COMPLEXITY ABSTRACTS 2001. Vol XI

Abstract

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

Improved Bounds on the Weak Pigeonhole Principle

On the (Im)possibility of Obfuscating Programs

Enumerations of the Kolmogorov Function

Are Cook and Karp every the same?

Tighter Constant-Factor Time Hierarchies

Element Distinctness on 1-Tape TM's: A Complete Solution

When Plans Distinguish Bayes Nets

Lower Bounds for Linear Locally Decodable Codes and PIR

On Interactive Proofs with a Laconic Prover

Relating Partial and Complete Solutions and the Complexity of Computing Smallest Solutions

P-Immune Sets with Holes Lack Self-Reducibility Properties

The Complexity of Computing the Size of an Interval

If P$\neq$NP then Some Strongly Noninvertible Functions are Invertible

The Complexity of Computing the of Self-Avoiding Walks

On a p-optimal proof system for SAT

Improved Bounds on the Weak Pigeonhole Principle and Infinitely Many Primes from Weaker Axioms

Albert Atserias, Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, c/ Jordi Girona Salgado 1-3, C5, 08034 Barcelona, SPAIN, (atserias@lsi.upc.es)

Abstract 01-1

We show that the known bounded-depth proofs of the Weak Pigeonhole Principle PHP_n^{2n} in size $n^{O(\log(n))}$ are not optimal in terms of size. More precisely, we give a size-depth trade-off upper bound: there are proofs of size $n^{O(d(\log(n))^{2/d})}$ and depth $O(d)$. This solves an open problem of Maciel, Pitassi and Woods (2000). Our technique requires formalizing the ideas underlying Nepomnjaščij's Theorem which might be of independent interest. Moreover, our result implies a proof of the unboundedness of primes in $I\Delta_0$ with a provably weaker 'large number assumption' than previously needed.

These results will be presented at MFCS'2001. A full paper is available at http://www.lsi.upc.es/ atserias.

On the (Im)possibility of Obfuscating Programs

Boaz Barak, Department of Computer Science, Weizmann Institute of Science, Rehovot, ISRAEL. (boaz@wisdom.weizmann.ac.il)
Oded Goldreich, Department of Computer Science, Weizmann Institute of Science, Rehovot, ISRAEL. (oded@wisdom.weizmann.ac.il)
Russell Impagliazzo, Department of Computer Science and Engineering, University of California, San Diego, La Jolla, CA 92093-0114, USA. (russell@cs.ucsd.edu)
Steven Rudich, Computer Science Department, Carnegie Mellon University, 5000 Forbes Ave. Pittsburgh, PA 15213, USA. (rudich@cs.cmu.edu)
Amit Sahai, Department of Computer Science 35 Olden St. Princeton, NJ 08540, USA. (sahai@cs.princeton.edu)
Salil Vadhan, Division of Engineering and Applied Sciences, Harvard University, 33 Oxford Street, Cambridge, MA 02138, USA. (salil@deas.harvard.edu)
Ke Yang, Computer Science Department, Carnegie Mellon University, 5000 Forbes Ave. Pittsburgh, PA 15213, USA. (yangke@cmu.edu)

Abstract 01-2

Informally, an *obfuscator* $\mathcal{O}$ is an (efficient, probabilistic) "compiler" that takes as input a program (or circuit) P and produces a new program $\mathcal{O}(P)$ that has the same functionality as P yet is "unintelligible" in some sense. Obfuscators, if they exist, would have a wide variety of cryptographic and complexity-theoretic applications.
In this work, we initiate a theoretical investigation of obfuscation. Our main result is that, even under very weak formalizations of the above intuition, obfuscation is impossible. We prove this by constructing a family of functions $\mathcal{F}$ that are *inherently unobfuscatable* in the following sense: there is a property $\pi : \mathcal{F} \to \{0,1\}$ such that (a) given *any* program that computes a function $f \in \mathcal{F}$, the value $\pi(f)$ can be efficiently computed, yet (b) given oracle access to a (randomly selected) function $f \in \mathcal{F}$, no efficient algorithm can compute $\pi(f)$ much better than random guessing.
We extend our impossibility result in a number of ways, including even obfuscators that (a) are not necessarily computable in polynomial time, (b) only *approximately* preserve the functionality, and (c) only need to work for very restricted models of computation ($\mathbf{TC_0}$). We also rule out several potential applications of obfuscators, by constructing "unobfuscatable" signature schemes, encryption schemes, and pseudorandom function families.

An extended abstract will appear in CRYPTO 2001. A full paper is available by email to boaz@wisdom.weizmann.ac.il

Enumerations of the Kolmogorov Function

Richard Beigel, Temple University
Harry Buhrman, CWI
Lance Fortnow, NEC Research Institute
Luc Longpré, University of Texas at El Paso
Leen Torenvliet, University of Amsterdam

Abstract 01-3

We consider the hardness of enumerating k possible values for the Kolmogorov complexity function $C(x)$ so that one of them is correct. We show several results including

- Any computable enumerator for $C(x)$ must enumerate $\Omega(n)$ possibilities.
- If a k-enumerator (fixed k) for C is reducible to an r.e. set A then A is Turing-equivalent to the halting problem.
- Every nonrecursive set is not weak-truth-table reducible to some 2-enumerator for $C(x)$ or any other recursively-bounded function.
- The time-bounded enumeration question gives a new characterization of the class S_2^p defined by Russell and Sundaram.
- Enumerating $O(\log n)$ values of the space-bounded Kolmogorov function remains hard for PSPACE.

A preliminary paper is available at
http://www.neci.nj.nec.com/homepages/fortnow/papers.

Are Cook and Karp Ever the Same?

Richard Beigel, Temple University
Lance Fortnow, NEC Research Institute

Abstract 01-4

We consider the question as to whether there exists a set A such that every set polynomial-time Turing equivalent to A is also many-one equivalent to A. We show that if E = NE then no sparse set has this property. We give the first relativized world where there exists a set with this property, and in this world the set A is sparse.

A preliminary paper is available at
http://www.neci.nj.nec.com/homepages/fortnow/papers.

Tighter Constant-Factor Time Hierarchies

Amir M. Ben-Amram,
The Academic College of Tel-Aviv Yaffo, 4 Antokolski Str., 64044 Tel-Aviv, Israel (amirben@mta.ac.il)

Abstract 01-5

For certain computational models, a constant-factor time hierarchy theorem is known, showing that multiplying a time bound by a constant can make a difference in problem-solving power (unlike the situation with Turing machines). My favourite examples are the random access machine and the programming-language model of Neil Jones (see STOC 1993 and later papers); these models have a flavour of being more "realistic" than the Turing machine and this agrees with the idea, that in real life constant factors do matter.

A typical hierarchy theorem, as has been proved for the above models, follows:
Theorem. For every time-constructible function $T(n) \geq n$, there is a constant b such that $\mathrm{TIME}(b \cdot T(n)) \setminus \mathrm{TIME}(T(n))$ is not empty.
We improve the result by means of the "translational technique" (padding), known from work on Turing-machine time hierarchies. For polynomially-bounded time bounds $T(n) \gg n$, we show a *multiplicatively dense hierarchy*: $\mathrm{TIME}(b \cdot T(n)) \setminus \mathrm{TIME}(T(n))$ is not empty for *any* $b > 1$. For linear time bounds we have a slightly weaker result: there is a constant Δ such that for every $a \geq 1$, $\mathrm{TIME}((a + \Delta)n) \setminus \mathrm{TIME}(an)$ is not empty.

A full paper is available by email to amirben@mta.ac.il

Element Distinctness on One-Tape Turing Machines: A Complete Solution

Amir M. Ben-Amram,
The Academic College of Tel-Aviv-Yaffo, 4 Antokolski St., 64044 Tel-Aviv, Israel (amirben@mta.ac.il)
Omer Berkman,
The Academic College of Tel-Aviv-Yaffo, 4 Antokolski St., 64044 Tel-Aviv, Israel (omer58@netvision.net.il)
Holger Petersen,
University of Stuttgart, Breitwiesenstraße 20–22, D-70565 Stuttgart, Germany (petersen@informatik.uni-stuttgart.de)

Abstract 01-6

We give a complete characterization of the complexity of the element distinctness problem for n elements of $m \geq \log n$ bits each on deterministic and nondeterministic one-tape Turing machines. We present an algorithm running in time $O(n^2 m(m + 2 - \log n))$ for deterministic machines, and nondeterministic solutions of time complexity $O(nm(n+\log m))$. For elements of logarithmic size $m = O(\log n)$, on nondeterministic machines, these results close the gap between the known lower bound $\Omega(n^2 \log n)$ and the previously best upper bound $O(n^2(\log n)^{3/2}(\log\log n)^{1/2})$. We prove additional lower bounds to show that our upper bounds are optimal for all other possible relations between m and n. The upper bounds employ hashing techniques, while the lower bounds make use of the communication complexity of set disjointness.

A full paper is available by email to petersen@informatik.uni-stuttgart.de.

When Plans Distinguish Bayes Nets

Alex Dekhtyar and Judy Goldsmith, Department of Computer Science, University of Kentucky, 773 Anderson Hall, Lexington, KY 40502, ({dekhtyar,goldsmit}@cs.uky.edu)
Janice Pearce, Department of Mathematics and Computer Science, Berea College, Berea, KY, (pearce@berea.edu)

Abstract 01-7

Bayes nets model stochastic processes such as medical systems, industrial systems such as nuclear power plants, military scenarios, and academic advising systems. They can be used for probabilistic inference or decision-theoretic planning.

In order to model a system, the system states must be factored into parameters/variables/nodes, so that each node depends on some (preferably small) subset of other nodes. These dependencies are described by a directed, acyclic graph. They are then fully specified by conditional probability tables for each node with in-degree > 0.

Decision-theoretic planning is the process of choosing actions in order to maximize the probability of the system achieving one of pre-defined desirable states. Each action determines its own conditional probability tables for the system. Thus, the choice of an action at each stage determines the probabilistic evolution of the system to the next stage.

A *plan* is a mapping from system states to actions; if the system states are not necessarily observable, then it is a mapping from beliefs about the current state of the system (probability distributions over the states) to actions. A *planning algorithm* is a function that, given a Bayes net, produces a plan.

Information may be hard to come by. In some situations, such as medical applications, there may be a plethora of data, though different data sets may yield different models. In others, such as military modeling, there are situations that should not be tested. In those cases, the modelers must depend on expert opinions — which are likely to differ.

Thus, the problem of data fusion or reconciliation becomes a central one for modeling these systems. One solution is to determine that the differences are too minor to matter.

We ask when two sets of conditional probability tables for the same Bayes net structure are *equivalent* relative to a planning algorithm, under the following notions of equivalence: (i) the algorithm produces the same plan for each version of the Bayes net, and (ii) the algorithm's plans have the same probability of success for each version of the Bayes net. We show that the first problem is coNP-complete, and the second is coNP^{PP}-hard. We also give restrictions on either the type of plan considered or the structure of the Bayes net that make both problems easy.

A full paper is available at http://www.cs.uky.edu/~dekhtyar/dblab/robust.ps The paper is listed as University of Kentucky Technical Report TR 325-01.

Lower Bounds for Linear Locally Decodable Codes and PIR

Oded Goldreich, Weizmann Institute, Rehovot, ISRAEL. (oded@wisdom.weizmann.ac.il)
Howard Karloff, AT&T Labs–Research, USA, (howard@research.att.com)
Leonard Schulman, Caltech, USA. (schulman@cs.caltech.edu)
Luca Trevisan, Computer Science Division, UC-Berkeley, USA. (luca@eecs.berkeley.edu)

Abstract 01-8

We prove that if a linear error correcting code $C : \{0,1\}^n \to \{0,1\}^m$ is such that a bit of the message can be probabilistically reconstructed by looking at two entries of a corrupted codeword, then $m = 2^{\Omega(n)}$. We also present several extensions of this result.

We show a reduction from the complexity of one-round information-theoretic Private Information Retrieval Systems (with two servers) to Locally Decodable Codes, and conclude that if all the servers' answers are linear combinations of the database content, then $q = \Omega(n/2^a)$, where q is the length of the user's query and a is the length of the servers' answers. Actually, 2^a can be replaced by $O(a^k)$, where k is the number of bits in the answer that are actually used in the reconstruction.

A full version is available from the authors.

On Interactive Proofs with a Laconic Prover

Oded Goldreich, Weizmann Institute, Rehovot, ISRAEL. (oded@wisdom.weizmann.ac.il)
Salil Vadhan, Harvard University, Cambridge, MA, USA. (salil@deas.harvard.edu)
Avi Wigderson, Institute for Advanced Study, Princeton, NJ, USA. (avi@ias.edu)

Abstract 01-9

We continue the investigation of interactive proofs with bounded communication, as initiated by Goldreich and Håstad (IPL 1998). Let L be a language that has an interactive proof in which the prover sends few (say m) bits to the verifier. We prove that the complement $\bar{L}$ has a *constant-round* interactive proof of complexity that depends only exponentially on m. This provides the first evidence that for NP-complete languages, we cannot expect interactive provers to be much more "laconic" than the standard NP-proof.
When the proof system is further restricted (e.g., when $m = 1$, or when we have perfect completeness), we get significantly better upper bounds on the complexity of $\bar{L}$.

An extended abstract will appear in the proceedings of *ICALP'01*. A full version is available from the authors.

Relating Partial and Complete Solutions and the Complexity of Computing Smallest Solutions

André Große, Institut für Informatik, Friedrich-Schiller-Universität Jena, 07740 Jena, Germany. Email: grosse@informatik.uni-jena.de.
Jörg Rothe, Abteilung für Informatik, Heinrich-Heine-Universität Düsseldorf, 40225 Düsseldorf, Germany. Email: rothe@cs.uni-duesseldorf.de.
Gerd Wechsung, Institut für Informatik, Friedrich-Schiller-Universität Jena, 07740 Jena, Germany. Email: wechsung@informatik.uni-jena.de.

Abstract 01-10

We prove that computing a single pair of vertices that are mapped onto each other by an isomorphism ϕ between two isomorphic graphs is as hard as computing ϕ itself. This result optimally improves upon a result of Gál et al. [1]. We establish a similar, albeit slightly weaker, result about computing complete Hamiltonian cycles of a graph from partial Hamiltonian cycles. We also show that computing the lexicographically first four-coloring for planar graphs is Δ_2^p-hard. This result optimally improves upon a result of Khuller and Vazirani who prove this problem to be NP-hard, and conclude that it is not self-reducible in the sense of Schnorr [2], assuming P $\neq$ NP. We discuss this application to non-self-reducibility and provide a general related result.

References

[1] A. Gál, S. Halevi, R. Lipton, and E. Petrank. Computing from partial solutions. In *Proceedings of the 14th Annual IEEE Conference on Computational Complexity*, pages 34–45. IEEE Computer Society Press, May 1999.

[2] C. Schnorr. Optimal algorithms for self-reducible problems. In S. Michaelson and R. Milner, editors, *Proceedings of the 3rd International Colloquium on Automata, Languages, and Programming*, pages 322–337, University of Edinburgh, July 1976. Edinburgh University Press.

A preliminary version of this paper appears in ICTCS 2001. A full paper is available by email to the authors.

P-Immune Sets with Holes Lack Self-Reducibility Properties

Lane A. Hemaspaandra, Department of Computer Science, University of Rochester, Rochester, NY 14627, USA. Email: lane@cs.rochester.edu.
Harald Hempel, Institut für Informatik, Friedrich-Schiller-Universität Jena, 07743 Jena, Germany. Email: hempel@informatik.uni-jena.de.

Abstract 01-11

We show that no P-immune set having exponential gaps is positive-Turing self-reducible (or even locally left-positive-Turing word-decreasing-self-reducible).

A preliminary version of this paper will appear in DMTCS 2001. The most current version is available via email to the authors.

The Complexity of Computing the Size of an Interval

Lane A. Hemaspaandra, Department of Computer Science, University of Rochester, Rochester, NY 14627, USA, (lane@cs.rochester.edu)
Christopher Homan, Department of Computer Science, University of Rochester, Rochester, NY 14627, USA, (choman@cs.rochester.edu)
Sven Kosub, Theoretische Informatik, Julius-Maximilians-Universität Würzburg, Am Hubland, D-97074 Würzburg, Germany, (kosub@informatik.uni-wuerzburg.de)
Klaus
W. Wagner, Theoretische Informatik, Julius-Maximilians-Universität Würzburg, Am Hubland, D-97074 Würzburg, Germany, (wagner@informatik.uni-wuerzburg.de)

Abstract 01-12

We study the complexity of counting the number of elements in intervals of feasible partial orders. Depending on the properties that partial orders may have, such counting functions have different complexities. If we consider total, polynomial-time decidable orders then we obtain exactly the #P functions. We show that the interval size functions for polynomial-time adjacency checkable orders are tightly related to the class FPSPACE(poly): Every FPSPACE(poly) function equals a polynomial-time function subtracted from such an interval size function. We study the function #DIV that counts the nontrivial divisors of natural numbers, and we show that #DIV is the interval size function of a polynomial-time decidable partial order with polynomial-time adjacency checks if and only if primality is in polynomial time.

A precursor of this paper appears in ICALP 2001. The full paper will be available soon via email to the authors.

If P $\neq$ NP then Some Strongly Noninvertible Functions are Invertible

Lane A. Hemaspaandra, Department of Computer Science, University of Rochester, Rochester, NY 14627, USA. Email: lane@cs.rochester.edu.
Kari Pasanen, University of Jyväskylä and Nokia Networks, Jyväskylä, Finland. Email: kari.pasanen@nokia.com.
Jörg Rothe, Abteilung für Informatik, Heinrich-Heine-Universität Düsseldorf, 40225 Düsseldorf, Germany. Email: rothe@cs.uni-duesseldorf.de.

Abstract 01-13

Rabi, Rivest, and Sherman alter the standard notion of noninvertibility to a new notion they call strong noninvertibility, and show—via explicit cryptographic protocols for secret-key agreement ([RS93, RS97] attribute this protocol to Rivest and Sherman) and digital signatures [RS93, RS97]—that strongly noninvertible functions are very useful components in protocol design. Their definition of strong noninvertibility has a small twist ("respecting the argument given") that is needed to ensure cryptographic usefulness. In this paper, we show that this small twist has a consequence: Unless P = NP, some strongly noninvertible functions are invertible.

References

[RS93] M. Rabi and A. Sherman. Associative one-way functions: A new paradigm for secret-key agreement and digital signatures. Technical Report CS-TR-3183/UMIACS-TR-93-124, Department of Computer Science, University of Maryland, College Park, Maryland, 1993.

[RS97] M. Rabi and A. Sherman. An observation on associative one-way functions in complexity theory. *Information Processing Letters*, 64(2):239–244, 1997.

A preliminary version of this paper appears in FCT 2001. A full paper is available by email to the authors.

The Complexity of Computing the of Self-Avoiding Walks in Two-Dimensional Grid Graphs and in Hypercube Graphs

Mitsunori Ogihara, Department of Computer Science, University of Rochester, Rochester, NY 14627-0225, USA (ogihara@cs.rochester.edu)
Seinosuke Toda, Department of Applied Mathematics, Nihon University, 3-25-40 Sakurajyou-shi, Setagaya-ku, Tokyo 156, Japan (*toda@am.chs.nihon-u.ac.jp*)

Abstract 01-14

Valiant (*SIAM Journal on Computing* 8, pages 410–421) showed that the problem of counting the number of *s-t* paths in graphs (both in the case of directed graphs and in the case of undirected graphs) is complete for #P under polynomial-time one-Turing reductions (namely, some post-computation is needed to recover the value of a #P-function). Valiant then asked whether the problem of counting the number of self-avoiding walks of length n in the two-dimensional grid is complete for $\#P_1$, i.e., the tally-version of #P. This paper offers a partial answer to the question. It is shown that a number of versions of the problem of computing the number of self-avoiding walks in two-dimensional grid graphs (graphs embedded in the two-dimensional grid) is polynomial-time one-Turing complete for #P.
This paper also studies the problem of counting the number of self-avoiding walks in graphs embedded in a hypercube. It is shown that a number of versions of the problem is polynomial-time one-Turing complete for #P, where a hypercube graph is specified by its dimension, a list of its nodes, and a list of its edges. By scaling up the completeness result for #P, it is shown that the same variety of problems is polynomial-time one-Turing complete for #EXP, where the post-computation required is right bit-shift by exponentially many bits and a hypercube graph is specified by: its dimension, a boolean circuit that accept its nodes, and one that accepts its edges.

An extended abstract is available by email to ogihara@cs.rochester.edu.

On a p-optimal proof system for SAT

Zenon Sadowski, Institute of Mathematics, University of Białystok, 15-267 Białystok, ul. Akademicka 2, POLAND, (sadowski@math.uwb.edu.pl)

Abstract 01-15

The question of the existence of a p-optimal proof system for SAT (the set of all satisfiable boolean formulas) was posed by J. Köbler and J. Messner. This question like the similar and older problem of the existence of a p-optimal proof system for $TAUT$ is still open.

A proof system for a language L is a polynomial time computable function whose range is L. This notion was introduced by S. Cook and R. Reckhow. Intuitively a proof system h p-simulates a second one g if there is a polynomial time computable function t translating proofs in g into proofs in h. A proof system is called p-optimal for L when it p-simulates any proof system for L.

It is not currently known whether UP and other promise classes have complete languages. J. Hartmanis and L. Hemachandra pointed out that UP possesses complete languages if and only if there is a recursive enumeration of polynomial time clocked Turing machines covering all languages from this class. Earlier, the question of whether NP $\cap$ co-NP possesses complete languages was related to an analogous statement.

In this paper we show that the problem of the existence of a p-optimal proof system for SAT can be characterized in a similar manner. Namely, there exists a p-optimal proof system for SAT if and only if there is a recursive enumeration of polynomial time clocked Turing machines covering all easy (polynomial time recognizable) subsets of SAT (a suitable recursive presentation of the class of all easy subsets of SAT).

Using this characterization we prove that if there does not exist a p-optimal proof system for SAT, then for every theory T there exists an easy subset of SAT which is not T-provably easy.

A full paper is available by email to sadowski@math.uwb.edu.pl