND complexity collide. We give a precise characterization of when we have equality of these classes, and some interesting consequences thereof.

An extended abstract is available from http://www.cs.uchicago.edu/~fortnow/papers.

Lindström Quantifiers and Leaf Language Definability

Hans-Jörg Burtschick, Fachbereich Informatik, TU–Berlin, Franklinstraße 28/29, D-10587 Berlin, Germany. (hjoerg@platine.cs.tu-berlin.de)
Heribert Vollmer, Theoretische Informatik, Universität Würzburg, Am Exerzierplatz 3, D-97072 Würzburg, Germany. (vollmer@informatik.uni-wuerzburg.de)

Abstract Number 96-10

We examine the expressive power of second-order formulae enriched by Lindström quantifiers over ordered finite structures. It turns out that the concept of leaf language definability can be logically characterized in this way. As an application, we get a new model-theoretic characterization of PSPACE.
We give a connection between model classes of second-order formulae and leaf language defined classes where the leaf language is specified in terms of a first-order formula. It turns out that the quantifier structure in both the second-order and first-order formula is identical. This result tightens the best up to now known leaf language characterization of the classes of the polynomial time hierarchy.
Finally, we generalize this connection for formulae that start with Lindström quantifiers.

A full paper is available by email to the authors.

Lectures on Proof Theory

Samuel R. Buss, University of California San Diego

Abstract Number 96-11

The 1994 McGill–Montréal Invitational Workshop on Complexity Theory was held at McGill University's Bellairs Research Institute in Barbados from March 6 to March 11, organized by Pierre McKenzie (U. de Montréal) and Denis Thérien (McGill U.).

The core of this year's workshop was a series of lectures given by Samuel R. Buss from the University California San Diego. Different topics on the theme of logic (mostly proof theory) and complexity theory were treated: interpolation theorems for propositional logic, natural proofs and split versions of bounded arithmetic, ALOGTIME algorithm for Boolean sentence evaluation, and cutting planes proof systems. This technical report consists of notes taken by the attendees on the content of the five two-hour lectures.

Available by anonymous ftp from ftp.cs.mcgill.ca, cd pub/tech-reports/library/reports/96, get TR.96.1.ps.gz. Send correspondence to Denis Thérien, School of Computer Science, McGill University, Montréal (Québec), Canada, H3A 2A7; `denis@cs.mcgill.ca`.

Resolution of Hartmanis' Conjecture for NL-Hard Sparse Sets

Jin-Yi Cai and D. Sivakumar, Dept. of Computer Science, State Univ. of NY at Buffalo, Buffalo, NY 14260, USA. ({cai,sivak-d}@cs.buffalo.edu)

Abstract Number 96-12

We resolve a conjecture of Hartmanis from 1978 about sparse hard sets for nondeterministic logspace (NL). We show that there exists a sparse hard set S for NL under logspace many-one reductions if and only if NL = L (deterministic logspace).

A set is sparse if it has at most a polynomial number of strings of each length n. In 1978, while studying the isomorphism problem for P and NL under logspace many-one reductions, Hartmanis conjectured that there is no sparse complete set for P or for NL under logspace many-one reductions (unless P = L or NL = L, respectively).

Very little was known concerning the above conjectures until the recent breakthrough by Ogihara (FOCS '95), who showed that if P has a sparse hard set under logspace many-one reductions, then P $\subseteq$ DSPACE[$\log^2 n$]. The conjecture of Hartmanis for P was finally settled by Cai and Sivakumar (FOCS '95), who showed that P has a sparse hard set under logspace many-one reductions iff P = L. Cai, Naik, and Sivakumar (STACS '96) gave a partial answer to the question for NL, and showed that if NL has a sparse hard set, then NL can be simulated in RNC^1 with oracle access to the reduction to the sparse set.

In this paper, we finally settle the Hartmanis conjecture for NL. We show that there is a sparse hard set for NL under logspace many-one reductions iff NL = L. Our proof uses the algebraic techniques of devised for the resolution of the conjecture for P. An additional crucial ingredient in the proof is the famous result of Immerman and Szelepcsényi that NL = *co*NL. Assuming the existence of a sparse hard set for NL, our proof gives a parallel algorithm for an NL-complete problem. This parallel algorithm can be implemented by a logspace-uniform circuit of polynomial-size, log-depth circuit that makes polynomially many parallel calls to the reduction from an NL set to the sparse set. This implies that if NL has a sparse hard set under logspace many-one reductions, then NL = L, and if NL has a sparse hard set under (logspace-uniform) NC^1 many-one reductions, then NL = (logspace-uniform) NC^1.

A draft paper is available by email to the authors.

Constant Depth Circuits and the Lutz Hypothesis

Jin-Yi Cai & D. Sivakumar, Department of Computer Science, SUNY at Buffalo, Buffalo, NY 14260. ({cai,sivak-d}@cs.buffalo.edu)
Martin Strauss, Department of Computer Science, Iowa State University, Ames, IA 50011. (mstrauss@cs.iastate.edu)

Abstract Number 96-13

The central hypothesis in the theory of resource-bounded measure is the assertion that NP does not have measure 0 in Exponential Time. This is a quantitative strengthening of the assertion NP $\neq$P. We show that the analog in P of this hypothesis fails dramatically. In fact, we show that NTIME$[n^{1/11}]$ has measure zero at P. These follow as consequences of our main theorem that for all d, the class of languages accepted by depth d Boolean circuits of size at most $2^{n^{1/(2d+6+o(1))}}$ has measure 0 at P.

Our proof is based on techniques from circuit complexity theory and pseudorandom generators. The main ingredients consist of Håstad's Switching Lemma, Nisan's pseudorandom generator secure against constant depth circuits, and error-correcting codes. For the class of constant depth circuits of the appropriate size bound, we first apply Håstad's Switching Lemma to conclude that circuits would have been significantly "demolished" by a random restriction. We then apply Nisan's pseudorandom generator to substitute the truly random restrictions by pseudorandom restrictions, and argue that the pseudorandom restrictions work almost as well as truly random restrictions. The pseudorandom restrictions obtained directly from Nisan's generator turn out to be not quite good enough for the purpose of constructing our martingales. For that purpose, we need some additional structures, especially error-correcting codes. It turns out that we can modify Nisan's construction so that the pseudorandom restrictions produced by the modified generator not only are pseudorandom but also form a certain error-correcting code, which makes the construction of our martingales possible.

Thus the Switching Lemma enters our proof in two ways. First, it implies that when a constant depth circuit is subject to a random restriction, it is very likely to collapse to a constant. Second, if we replace truly random restrictions by a family of pseudorandom restrictions, the circuit is also likely to collapse, because it couldn't tell the difference—and the reason why it could not tell the difference is precisely, again, (the decision tree version of) the Switching Lemma. In addition, the weaker pseudorandom family of restrictions (of roughly the same size) turn out to be more convenient in terms of the additional error-correcting properties that we need.

A draft paper is available by email to the authors.

Uniformly Hard Languages

Rod Downey, Department of Mathematics, Victoria University, P. O. Box 600, Wellington, New Zealand. Email: downey@math.vuw.ac.nz.
Lance Fortnow, Department of Computer Science, University of Chicago, 1100 E. 58th St., Chicago, IL 60637. Email: fortnow@cs.uchicago.edu.

Abstract Number 96-14

If P $\neq$ NP then we know that the NP-complete sets and the sets in P are disjoint. Must NP have any other sets? Ladner gives a positive answer by showing that if P $\neq$ NP then there exist NP sets not in P and not NP-complete.
In fact, Ladner actually proves a much more general result: For every recursive set A not in P, there exists a set B such that B is not in P, B polynomial-time honest many-one reduces to A, and A does not polynomial-time honest many-one reduce to B.
Ladner's proof works by creating a set B which alternates between looking like A and looking like some set in P such as $\emptyset$. These alternations may take a very long time and thus cause B to look like an easy set for very long sequences of input lengths.
Can one prove a Ladner-like theorem which does not have these large gaps of easiness in them? We will argue for the negative.
We define uniformly hard sets A where no set B in P can look like A for arbitrarily large polynomially long sequences of inputs. Formally, we say that $A \subseteq \Sigma^*$ is *uniformly hard* if for any language $B \in$ P, there is a k such that for all $m \geq 2$, $A(x) \neq B(x)$ for some x with $m \leq |x| \leq m^k$.
Ladner's proof fails miserably to produce uniformly hard sets no matter what set one starts with.
We show that if P = PSPACE then there exists minimal recursive uniformly hard sets under polynomial-time honest many-one reductions. This shows that Ladner's result must produce sets with long sequences of easy instances unless P = PSPACE.
Of course we do not believe P = PSPACE, but disproving this statement has eluded complexity theorists. Also, one could start with a relativized world where P = PSPACE to show that minimal uniformly hard sets exist in a relativized world. All of the results in the area, including Ladner's, relativize.
We show that a strange quirk in Ladner's proof that causes a hard set with long stretches of easiness is a quirk that cannot be eliminated.

An extended abstract is available from http://www.cs.uchicago.edu/~fortnow/papers.

Towards Theoretical Foundations of Software Engineering Heuristics: The Use of Kolmogorov Complexity

Ann Q. Gates, Vladik Kreinovich, and Luc Longpré, Department of Computer Science, University of Texas at El Paso, El Paso, TX 79968, USA. ({agates,vladik,longpre}@cs.utep.edu)

Abstract Number 96-15

We show that several heuristic techniques for software testing that have been developed in software engineering can be rigorously justified if we use Kolmogorov complexity $C(x)$ to formalize the terms "simple" and "random" that these techniques use. The successful formalization of simple heuristics is a good indication that Kolmogorov complexity may be useful in formalizing more complicated heuristics as well.

Definition 1. By a *software testing situation*, we mean a triple (p, P, t), where:
• p is a program (that transforms binary sequences into binary sequences); this program will be called a *tested program*;
• P is a binary property $P(x, y)$; this property is called a *specification*, or *desired property*;
• t is a program that, given x and y, checks whether $P(x, y)$ is true.
By an *input*, we mean a word x. We say that a program p *satisfies the specifications for an input* x if $t(x, p(x))$ ="true". We say that a program *satisfies specifications* if $t(x, p(x))$ ="true" for all x.

THEOREM 1. *There exists a number c with the following property: for every software testing situation (p, P, t), if p satisfies specifications for all inputs x with $C(x) \le c + l(p) + l(t)$, then the program p satisfies specifications for all possible inputs.*

Definition 2. Let integer $L > 0$ and $C > 0$ be fixed.
• We say that an input x is *random* if its length is equal to L ($l(x) = L$), and $C(x) \ge l(x) - C$.
• We say that $x_1, \ldots, x_k$ is a *sequence of k random words* if $l(x_1) = \ldots = l(x_k) = L$ and $C(x) \ge l(x) - C$, where $x = x_1 \ldots x_k$ is a concatenation of the words $x_1, \ldots, x_k$.
• We say that a program p *satisfies specifications for k random inputs* if for some sequence of k random words, $t(x_i, p(x_i))$ is true for all $i = 1, \ldots, k$.

THEOREM 2. *There exists a number c with the following property: If a program p satisfies specifications t for k random inputs, then the fraction F of all x for which $p(x)$ satisfies this specification satisfies the inequality $F \ge 2^{-a(k)}$, where $a(k) = (c + l(p) + l(t) + \log_2(k) + 2 + C)/k$.*

A draft is available by email to vladik@cs.utep.edu; a full paper will be available shortly.

On the Message Complexity of Interactive Proof Systems

Oded Goldreich, Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot, ISRAEL, (oded@wisdom.weizmann.ac.il).
Johan Håstad, Department of Computer Science, Royal Institute of Technology, 10044 Stockholm, SWEDEN, (johanh@nada.kth.se).

Abstract Number 96-16

We investigate the computational complexity of languages which have interactive proof systems of bounded message complexity. In particular, we show that

- If L has an interactive proof in which the total communication is bounded by $c(n)$ bits then L can be recognized a probabilitic machine in time exponential in $O(c(n) + \log(n))$.
- If L has an AM-proof in which the prover sends $c(n)$ bits then L can be recognized a probabilitic machine in time exponential in $O(c(n)\log(c(n)) + \log(n))$.
- If L has an interactive proof in which the prover sends $c(n)$ bits then L can be recognized a probabilitic machine with an NP-oracle in time exponential in $O(c(n)\log(c(n)) + \log(n))$.

A full version is available from ECCC (http://www.eccc.uni-trier.de/eccc/).

A Small Span Theorem within P

Wolfgang Lindner and Rainer Schuler, Abteilung Theoretische Informatik, Universität Ulm, Oberer Eselsberg, 89069 Ulm, GERMANY (lindner@informatik.uni-ulm.de, schuler@informatik.uni-ulm.de)

Abstract Number 96-17

The development of Small Span Theorems for various complexity classes and reducibilities plays a basic role in (resource bounded) measure-theoretic investigations of efficient reductions. A Small Span Theorem for a complexity class $\mathcal{C}$ and reducibility $\leq_r$ is the assertion that, for all sets A in $\mathcal{C}$, at least one of the cones below or above A is a negligible small class with respect to $\mathcal{C}$, where the cones below or above A refer to the sets $\{B : B \leq_r A\}$ and $\{B : A \leq_r B\}$, respectively. That is, a Small Span Theorem rules out one of the four possibilities of the size of upper and lower cones for a set in $\mathcal{C}$.

Here we show two Small Span Theorems for polynomial-time complexity classes and sublinear-time reducibilities, namely a Small Span Theorem for P and uniform projections, and for P^{NP} and DLOGTIME-transformations. (Reducibility under uniform projections is a logarithmic-time uniform version of the algebraic projections of Valiant.)

We use the recent formulation of resource-bounded measure of Allender and Strauss which allows meaningful notions of measure on polynomial-time complexity classes [FOCS '94]. The definition involves polylogarithmic-time bounded machines with certain restrictions concerning the access to the input. We compare the two possibilities allowing the machine adaptive or non-adaptive access to the input. (That is, in the latter case the bits queried by the machine must not depend on the actual input but only on its length.)

A full paper is available by email to the authors

Towards the actual relationship between NP and Exponential Time

Gerhard Lischke, Institut fuür Informatik, Friedrich-Schiller-Universität Jena, Ernst-Abbe-Platz 3, 07743 Jena, GERMANY. (lischke@minet.uni-jena.de)

Abstract Number 96-18

Knowing that two computational complexity classes $\mathcal{A}$ and $\mathcal{B}$ are separated we consider three aspects of the quality of this separation:
1. Is there a strict inclusion $\mathcal{A} \subset \mathcal{B}$ or $\mathcal{B} \subset \mathcal{A}$ or do we have incomparability with respect to inclusion between $\mathcal{A}$ and $\mathcal{B}$?
2. Do there exist immune sets in the difference between $\mathcal{A}$ and $\mathcal{B}$ or not (i.e. is it a strong or a weak separation)?
3. Do there exist sparse sets in the difference between $\mathcal{A}$ and $\mathcal{B}$ or not?
Combining these aspects we get 24 different cases for the exact relationship between separated classes $\mathcal{A}$ and $\mathcal{B}$.
We transfer these questions to the relationship between NP and EL_k for arbitrary fixed $k \geq 1$, where the latter denotes the exponential time class where the exponential is a polynomial of degree k. We show that only 8 of the 24 cases are possible to occur in the real nonrelativized world as well as in any relativized world. We do not know which of these 8 cases is the actual one in the real nonrelativized world but we can construct for every of them such recursive oracles, so that exactly this case is true under the appropriate relativization. Thus the question does arise: which of the 8 cases can we expect under relativization if we choose the oracle randomly? We show that for almost all oracles A, the classes NP^A and EL_k^A are incomparable with respect to inclusion, and in both classes there are sparse sets which are immune with respect to the other class.

A full paper is available by email to lischke@minet.uni-jena.de

Quantum Cosmology: When Are Two Wave Functions Distinguishable?

Luc Longpré, Vladik Kreinovich, Department of Computer Science, University of Texas at El Paso, El Paso, TX 79968 ({longpre,vladik}@cs.utep.edu)

Abstract Number 96-19

Traditional quantum mechanics (QM) predicts probabilities of different events. If we describe an elementary particle, then, experimentally, these probabilities mean that if we repeat the same measurement procedure with multiple particles in the same state, the resulting sequence of measurement results will be random w.r.t. the corresponding probability measure. In quantum cosmology, QM is used to describe the world as a whole; we have only one copy of the word, so multiple measurements are impossible. How to interpret these probabilities?

The Universe consists of many parts with very weak interaction between them. Therefore, with a good accuracy, we can assume that these parts are independent and can, therefore, be described by separate wavefunctions $\varphi_1, ..., \varphi_n, ...$ The state of the Universe corresponds to the sequence $\varphi = (\varphi_1, ...)$ of functions φ_i from the Hilbert space L^2. In each part of the world, we can perform a separate measurement A_i. Standard formulas of QM provide us with a probability measure $\mu_i(A_i, \varphi_i)$ that describe the probabilities of different results of $i-$th measurement: namely, this measure is concentrated on the eigenvalues λ_{ij} of the operator A_i, and the probability of $j-$th eigenvalue λ_{ij} if equal to $|P_{ij}(\varphi_i)|^2$, where P_{ij} denotes the projection on the corresponding eigenspace. Since we assumed that the parts are independent, we can assume that these random variables (measurement results) are also independent, i.e., that the resulting measure $\mu(A, \varphi)$ on the sequences of real numbers (measurement results) is equal to the Cartesian product of the measures μ_i. We then require that the actual sequence of results is *random* w.r.t. $\mu(A, \varphi)$ (random in the sense of Kolmogorov-Martin-Löf or in a more general sense).

The problem that we want to solve is: *When are two states* $\varphi = (\varphi_1, ...)$ *and* $\psi = (\psi_1, ...)$ *distinguishable?*

In quantum measurements, some information about the state is lost, so, if we choose the wrong variables A_i to measure, we may lose this distinction. Hence, the question is: when is it possible to have a sequence of measurements $A_1, ..., A_n, ...$ for which no sequence can be random w.r.t. both probability measures $\mu(A, \varphi)$ and $\mu(A, \psi)$? The answer is: iff $\sum(1 - |(\psi_i, \varphi_i)|) = \infty$.

A full paper will be available shortly.

Genericity and Randomness over Feasible Probability Measures

Amy K. Lorentz and *Jack H. Lutz*, Department of Computer Science, Iowa State University, Ames, IA 50011

Abstract Number 96-20

This paper investigates the notion of resource-bounded genericity developed by Ambos-Spies, Fleischhack, and Huwig [1,2]. For every constant $c \geq 1$, Ambos-Spies, Neis, and Terwijn [3] have recently shown that every language that is n^{c+3}-random over the uniform probability measure is n^c-generic. It is shown that, in fact, every language that is n^{c+3}-random over *any* strongly positive n^c-time computable probability measure is n^c-generic. Roughly speaking, this implies that, when genericity is used to prove a measure result, the result is not specific to the underlying probability measure.

A full paper will be available soon.

References

[1] K. Ambos-Spies, H. Fleischhack, and H. Huwig. Diagonalizing over polynomial time computable sets. *Theoretical Computer Science*, 51:177–204, 1987.

[2] K. Ambos-Spies, H. Fleischhack, and H. Huwig. Diagonalizing over deterministic polynomial time. In *Proceedings of Computer Science Logic '87*, pages 1–16. Springer-Verlag, 1988.

[3] K. Ambos-Spies, C. Neis, and S. A. Terwijn. Genericity and measure for exponential time. *Theoretical Computer Science*. To appear.

The Isomorphism Problem for One-Time-Only Branching Programs

Thomas Thierauf, Abteilung Theoretische Informatik, Universität Ulm, Oberer Eselsberg, 89069 Ulm, Germany. (thierauf@informatik.uni-ulm.de)

Abstract Number 96-21

We investigate the computational complexity of the *isomorphism problem for one-time-only branching programs (BP1-Iso)*: on input of two one-time-only branching programs B_0 and B_1, decide whether there exists a permutation of the variables of B_1 such that it becomes equivalent to B_0.

Our main result is a two-round interactive proof for the complement of BP1-Iso. The protocol is based on the Schwartz-Zippel Theorem to probabilistically check polynomial idendities. As a consequence, BP1-Iso cannot be NP hard unless the polynomial hierarchy collapses.

We extend the protocol to get an interactive proof to decide the non-isomorphism of multivariate polynomials over an arbitrary field.

Finally, we show that BP1-Iso has a zero-knowledge interactive proof.

A full paper will be available shortly. Send email to the author.

Randomness and Complexity (Ph.D. Thesis)

Yongge Wang, Mathematisches Institut, Universität Heidelberg, Im Neuenheimer Feld 294, 69120 Heidelberg, GERMANY, (wang@math.uni-heidelberg.de)

Abstract Number 96-22

The topic of this thesis is the study of randomness concepts and their applications in computational complexity theory. In Chapter 3, we discuss the classical notions of randomness. We give a systematic study of various notions of randomness, especially, of the following concepts defined in terms of typicalness: Martin-Löf randomness, Lutz randomness, Schnorr randomness, Ko randomness, and Kurtz randomness. We study each notion of typicalness by using three different approaches: the approach based on constructive null covers, the approach based on martingales and the approach based on Solovay style criteria.

Schnorr has shown that Martin-Löf randomness is a proper refinement of Lutz randomness, but his proof was not correct, and he left open the question whether Lutz randomness is a proper refinement of Schnorr randomness. In the sequel, the later was conjectured to be true by van Lambalgen and Lutz. We prove this conjecture and we correct the proof of Schnorr's result, thereby completely clarifying the relations among the above cited important randomness concepts. At the same time, we will show that there is a Schnorr random sequence which is not Church stochastic.

In Chapter 5, we give a survey of notions of resource bounded randomness and show their relations to each other. Moreover, we introduce several new notions of resource bounded randomness corresponding to the classical notions of randomness discussed in Chapter 3. We show that, for the polynomial time bound, the notion of Ko randomness is independent of the notions of Lutz, Schnorr and Kurtz randomness. Lutz has conjectured that, for a given time or space bound, the corresponding resource bounded Lutz randomness is a proper refinement of resource bounded Schnorr randomness. We answer this conjecture affirmatively. In contrast to this result, however, we also show that the notions of polynomial time bounded Lutz, Schnorr and Kurtz randomness coincide in the case of recursive sets, whence it suffices to study the notion of resource bounded Lutz randomness in the context of complexity theory.

The stochastic properties of resource bounded random sequences (i.e., resource bounded typical sequences) will be discussed in detail. We show that the law of the iterated logarithm holds for p-random sequences. Hence almost all sets in the exponential time complexity class are "hard" from the viewpoint of statistics. This law also gives a quantitative characterization of the density of p-random sets. And, when combined with an invariance property of p-random sets, these laws are useful in proving that some classes of sets have p-measure 0.

Polynomial time safe and unsafe approximations for intractable sets were introduced by Meyer, Paterson, Yesha, Duris, Rolim and Ambos-Spies, respectively. The question of which sets have optimal safe and unsafe approximations has been investigated extensively. Using the law of the iterated logarithm for p-random sequences discussed in Chapter 5, we show that both the class of Δ-levelable sets and the class of sets which have optimal polynomial time unsafe approximations have p-measure 0. Hence p-random sets do not have optimal polynomial time unsafe approximations.

In the last chapter, we show that no **P**-selective set is $\leq_{tt}^{p}$-hard for **NP** unless **NP** is small.

The full thesis is available by email to wang@math.uni-heidelberg.de

Genericity, Randomness and Polynomial Time Approximations

Yongge Wang, Mathematisches Institut, Universität Heidelberg, Im Neuenheimer Feld 294, 69120 Heidelberg, GERMANY, (wang@math.uni-heidelberg.de)

Abstract Number 96-23

Polynomial time safe and unsafe approximations for intractable sets were introduced by Meyer and Paterson and Yesha respectively. The question of which sets have optimal safe and unsafe approximations has been investigated extensively. Recently, Duris and Rolim and Ambos-Spies showed that the existence of optimal polynomial time approximations for the safe and unsafe cases is independent. Using the law of the iterated logarithm for p-random sequences (which has recently been proved by Wang), we extend this observation by showing that both the class of polynomial time Δ-levelable sets and the class of sets which have optimal polynomial time unsafe approximations have p-measure 0. Hence typical sets in E (in the sense of p-measure) do not have optimal polynomial time unsafe approximations. We will also establish the relations between resource bounded genericity concepts and the polynomial time safe and unsafe approximation concepts.

A full paper is available by email to wang@math.uni-heidelberg.de

Resource Bounded Randomness and Computational Complexity

Yongge Wang, Mathematisches Institut, Universität Heidelberg, Im Neuenheimer Feld 294, 69120 Heidelberg, GERMANY, (wang@math.uni-heidelberg.de)

Abstract Number 96-24

We give a survey of resource bounded randomness concepts and show their relations to each other. Moreover, we introduce several new resource bounded randomness concepts corresponding to the classical randomness concepts. We show that the notion of polynomial time bounded Ko randomness is independent of the notions of polynomial time bounded Lutz, Schnorr and Kurtz randomness. Lutz has conjectured that, for a given time or space bound, the corresponding resource bounded Lutz randomness is a proper refinement of resource bounded Schnorr randomness. We answer this conjecture affirmatively. Moreover, we show that resource bounded Schnorr randomness is a proper refinement of resource bounded Kurtz randomness too. In contrast to this result, however, we also show that the notions of polynomial time bounded Lutz, Schnorr and Kurtz randomness coincide in the case of recursive sets, whence it suffices to study the notion of resource bounded Lutz randomness in the context of complexity theory. The stochastic properties of resource bounded random sequences will be discussed in detail. Schnorr has already shown that the law of large numbers holds for p-random sequences. We show that another important law in probability theory, the law of the iterated logarithm, holds for p-random sequences too. Hence almost all sets in the exponential time complexity class are "hard" from the viewpoint of statistics.

A full paper is available by email to wang@math.uni-heidelberg.de