Mathematisches Forschungsinstitut Oberwolfach

Report No. 31/2007

Complexity Theory

Organised by
Joachim von zur Gathen, Universität Bonn
Oded Goldreich, Weizmann Institute, Rehovot
Madhu Sudan, MIT, Cambridge

June 24th – June 30th, 2007

Abstract. Computational Complexity Theory is the mathematical study of the intrinsic power and limitations of computational resources like time, space, or randomness. The current workshop focused on recent developments. Connections to the theory of error-correcting codes played a central role in many of these developments.

Mathematics Subject Classification (2000): 68-06, 68Q01, 68Q10, 68Q15, 68Q17, 68Q25, 94B05, 94B35.

Introduction by the Organisers

The workshop *Complexity Theory* was organized by Joachim von zur Gathen (Universität Bonn), Oded Goldreich (Weizmann Institute), and Madhu Sudan (MIT). The workshop was held on June 24th–30th 2007, and attended by approximately 50 participants spanning a wide range of interests within the field of Computational Complexity. The plenary program, attended by all participants, featured eight long lectures as well as short (10-minute) reports by almost all participants. In addition, extensive interaction took place in smaller groups.

The Oberwolfach Meeting on Complexity Theory is marked by a long tradition and a continuous transformation. Originally starting with a focus on algebraic and Boolean complexity, the meeting has continuously evolved to cover a wide variety of areas, most of which were not even in existence at the time of the first meeting (in 1972). While inviting many of the most prominent researchers in the field, the organizers try to identify and invite a fair number of promising young researchers.

Computational complexity (a.k.a. complexity theory) is a central field of computer science with a remarkable list of celebrated achievements as well as a vibrant research activity. The field is concerned with the study of the *intrinsic complexity* of computational tasks, and this study tends to *aim at generality*: it focuses on natural computational resources, and considers the effect of limiting these resources on the class of problems that can be solved. Computational complexity is related to and has substantial interaction with other areas of mathematics such as number theory, algebra, combinatorics, coding theory, and optimization.

The workshop focused on several sub-areas of complexity theory and its nature may be best illustrated by a brief survey of some of the meeting's highlights.

Connections to the Theory of Error-Correcting Codes. The interplay between coding theory and complexity theory first emerged in the context of "hardness amplification" (almost two decades ago) and other connections are less than a decade old (e.g., the connection to probabilistic checking of proofs and extraction of pure randomness). Several applications of the known connections were presented in the current meeting, and in addition a new connection to algebraic complexity was presented.

While previous applications of the aforementioned connections went in the direction of coding theory to complexity theory, a recent result reported by Venkat Guruswami goes in the opposite direction. This work, by Guruswami and his graduate student (Rudra), resolves a decades-old central problem in coding theory by presenting an explicit error-correcting code of constant-size alphabet that approaches the capacity bound (under worst-case errors, using list decoding).

Extracting randomness. Extracting almost-perfect randomness from weak sources of (imperfect) randomness is crucial for the actual use of randomized procedures. Typical analyses of randomized procedures assume that the procedures have access to a perfect random source. However, in reality one only has access to sources of weak randomness (e.g., having constant entropy rate). Indeed, the problem has attracted a lot of attention in the last couple of decades.

In the meeting, Chris Umans has presented recent work with Guruswami and Vadhan, which utilizes recent algebraic and coding theoretic techniques to the construction of (single-source) randomness extractors. This construction meets (and actually improves) the best known parameters for the problem (which are almost optimal), but does so by a relatively simple construction rather than by a complex combination of numerous constructs (as done in prior work). Furthermore, the new work introduces improved constructions for an intermediate primitive (called randomness condenser), which is of independent interest.

While single-source randomness extractors must utilize an auxiliary random seed (which may be very short), some applications do not allow for such a seed. In this case, extraction from several (e.g., two) independent sources of weak randomness is called for. An important step in the study of this direction was made by Anup Rao, and presented by him in the meeting.

Algebraic complexity and modular polynomial composition. An important task in algebraic computation is modular polynomial composition; that is, given three univariate polynomials f, g and h, one is required to obtain the coefficients of the polynomial $f \circ g \bmod h$. This task has many applications, most notably as an ingredient in algorithms for polynomial factorization. The previously best algorithm was presented 30 years ago and uses $O(n^{1.7})$ arithmetic operations, where n denotes the maximum degree of the polynomials.

In the meeting, Chris Umans presented significant progress on this celebrated open problem in the form of an almost linear-time algorithm that works for fields of small characteristic. This major progress on a purely algebraic problem is essentially based on methods that were introduced into coding theory by Guruswami and Rudra, and then applied to complexity theory in the context of randomness extractors (see foregoing paragraphs). All three results, which are major achievements in their respective areas, were presented at the meeting.

Cryptography and Zero-Knowledge. Zero-knowledge proofs are fascinating concepts and extremely useful constructs. Their fascinating nature is due to their seemingly contradictory definition that mandates that they be convincing and yet yield nothing beyond the validity of the assertion being proved. Their applicability in the domain of cryptography is vast; they are typically used to force malicious parties to behave according to a predetermined protocol. In addition to their direct applicability in cryptography, zero-knowledge proofs serve as a good benchmark for the study of various problems regarding cryptographic protocols. Zero-knowledge proofs come in many flavors, and it is of great theoretical and practical importance to investigate the relationship among them.

A central problem in this area, which has been open since 1986, refers to the gap between the known results regarding two dual notions: the notion of general zero-knowledge proofs (in which the secrecy condition holds with respect to feasible adversaries) and the notion of statistical zero-knowledge arguments (in which the soundness condition holds with respect to feasible adversaries). This gap was bridged in a recent work of Salil Vadhan, jointly with his graduate students (Nguyen and Ong), and was presented by Vadhan in this meeting.

A problem related to both cryptography and coding theory is the problem of constructing private information retrieval schemes and/or locally decodable codes. In the context of error-correcting codes, such schemes should allow the recovery of any bit in the original message based on a constant number (e.g., three) probes to the corrupted codeword. For more than a decade it was believed that the length of such codewords must be (weakly) exponential in the length of the message. In the meeting, Sergey Yekhanin (PhD student) presented his recent result that refutes this belief.

Delegating your work to an untrusted entities. Needless to say, it is nice to delegate your work to others, but what if you don't trust the others? The very definition of a proof system refers to such a possibility – the hard task of finding a proof is delegated to the outside while you make sure that the proof is valid by performing the easier task of verification. However, facilitating verification may mean making the task of finding adequate proofs even harder. In the context of *program checking* this phenomenon is explicitly disallowed: wishing to solve some problem you may use an untrusted program that supposedly solves this problem (but not a program that solve more complex problems). Needless to say, the aim is allowing the delegator, called a checker, to use significantly fewer resources than any program that correctly solves the problem. In the meeting, Shafi Goldwasser presented a recent result that achieves this goal for a natural complexity measure (circuit depth) and for a wide class of problems (i.e., NC).

A characterization of testable graph properties. The area of *property testing* is concerned with promise problems that call for distinguishing those objects that have a predetermined property from objects that are "far" from any object having this property. The focus is on sub-linear time algorithms that probe the given object at few (randomly selected) locations. In some cases, one may perform the task by using a number of probes that only depends on the proximity parameter (and is independent of the size of the object). In the meeting, Noga Alon presented a recent result that characterizes the class of graph properties (where graphs are represented by their adjacency matrices) for which such a phenomenon holds.

The rest of this report. This report contains extended abstracts of the eight plenary lectures as well as abstracts of forty short reports.

Workshop: Complexity Theory

Table of Contents

Abstracts

Characterizing the testable graph properties via the regularity lemma

Noga Alon
(joint work with Eldar Fischer, Ilan Newman, Asaf Shapira)

A graph G on n vertices is *ϵ-far* from satisfying a property P, if one needs to add and/or delete at least ϵn^2 edges to G in order to turn it into a graph satisfying P. A tester for P should distinguish with high probability, say $2/3$, between the case that G satisfies P and the case that G is ϵ-far from satisfying P. Here we assume that the tester can query some oracle whether a pair of vertices, i and j, are adjacent in the input graph G.

Definition 1 (Testable)**.** *A graph property P is* testable *if there is a randomized algorithm T, that can distinguish for every $\epsilon > 0$ and with probability $2/3$, between graphs satisfying P and graphs that are ϵ-far from satisfying P, while making a number of edge queries which is bounded by some function $q(\epsilon)$ that is independent of the size of the input.*

This notion was introduced by Goldreich, Goldwasser and Ron [2] and received a considerable amount of attention by various researchers. The problem of characterizing the testable graph properties was naturally one of the main open problems in the study of the area and was raised already in this paper and mentioned in several subsequent ones. Here we give such a characterization. It is based on the regularity lemma of Szemerédi [3].

For every two nonempty disjoint vertex sets A and B of a graph G, we define $e(A,B)$ to be the number of edges of G between A and B. The *edge density* of the pair is defined by $d(A,B) = e(A,B)/(|A||B|)$.

Definition 2 (γ-regular pair)**.** *A pair (A,B) is* γ-regular*, if for any two subsets $A' \subseteq A$ and $B' \subseteq B$, satisfying $|A'| \geq \gamma|A|$ and $|B'| \geq \gamma|B|$, the inequality $|d(A',B') - d(A,B)| \leq \gamma$ holds.*

Definition 3 (γ-regular equipartition)**.** *An equipartition $B = \{V_i \mid 1 \leq i \leq k\}$ of the vertex set of a graph is called* γ-regular *if all but at most $\gamma\binom{k}{2}$ of the pairs (V_i, V_j) are γ-regular.*

An equipartition is said to *refine* another if every set of the former is contained in one of the sets of the latter. Szemerédi's regularity lemma can be formulated as follows.

Lemma 4 ([3])**.** *For every m and $\gamma > 0$ there exists $T = T_4(m,\gamma)$ with the following property: If G is a graph with $n \geq T$ vertices, and A is any equipartition of the vertex set of G of order at most m, then there exists a refinement B of A of order k, where $m \leq k \leq T$ and B is γ-regular. In particular, for every m and $\gamma > 0$ there exists $T = T_4(m,\gamma)$, such that any graph with $n \geq T$ vertices has a γ-regular equipartition of order k, where $m \leq k \leq T$.*

It seems natural to define a graph property, which states that a graph has a given γ-regular partition, that is, an equipartition which is γ-regular and such that the densities between the sets V_i belong to some predefined set of densities.

Definition 5 (Regularity-Instance). *A* regularity-instance R *is given by an error-parameter* $0 < \gamma \leq 1$, *an integer* k, *a set of* $\binom{k}{2}$ *densities* $0 \leq \eta_{ij} \leq 1$ *indexed by* $1 \leq i < j \leq k$, *and a set* $\overline{R}$ *of pairs* (i,j) *of size at most* $\gamma\binom{k}{2}$. *A graph is said to* satisfy *the regularity-instance if it has an equipartition* $\{V_i \mid 1 \leq i \leq k\}$ *such that for all* $(i,j) \notin \overline{R}$ *the pair* (V_i, V_j) *is* γ*-regular and satisfies* $d(V_i, V_j) = \eta_{i,j}$. *The* complexity *of the regularity-instance is* $\max(k, 1/\gamma)$.

The first main result in this work is the following:

Theorem 6. *For any regularity-instance* R, *the property of satisfying* R *is testable.*

Definition 7 (Regular-Reducible). *Graph property* P *is* regular-reducible *if for any* $\delta > 0$ *there exists an* $r = r_P(\delta)$ *such that for any* n *there is a family* R *of at most* r *regularity-instances each of complexity at most* r, *such that the following holds for every* $\epsilon > 0$ *and every* n*-vertex graph* G:

(1) *If* G *satisfies* P *then for some* $R \in R$, G *is* δ*-close to satisfying* R.
(2) *If* G *is* ϵ*-far from satisfying* P, *then for any* $R \in R$, G *is* $(\epsilon - \delta)$*-far from satisfying* R.

Observe that in the above definition the value of δ may be arbitrarily close to 0. If we think of $\delta = 0$ then we get that a graph satisfies P if and only if it satisfies one of the regularity instances of R. With this (rough) interpretation in mind, in order to test P one can test the property of satisfying any one of the instances of R. Therefore, in some sense we "reduce" the testing of property P to the testing of regularity-instances. We are now ready to state the characterization of the testable graph properties.

Theorem 8. *A graph property is testable* **if and only if** *it is regular-reducible.*

The detailed proofs, applications, discussions and further references can be found in [1].

References

[1] N. Alon, E. Fischer, I. Newman and A. Shapira, A combinatorial characterization of the testable graph properties: it's all about regularity, Proc. of the 38^{th} Annual ACM Symposium on Theory of Computing (STOC), ACM Press (2006), 251-260.

[2] O. Goldreich, S. Goldwasser and D. Ron, Property testing and its connection to learning and approximation, Proc. of 37^{th} Annual IEEE FOCS, (1996), 339–348. Also, JACM 45(4): 653-750 (1998).

[3] E. Szemerédi, Regular partitions of graphs, In: *Proc. Colloque Inter. CNRS* (J. C. Bermond, J. C. Fournier, M. Las Vergnas and D. Sotteau, eds.), 1978, 399–401.

A (De)constructive Approach to Program Checking (or How to Delegate Work to Your Very Own Adversary)

Shafi Goldwasser
(joint work with Dan Gutfreund, Alexander Healy, Tali Kaufman), and Guy Rothblum)

One of the main challenges in software engineering is verifying the correctness of software. In the eighties Blum and Kannan [1] proposed the methodology of program "result checking", which focuses on correctness of the code *per input* rather than full program verification. The methodology associates every function to be computed with a new piece of code called the *checker*. Then, given any possibly buggy program for the function and any input, the checker "checks" whether the program on this input computes the function correctly. The work of Blum, Luby, and Rubinfeld [2] further introduced the notion of program *testers* and *correctors*. A tester determines whether a given program for a function is correct on random inputs (with relatively high probability). A corrector of a function is given an input and a program that is guaranteed to compute the function correctly on random inputs (but may be buggy on some inputs), and computes (with high probability) the correct output for the given input.

The focus of the rich body of work in the result checking field has been the design of efficient checkers (and tester/correctors) for many *specific* functions, by exploiting either their algebraic or combinatorial properties. Most notably, these functions include arithmetic operations, matrix operations, and certain graph and group operations. By and large, these are function families which possess random and downwards self-reducibility properties.

This body of work has also found applications beyond the field of program checking. The techniques introduced were pivotal in showing the expressive power of IP and PCP proof systems, and the notion of testers in and of itself has evolved into the successful field of property testing.

Since a correct algorithm for a given function is also trivially a checker for the function, [1] required, in order to avoid triviality, that checkers have the *little-oh time property*: the running time of the checker must be little-oh of the running time of the most efficient *known* program that computes the function. An analogue *little-oh parallel time property* was considered by Rubinfeld [3]: a checker's parallel running time should be little-oh of the parallel running time of the most efficient known program that computes the function. (Throughout, the standard complexity measure of oracle algorithms is used, where the complexity of the algorithm is measured *without* the complexity of the oracle's computations.)

New Work

The work demonstrates new checkers, testers and correctors that are all provably more efficient than the optimal program in terms of circuit depth for the functions at hand. These are designed using a new composition methodology for improving the circuit depth of checkers, testers and correctors. This approach may, in principal, also be useful to improve other complexity measures.

The idea is the following. We observe that a checker for a function f has access to a potentially powerful resource: the (allegedly correct) program P it is checking, which can often compute a complex function. Our goal is thus to *delegate* computations from the checker to the program being checked, all the while verifying that the results returned by the delegated computations are correct. To achieve this we start with a checker C for the function at hand – this C may be a previously designed checker, or even just a correct program for the function (which trivially gives a checker) – and then to decompose this checker into *sub-computations.* The work of these sub-computations is in turn replaced by calls to P, the potentially faulty program being checked, on appropriate inputs. This is done by applying a reduction that maps sub-computations to instances of the function f being allegedly computes. The correctness of these delegated sub-computations performed by P is finally verified by checkers for the sub-computations. When the checkers for the sub-computations are more efficient than the sub-computations themselves, this results in a new checker with improved efficiency.

The composition methodology provides a simple way to design checkers that is very similar to the top-down approach of algorithm design: break down the solution of a complex problem into the solution of smaller (and easier) sub-problems, and then combine these solutions, all the while ensuring errors are kept under control. In particular, this approach enables us to construct checkers for functions that do not necessarily have the type of self-reducibility or completeness properties exploited in previous works of [1, 2, 4, 5, 6] as follows.

We first use the Composition Theorem to build checkers that are provably more efficient than the functions they check (in terms of circuit depth) for entire complexity classes, and not just specific functions with special algebraic or combinatorial properties.

Theorem 1. *For every $i \geq 1$, every language in RNC^{i} that is NC^1-hard under NC^0-reductions has a checker in RNC^{i-1}. Every language in RNC^{i} that is NC^1-hard under AC^0 reductions has a tester and corrector that are in RAC^{i-1}.*

The requirement of being NC^1-hard under NC^0 reductions turns out to not be very restrictive. Examples of natural functions and languages that satisfy the theorem requirements include *graph connectivity* (in its many variants), deciding whether a given graph has a *prefect matching* and *bounded-degree graph isomorphism*, computing the determinant of a matrix, matrix exponentiation, and more.

Next, we turn to the design of parallel depth checkers for matrix functions.

Blum, Luby, and Rubinfeld [2] considered the problem of testing and correcting matrix functions such as multiplication, inverse, determinant and rank. However, they suggested a non-standard model in which the checker/tester/corrector can access (with unit cost) not only the program to be checked, but also a **library** of (possibly faulty) programs that allegedly compute other related functions. Within this extended model, they show how to test and correct (and thus check) programs for the above matrix functions.

Here, we present *standard* checkers, testers and correctors for matrix multiplication, inversion, rank and determinant, removing altogether the need for the matrix library model. These checkers/testers/correctors can be implemented in AC^0 (and for some ranges of parameters even in NC^0). They are provably more efficient than the optimal program for computing these functions in terms of circuit depth. Furthermore, we note that the checkers we build for matrix multiplication and matrix inversion are optimal up to constant factors in every parameter: depth (or parallel time), size (or number of processors) and number of queries.

Theorem 2. *Matrix multiplication, inversion, determinant and rank have all probabilistic AC^0 checkers, testers and correctors. For rank the result holds only over fields that are of size polynomial in the input length. Over a field of cardinality 2^s for a constant s, matrix multiplication and inversion have probabilistic NC^0 checkers, testers and correctors that perform a constant number of calls to the program.*

A few additional remarks are in order.

Important building blocks used in applying our composition methodology are efficient checkers for complete languages for low-level complexity classes. Our work shows for the first time how to leverage checkers for complete-problems toward the design of checkers for other non-complete problems in the class. Unlike other properties of functions (or languages), the existence of checkers for complete languages did not previously seem to imply or be related to the existence of checkers for non-complete language. Indeed, likely for this reason past work was more concerned with checkers for useful and practical functions and less with checkers for complete languages.

Finally, the paradigm of delegating computation to an untrusted component, which originated in this work, has yielded applications in other settings. In [8] we present new interactive proof systems where very efficient (NC^0) verifiers delegate their work to the provers, and new error correcting codes, where much of the decoder's work is delegated to the encoder (and embedded in the codeword itself). This once again illustrates the fruitful interplay between program checking and other areas in complexity theory.

References

[1] Manuel Blum and Sampath Kannan. Designing programs that check their work. *Journal of the ACM*, 42(1):269–291, 1995.

[2] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *Journal of Computer and System Sciences*, 47(3):549–595, 1993.

[3] R. Rubinfeld. Designing checkers for programs that run in parallel. *Algorithmica*, 15(4):287–301, 1996.

[4] R. Lipton. New directions in testing. *Proceedings of DIMACS workshop on distributed computing and cryptography*, 2:191–202, 1991.

[5] A. Shamir. IP = PSPACE. In *Proceedings of the 31st Annual IEEE Symposium on Foundations of Computer Science*, pages 11–15, 1990.

[6] L. Babai, L. Fortnow, and C. Lund. Non-deterministic exponential time has two-prover interactive protocols. In *Proceedings of the 31st Annual IEEE Symposium on Foundations of Computer Science*, pages 16–25, 1990.
[7] D.M. Barrington. Bounded-width polynomial-size branching programs recognize exactly those languages in NC^1. *Journal of Computer and System Sciences*, 38, 1989.
[8] Shafi Goldwasser, Dan Gutfreund, Alexander Healy, Tali Kaufman, and Guy Rothblum. Verifying and decoding in constant depth. *Proceedings of the 39th Annual ACM Symposium on Theory of Computing*, San Diego, CA, June 2007.

Reed-Solomon codes, low-degree automorphisms, and Error-correction with optimal rate

Venkatesan Guruswami
(joint work with Atri Rudra)

1. Introduction

A fundamental trade-off in coding theory is the one between the percentage of redundancy built into codewords and the fraction of errors that can be corrected. In this talk, we describe an *explicit* construction of codes that achieves the *optimal trade-off* between these parameters, along with polynomial time algorithms for performing the error-correction. Formally, for every $0 < p < 1$ and $\varepsilon > 0$, we present an explicit construction of codes with rate $(1 - p - \varepsilon)$ (i.e., which encode $(1-p-\varepsilon)n$ information symbols into n codeword symbols) that can be *list decoded* in polynomial time from up to a fraction p of errors.[1] Note that the trade-off we achieve is information-theoretically the best possible, since if the proportion of corectly received symbols is less than the rate, then we have less information at the receiving end than is contained in the message and error-correction is therefore not possible. For low noise levels (small p), our codes require almost a factor 2 less redundancy over the best previous result. We stress that our result holds in a worst-case noise model where the channel can corrupt the codeword arbitrarily subject to the bound p on the proportion of errors.

Our codes are simple to describe: they are certain "folded" Reed-Solomon (RS) codes, which are obtained from the classical RS codes via a bundling together of codeword symbols (the resulting encoding is treated as a string over a larger alphabet). A central algebraic idea used in our work is that certain automorphisms of rational function fields induce a low-degree map on the residue of polynomials modulo a certain large degree irreducible. Discovering a similar phenomenon over more general function fields, with the space of sections of some bounded-degree

[1] Under list decoding, the decoder must output a list of all possible messages that could have been transmitted given that up to a fraction p of errors could have occurred. In order to decode to achieve a rate exceeding $(1-2p)$, it is known that the decoder's worst-case output list size must exceed one. Allowing even moderate sized lists, however, opens up the possibility of achieving a much larger rate, and in particular approaching the information-theoretically optimal bound of $1 - p$.

divisor replacing polynomials, and their evaluations at a large degree "place" replacing residues modulo an irreducible, could lead to exciting new algebraic codes, and perhaps also progress towards achieving list decoding capacity over small, fixed alphabets.

The talk introduced and motivated the problem of list decoding, and then gave a peek into the technical ideas underlying the above result. The algebraic ideas behind these codes, and specifically their precursor, the Parvaresh-Vardy codes [5], have since yielded a new, simple construction of unbalanced bipartite expanders with expansion close to the degree [4]. In turn, this has led to the best known constructions (to date) of randomness extractors. These applications of the Parvaresh-Vardy codes in pseudorandomness were presented in detail in another talk by C. Umans.

We now present some technical details about the folded Reed-Solomon code construction and a formal statement of our main result.

2. Code description and Main result

Consider a Reed-Solomon code C_{RS} consisting of evaluations of degree k polynomials over a finite field $\mathbb{F}$ at the set $\mathbb{F}^*$ of nonzero elements of $\mathbb{F}$. Let $q = |\mathbb{F}| = n+1$. Let γ be a generator of the multiplicative group $\mathbb{F}^*$, and let the evaluation points be ordered as $1, \gamma, \gamma^2, \ldots, \gamma^{n-1}$. Using all nonzero field elements as evaluation points is one of the most commonly used instantiations of Reed-Solomon codes.

Let $m \geq 1$ be an integer parameter called the *folding parameter*. For convenience, let us assume that m divides $n = q - 1$.

Definition 1 (Folded Reed-Solomon Code)**.** *The m-folded version of the RS code C_{RS} is a code of block length $N = n/m$ over $\mathbb{F}^m$. The encoding of a message $f(X)$, a polynomial over $\mathbb{F}$ of degree at most k, has as its j'th symbol, for $0 \leq j < n/m$, the m-tuple $(f(\gamma^{jm}), f(\gamma^{jm+1}), \cdots, f(\gamma^{jm+m-1}))$. In other words, the codewords of the m-folded RS code are in one-one correspondence with those of the RS code C_{RS} and are obtained by bundling together consecutive m-tuple of symbols in codewords of C_{RS}.*

The folded version of a RS code thus carries the same information, just "bundled" differently. It is a code of exactly the same rate as the original RS code, but is defined over a larger alphabet. At a high level, folding restricts the flexibility in the subset of evaluation points that an adversary can corrupt.

We now state the main concerning decoding these codes from [2]. In the below statement, the parameter r governs the number of multiplicities at each point imposed during the interpolation (similar to the list decoding algorithm for Reed-Solomon codes from [6, 3]). The parameter s corresponds to the number of dimensions in the interpolation, and the stated bound is obtained through $(s+1)$-variate interpolation.

Theorem 2. *For every integer $r \geq 1$, every integer $m \geq 1$ and integer s, $1 \leq s \leq m$, the folded RS code with the parameters q, n, N, k from Definition (1) can be list* *decoded up to a radius*

$$N - \left(1 + \frac{s}{r}\right) \frac{(k^s n)^{1/(s+1)}}{m - s + 1} + 2 \ , \tag{1}$$

in time at most $(nr)^{O(s)}$*, and the list size output by the decoder will be at most* q^s*.*

By picking r, m large enough compared to s, and noting that the rate $R = k/n$ and $n = Nm$, the fraction of decoded errors can be made larger than $1 - (1 + \zeta)R^{s/(s+1)}$ for any desired $\zeta > 0$. In the limit of large s (specifically, for $s = \Theta(\varepsilon^{-1} \log(1/R))$), the decoding radius approaches the list decoding capacity $1 - R$, leading to the main conclusion of this work:

Theorem 3 (Explicit capacity-approaching codes)**.** *For every* $\varepsilon > 0$ *and* $0 < R < 1$*, there is a family of folded Reed-Solomon codes which have rate at least* R *and which can be list decoded up to a fraction* $1 - R - \varepsilon$ *of errors in time* $(N/\varepsilon^2)^{O(\varepsilon^{-1} \log(1/R))}$ *where* N *is the block length of the code. The alphabet size of the code as a function of the block length* N *is* $(N/\varepsilon^2)^{O(1/\varepsilon^2)}$*.*

3. Few words on the proof

We now describe at a very high level the key ingredients in proving the above results. For the full details, the reader is referred to [2, 1].

Let us focus on the statement of Theorem 2. By a rather straightforward extension of the bivariate interpolation based decoding paradigm from [6, 3] to the multivariate case, one can compute a non-zero $(s+1)$-variate polynomial $Q(X, Y_1, Y_2, \ldots, Y_s)$ over $\mathbb{F} = \mathbb{F}_q$ such that any degree k polynomial $f(X)$ whose folded RS encoding is within the radius (1) from the received word must satisfy

$$Q(X, f(X), f(\gamma X), \ldots, f(\gamma^{s-1} X)) = 0 \ . \tag{2}$$

The central algebraic step is then to find all degree k polynomials $f(X)$ for which (2) holds in an efficient manner, and in the process also prove that there cannot be too many solutions $f(X)$ to (2). The key insight here is to note the following two facts: (i) The (trivial) identity $f(\gamma X) \equiv f(X)^q \pmod{(X^{q-1} - \gamma)}$ that holds for all polynomials, and (ii) The polynomial $E(X) = X^{q-1} - \gamma$ is irreducible over $\mathbb{F}_q$. For $k < q - 1$, all solutions $f(X)$ of degree at most k to (2) can thus be found by solving the equation $Q(X, T, T^q, T^{q^2}, \ldots, T^{q^{s-1}}) = 0$ over $G = \mathbb{F}_q[X]/(E(X))$ for T. This amounts to finding the roots of a univariate polynomial over the extension field G, a task that can be performed efficiently.

The algebraic crux above was that the automorphism Γ of the function field $K = \mathbb{F}_q(X)$ induced by $X \mapsto \gamma X$ satisfied the identity $\Gamma(f) \mod E(X) = f^q \mod E(X)$ for all polynomials f. That is, the automorphism Γ induces a low-degree map w.r.t the evaluations of polynomials at the "place" corresponding to $E(X)$.

References

[1] V. Guruswami. *Algorithmic Results in List Decoding.* NOW publishers: Foundations and Trends in Theoretical Computer Science, Volume 2, Issue 2, 2007.

[2] V. Guruswami and A. Rudra. Explicit capacity-achieving list-decodable codes. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, pages 1–10, May 2006.

[3] V. Guruswami and M. Sudan. Improved decoding of Reed-Solomon and algebraic-geometric codes. *IEEE Transactions on Information Theory*, 45:1757–1767, 1999.

[4] V. Guruswami, C. Umans, and S. Vadhan. Unbalanced Expanders and Randomness Extractors from Parvaresh-Vardy Codes. In *Proceedings of the 22nd IEEE Conference on Computational Complexity*, pages 96–108, June 2007.

[5] F. Parvaresh and A. Vardy. Correcting errors beyond the Guruswami-Sudan radius in polynomial time. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 285–294, October 2005.

[6] M. Sudan. Decoding of Reed-Solomon codes beyond the error-correction bound. *Journal of Complexity*, 13(1):180–193, 1997.

Extractors for Independent Sources

Anup Rao

The use of randomness is widespread in computer science. Many of the best performing algorithms and protocols in many different areas of computer science are randomized. To guarantee their performance these algorithms usually rely on a perfect source of uncorrelated uniformly random bits, yet such a source may not be easy to obtain. We might instead have access to an imperfect random source where the bits are correlated and not uniformly random.

This motivates the study of objects called *extractors*. Informally, an extractor is an explicit efficiently computable function $\mathsf{Ext} : \{0,1\}^n \to \{0,1\}^m$ that takes as input bits from an imperfect random source and produces bits that are close to uniformly random (the distance of the output distribution from the uniform distribution is called the error of the extractor). If we had access to such a function, we could use it to extract truly random bits from an imperfect random source. We would then use the extracted bits in our application. Thus we could achieve performance guarantees even with imperfect sources of randomness.

The most general model for a defective source of randomness that has been considered to date is what we will call a *weak source* [CG88]. The only constraint on a weak source that supplies n total bits is that the probability of getting any particular string from the source is at most 2^{-k}, where k is called the *min-entropy* of the source. Such a source is called an (n, k)-source. Unfortunately it can be shown that there is no deterministic extractor that can extract from general weak sources.

A natural model for a source that would allow extraction to be feasible is to assume that the source consists of two or more independent parts, each with sufficient entropy. We say that a function Ext is a C-source extractor for entropy k if given any C independent sources with entropy k, $X_1, \ldots, X_{\mathsf{C}}$, $\mathsf{Ext}(X_1, \ldots, X_{\mathsf{C}})$ is close to being uniformly random.

Definition 1. *A function* $\mathsf{IndepExt} : (\{0,1\}^n)^{\mathsf{C}} \to \{0,1\}^m$ *is an* extractor for C independent sources *with min-entropy* k *and error* ϵ *if for any independent* (n,k) *sources* $X^1, \ldots, X^{\mathsf{C}}$ *we have that* $\mathsf{IndepExt}(X_1, \ldots, X_{\mathsf{C}})$ *is* ϵ*-close to the uniform distribution in terms of statistical distance.*

Another way to view 2-source extractors is as boolean matrices that look random in a strong sense: Every 2-source extractor for entropy k gives an $N \times N$ boolean matrix in which every $K \times K$ minor has roughly the same number of 1's and 0's, with $N = 2^n, K = 2^k$.

The independent sources model is one of the earliest models studied [SV86, Vaz85, CG88]. The probabilistic method shows that most functions are 2-source extractors requiring entropy that is just logarithmic in the total length of each of the sources. Explicit constructions are still very far from achieving this kind of performance. The classical Lindsey Lemma gives a 2-source extractor for sources on n bits with entropy $n/2$. No significant progress was made in improving the entropy requirements over this, until recently. In the last few years, sparked by new results in arithmetic combinatorics [BKT04], there were several results [BIW04, BKS$^+$05, Raz05, Bou05].

Construction	**Min-Entropy** k	**Output**	**Error**	**Ref**
poly$(1/\delta)$-source extractor	δn	$\Theta(n)$	$2^{-\Omega(n)}$	[BIW04]
3-source extractor	δn, any constant δ	$\Theta(1)$	$O(1)$	[BKS$^+$05]
3-source extractor	One source: δn, any constant δ. Other sources may have polylog(n) min-entropy.	$\Theta(1)$	$O(1)$	[Raz05]
2-source extractor	One source: $(0.5+\alpha)n$, $\alpha > 0$. Other source may have $k =$ polylog(n) min-entropy.	$\Theta(k)$	$2^{-\Omega(k)}$	[Raz05]
2-source extractor	$(0.5-\alpha_0)n$ for some universal constant $\alpha_0 > 0$	$\Theta(n)$	$2^{-\Omega(n)}$	[Bou05]
$O(1/\delta)$-source extractor	n^δ	$\Theta(k)$	$k^{-\Omega(1)}$	[Rao06]
$O(1/\delta)$-source extractor	n^δ	$\Theta(k)$	$2^{-k^{\Omega(1)}}$	[BRSW06]
3-source extractor (with constraints on input lengths)	n^δ for any constant δ (additional constraints apply)	$k - o(k)$	$2^{-k^{\Omega(1)}}$	[LRZ07]

Table 1. Performance of recent extractors for independent sources

Results in the Talk. We give a polynomial time computable extractor that extracts k random bits from $O(\frac{\log n}{\log k})$ independent (n,k)-sources with error k^{-c} for any $k(n) > \log^4 n$ and some universal constant $c > 1$. An interesting setting of parameters is when $k = n^\gamma$ for some $0 < \gamma < 1$. In this case we get an extractor for a constant number of sources that extracts a constant fraction of the total min-entropy with exponentially small error.

Techniques. Many extractor constructions in the past have been based on the paradigm of iterative condensing [RSW00, TUZ01, CRVW02, LRVW03, BIW04]. The idea is to start with some distribution that has low min-entropy and apply a function (called a *condenser*) whose output has a better min-entropy rate. Repeating this process, we eventually obtain a distribution which has very high min-entropy rate. Then we can apply some other extractor which works for such a high min-entropy rate to obtain random bits. The extractor in this paper can also be viewed as an example of this paradigm, with a slight twist.

We make progress by considering a more restricted model for sources called *somewhere random* sources [TS96]. A source is somewhere random if it samples from some distribution on boolean matrices, such that at least one of the rows of the matrix is distributed uniformly. An important concept that we introduced in that chapter is that of *aligned* somewhere random sources. Two somewhere random sources with the same number of rows are said to be *aligned* if there is an i such that the ith row of both sources are distributed uniformly.

We think of the number of rows of a somewhere random sources as a measure of the quality of the source. The fewer the number of rows, the better the quality is. Our construction works by iteratively improving the quality (reducing the number of rows) of the somewhere random sources that we are working with until extracting randomness from them becomes easy.

References

[BIW04] Boaz Barak, R. Impagliazzo, and Avi Wigderson. Extracting randomness using few independent sources. In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 384–393, 2004.

[BKS+05] Boaz Barak, Guy Kindler, Ronen Shaltiel, Benny Sudakov, and Avi Wigderson. Simulating independence: New constructions of condensers, Ramsey graphs, dispersers, and extractors. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 1–10, 2005.

[BRSW06] Boaz Barak, Anup Rao, Ronen Shaltiel, and Avi Wigderson. 2 source dispersers for $n^{o(1)}$ entropy and Ramsey graphs beating the Frankl-Wilson construction. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006.

[Bou05] Jean Bourgain. More on the sum-product phenomenon in prime fields and its applications. *International Journal of Number Theory*, 1:1–32, 2005.

[BKT04] Jean Bourgain, Nets Katz, and Terence Tao. A sum-product estimate in finite fields, and applications. *Geometric and Functional Analysis*, 14:27–57, 2004.

[CRVW02] M. Capalbo, Omer Reingold, Salil Vadhan, and Avi Wigderson. Randomness conductors and constant-degree lossless expanders. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pages 659–668, 2002.

[CG88] Benny Chor and Oded Goldreich. Unbiased bits from sources of weak randomness and probabilistic communication complexity. *SIAM Journal on Computing*, 17(2):230–261, 1988.

[LRZ07] Xin Li, Anup Rao, and David Zuckerman. Network extractor protocols and three-source extractors. *Manuscript*, 2007.

[LRVW03] C. J. Lu, Omer Reingold, Salil Vadhan, and Avi Wigderson. Extractors: Optimal up to constant factors. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pages 602–611, 2003.

[Rao06] Anup Rao. Extractors for a constant number of polynomially small min-entropy independent sources. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, 2006.

[Raz05] Ran Raz. Extractors with weak random seeds. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 11–20, 2005.

[RSW00] Omer Reingold, Ronen Shaltiel, and Avi Wigderson. Extracting randomness via repeated condensing. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science*, pages 22–31, 2000.

[SV86] Miklos Santha and Umesh V. Vazirani. Generating quasi-random sequences from semi-random sources. *Journal of Computer and System Sciences*, 33:75–87, 1986.

[TS96] Amnon Ta-Shma. Refining randomness. In *ECCCTH: Electronic Colloquium on Computational Complexity, theses*, 1996.

[TUZ01] Amnon Ta-Shma, Chris Umans, and David Zuckerman. Loss-less condensers, unbalanced expanders, and extractors. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, pages 143–152, 2001.

[Vaz85] Umesh Vazirani. Towards a strong communication complexity theory or generating quasi-random sequences from two communicating slightly-random sources (extended abstract). In *Proceedings of the 17th Annual ACM Symposium on Theory of Computing*, pages 366–378, 1985.

Unbalanced Expanders and Randomness Extractors from Parvaresh-Vardy Codes

Chris Umans

(joint work with Venkatesan Guruswami, Salil Vadhan)

A long line of work has been devoted to obtaining explicit constructions of *randomness extractors*, defined below:

Definition 1. *A function* $E : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *is a* (k, ϵ) extractor *if for every* $\mathbf{X}$ *with minentropy at least* k*,* $E(\mathbf{X}, \mathbf{Y})$ *is* ϵ*-close to uniform, when* $\mathbf{Y}$ *is uniformly distributed on* $\{0,1\}^d$*. An extractor is* explicit *if it is computable in polynomial time.*

Many applications have been found for these objects in a diverse range of settings, and consequently a lot of effort has been spent trying to find explicit constructions. An extractor construction matching the non-constructive bounds would have a seed length of $d = \log n + 2\log(1/\epsilon) + O(1)$ and an output length of $m = k + d - 2\log(1/\epsilon) - O(1)$. We still do not have explicit constructions of optimal extractors; the best construction prior to this work was by Lu, Reingold, Vadhan, and Wigderson [LRVW03], who get within constant multiplicative factors of optimal in both the seed length and the output length (for ϵ that is not too small).

In this talk we describe a new explicit construction of extractors that matches [LRVW03], and also handles small ϵ. One of the main advantages of our result is the simplicity of the construction and its proof.

The construction works by first building an intermediate object called a *lossless condenser*, defined below:

Definition 2. *A function* $C : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^m$ *is an* $k \to_\epsilon k'$ condenser *if for every* $\mathbf{X}$ *with minentropy at least* k*,* $C(\mathbf{X}, \mathbf{Y})$ *is* ϵ-close *to a distribution with minentropy* k'*, when* $\mathbf{Y}$ *is uniformly distributed on* $\{0,1\}^d$*. A condenser is* explicit *if it is computable in polynomial time. A condenser is called* lossless *if* $k' = k + d$*.*

Nonconstructively, there exist lossless condensers with seed length $d = \log n + \log(1/\epsilon) + O(1)$ and output length $m = k + d + \log(1/\epsilon) + O(1)$. The best previous constructions of lossless condensers were by Ta-Shma, Umans, and Zuckerman [TUZ01], with a further improvement in [TU06]. These construction get within a constant multiplicative factor of optimal in either of the two parameters (seed length and output length), at the expense of being a super-constant multiplicative factor away from optimal in the other. Our main construction achieves a lossless condenser that is within a constant factor of optimal in both parameters simultaneously.

It is most natural to describe our main construction as an explicit construction of yet another object, an (unbalanced, bipartite) *expander graph*, defined below:

Definition 3. *A bipartite (multi)graph* $G = (U, V, E)$ *is a* (K, A) expander *if for every set* $S \subseteq U$ *of size* K*, we have* $|\Gamma(S)| \geq A \cdot K$*.*

It turns out that a $(K = 2^k, (1-\epsilon)2^d)$ expander is equivalent to a $k \to_\epsilon k + d$ (lossless) condenser with seed length d [TUZ01]. In the language of expanders, our main construction achieves expansion $(1-\epsilon)D$ while the degree D is a polynomial in the optimal $O(\log N/\epsilon)$ (for the unbalanced case), and the right-hand side has size that is also a polynomial in the optimal $O(KD/\epsilon)$.

Our main construction is based on Parvaresh-Vardy codes [PV05], and its proof essentially amounts to a tight analysis of the so-called “list-recovering” properties of these codes, which follows the list-decoding analysis of [PV05] closely. Let F_q be the field with q elements, and let $h < q$ be a parameter we will set later. Let $E(Y)$ be a degree n polynomial that is irreducible over $\mathbb{F}_q$. We identify the left-hand-side of the bipartite graph with the set of degree $n-1$ univariate polynomials over $\mathbb{F}_q$. Given such a polynomial $f(Y)$, define $f_i(Y)$ to be $f(Y)^{h^i} \bmod E(X)$. For each field element $y \in \mathbb{F}_q$, the y-th neighbor of f is $(y, f_0(y), f_1(y), \ldots, f_m(y)) \in \mathbb{F}_q^{m+1}$. This is just y prepended to the y-th coordinate of the Parvaresh-Vardy codeword corresponding to f.

The natural way to prove expansion would be to fix a subset of size K of the left-hand-side, and argue that its neighbor set has size at least AK. An equivalent proof strategy, that will be useful here, fixes a subset T of size $AK-1$ of the right-hand-side, and argues that that the set $LIST(T)$, consisting of all left-hand-side vertices whose neighbor set is entirely contained within T, has size at most $K-1$.

Now we can present the proof, which is algebraic, short, and self-contained. The main lemma is:

Lemma 4. *For $A = q - nmh$, and $K = h^m$, if $T \subseteq \mathbb{F}_q^{m+1}$ has size $AK-1$ then $LIST(T) \subseteq \mathbb{F}_q^n$ has size at most $K-1$.*

Proof. Fix a set $T \subseteq \mathbb{F}_q^{m-1}$. Let $Q(Y, Z_0, \ldots, Z_{m-1})$ be a non-zero polynomial that vanishes on T, with $\deg(Y) \leq A-1$, and $\deg(Z_i) \leq h-1$. We assume that $E(Y)$ does not divide Q, without loss of generality.

Now consider a degree $n-1$ polynomial f in $LIST(T)$. By definition all of its neighbors are in T, and so $\forall y \;\; Q(y, f_0(y), \ldots, f_{m-1}(y)) = 0$. But this univariate polynomial in y has degree less than q, and it vanishes on q points, so it must be the zero polynomial. Substituting the definition of the f_i, we have

$$Q(Y, (f(Y) \bmod E(Y)), (f(Y)^h \bmod E(Y)), \ldots, (f(Y)^{h^{m-1}} \bmod E(Y))) \equiv 0.$$

and the same holds after taking the left-hand-side modulo $E(Y)$.

Thus, over the extension field $\mathbb{F}_q[Y]/E(Y)$ (in which $f(Y)$ is a field element), we have $Q(Y, f(Y), f(Y)^h, \ldots, f(Y)^{h^{m-1}}) = 0$, or equivalently, that f is a *root* of the univariate polynomial $Q^*(Z) = Q(Y, Z, Z^h, \ldots, Z^{h^{m-1}}) \bmod E(Y)$. Finally, the degree of Q^* is at most $(h-1)(1+h+h^2+\ldots+h^{m-1}) = h^m - 1$, which is an upper bound on the size of $LIST(T)$. $\square$

A minor tweak to this proof gives an upper bound of $K'-1$ on $|LIST(T)|$, given a T of size $AK'-1$, for any $K' \leq K$. Setting $h = (nm/\epsilon)^{1/\alpha}$ and $q = h^{1+\alpha}$ delivers the promised expansion, degree, and right-hand-side size.

Viewed as a condenser, this construction condenses an arbitrary source losslessly into one with entropy rate any constant arbitrarily close to 1. We can apply known extractors [Zuc97] to such a source to extract a constant fraction of the minentropy; for constant ϵ, we can even use the well-known "expander-walk" extractor which is very simple to describe and analyze with Chernoff-type bounds for expander walks [Gil98].

Other ideas from list-decodable error-correcting codes are useful in our setting as well. The "repeated roots" idea from [GS99] can be used to reduce the seed length (at the expense of some entropy loss). Ideas from [GR06] can be used to show that a very natural construction based on Reed-Solomon codes yields a good condenser. Specifically, the y-th neighbor is now $(f(y), f(\alpha y), \ldots, f(\alpha^{m-1}y)) \in \mathbb{F}_q^m$, where α is a generator of $\mathbb{F}_q^*$, and essentially the same proof technique shows that this is a condenser whose output retains a $(1-\delta)$ fraction of the entropy, for any $\delta > 0$.

In summary, we give the best explicit constructions of randomness extractors and unbalanced bipartite expander graphs (a.k.a. "lossless condensers") to date, using a simple construction and proof technique based on Parvaresh-Vardy codes.

References

[Gil98] D. Gillman. A Chernoff bound for random walks on expander graphs. *SIAM J. Comput.*, 27(4):1203–1220 (electronic), 1998.

[GR06] V. Guruswami and A. Rudra. Explicit capacity-achieving list-decodable codes. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, pages 1–10, 2006.

[GS99] V. Guruswami and M. Sudan. Improved decoding of Reed-Solomon and Algebraic-Geometry codes. *IEEE Transactions on Information Theory*, 45(6):1757–1767, 1999.

[LRVW03] C.-J. Lu, O. Reingold, S. Vadhan, and A. Wigderson. Extractors: Optimal up to constant factors. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pages 602–611, 2003.

[PV05] F. Parvaresh and A. Vardy. Correcting errors beyond the Guruswami-Sudan radius in polynomial time. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 285–294, 2005.

[TU06] A. Ta-Shma and C. Umans. Better lossless condensers through derandomized curve samplers. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, pages 177–186, 2006.

[TUZ01] A. Ta-Shma, C. Umans, and D. Zuckerman. Loss-less condensers, unbalanced expanders, and extractors. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, pages 143–152, 2001.

[Zuc97] D. Zuckerman. Randomness-optimal oblivious sampling. *Random Struct. Algorithms*, 11(4):345–367, 1997.

Statistically Hiding Commitments from Any One-Way Function

Salil Vadhan

(joint work with I. Haitner, M. Nguyen, S.J. Ong, O. Reingold)

As first discovered by Shannon [22] for the case of encryption, most interesting cryptographic tasks are impossible to achieve with absolute, information-theoretic security. Thus, modern cryptography aims to design protocols that are computationally intractable to break. Specifically, following Diffie and Hellman [5], this is typically done by showing that breaking the protocol is as hard as some intractable problem from complexity theory. Unfortunately, proving lower bounds of the sort needed seems beyond the reach of current techniques in complexity theory, and indeed would require at least proving $P \neq NP$.

Given this state of affairs, research in the foundations of cryptography has aimed to design cryptographic protocols based on complexity assumptions that are as weak and general as possible. This project was enormously successful in the 1980's. In a beautiful sequence of works, it was shown that many cryptographic primitives, such as pseudorandom generators, pseudorandom functions, private-key encryption and authentication, digital signatures, (computationally hiding) bit commitment, and (computational) zero-knowledge proofs could be constructed from any one-way function [12, 6, 21, 16, 8], and moreover this complexity assumption is minimal in the sense that each of these primitives (and indeed almost any cryptographic task) implies the existence of one-way functions [13, 20]. Moreover, it was shown that many of the remaining primitives, such as public-key encryption, collision-resistant hashing, and oblivious transfer, could not be reduced to the existence of one-way functions in a "black-box" manner [14, 23].

However, a few important primitives have resisted classification into the above categories. That is, it is only known how to build these primitives from seemingly stronger assumptions than the existence of one-way functions, yet there is no black-box separation between these primitives and one-way functions. In this work, we are interested in statistically hiding and computationally binding commitment schemes.

Statistically Hiding Commitments. A commitment scheme defines a two-stage interactive protocol between a sender S and a receiver R; informally, after the *commit stage*, S is bound to (at most) one value, which stays hidden from R, and in the *reveal stage* R learns this value. The two security properties hinted at in this informal description are known as *binding* (namely, that S is bound to at most one value after the commit stage) and *hiding* (namely, that R does not learn the value to which S commits before the reveal stage). In a statistically hiding computationally-binding commitment scheme the hiding property holds *even against all-powerful receivers* (i.e., hiding holds information-theoretically), while the binding property is required to hold only for polynomially-bounded senders.

Statistical commitment schemes can be used as a building block in constructions of statistical zero-knowledge arguments [3, 17] and certain coin-tossing protocols [15]. It therefore implies, via standard reduction, a way to transform a large class of protocols that are secure assuming an all powerful honest-but-curious party, into one that is secure even when this party maliciously deviates from the protocol. More generally, when used within protocols in which certain commitments are never revealed, statistical commitments have the following advantage over computationally-hiding commitment schemes: in such a scenario, it needs only be infeasible to violate the binding property *during the period of time the protocol is run*, whereas the committed values will remain hidden *forever* (i.e., regardless of how much time the receiver invests after completion of the protocol).

Perfectly-hiding[1] commitment schemes were first shown to exist based on specific number-theoretic assumptions [1, 3] or, more generally, based on any collection of claw-free permutations [9] with an efficiently-recognizable index set [7]. Statistical commitment schemes can also be constructed from collision-resistant hash functions [4, 18]. Naor et al. [17] showed a construction of a perfectly-hiding commitment scheme based on any one-way permutation. Haitner et. al. [10] make progress by constructing statistical commitment based on regular one-way functions and also on the so called approximable-size one-way functions. The question of whether one-way functions imply statistical commitments, however, was left open.

[1] Very informally, in a statistically-hiding commitment scheme the receiver learns only a negligible amount of information about the sender's committed value, whereas in a perfectly-hiding commitment scheme the receiver learns *nothing*. Note that any perfectly-hiding scheme is trivially also statistically hiding.

We mention that the complementary notion of commitment schemes, where the hiding is computational and the binding holds even w.r.t. an all powerful sender, was already known to be implied by the existence of one-way functions [12, 16].

Our result. Our main result is that the existence of one-way functions is a sufficient condition for the existence of statistical commitment. Namely, we prove the following theorem.

Theorem 1. *Assuming that one-way functions exist, then there exists a statistically-hiding and computationally-binding commitment scheme.*

By Impagliazzo and Luby [13], the existence of statistical commitment schemes implies the existence of one-way functions and thus the above result is tight.

One of the main applications of statistically hiding commitments are statistical zero-knowledge arguments [2, 8, 3]. These are zero-knowledge protocols for proving membership in any NP language where the zero knowledge property is statistical (i.e. even a computationally unbounded verifier learns nothing from the protocol) and the soundness is computational (i.e. no polynomial-time prover can convince the verifier of a false assertion, except with negligible probability). Thus, we also deduce:

Theorem 2. *Assuming that one-way functions exist, then every language in* NP *has a statistical zero-knowledge argument system.*

References

[1] J. F. Boyar, S. A. Kurtz, and M. W. Krentel. A discrete logarithm implementation of perfect zero-knowledge blobs. *J. Cryptology*, 2(2):63–76, 1990.

[2] G. Brassard, D. Chaum, and C. Crépeau. Minimum disclosure proofs of knowledge. *J. Computer & System Sci.*, 37(2):156–189, 1988.

[3] G. Brassard, C. Crépeau, and M. Yung. Constant-round perfect zero-knowledge computationally convincing protocols. *Theoretical Computer Sci.*, 84(1):23–52, 1991.

[4] I. B. Damgård, T. P. Pedersen, and B. Pfitzmann. Statistical secrecy and multibit commitments. *IEEE Trans. Information Theory*, 44(3):1143–1151, 1998.

[5] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Trans. Information Theory*, 22(6):644–654, 1976.

[6] O. Goldreich, S. Goldwasser, and S. Micali. How to construct random functions. *J. ACM*, 33(4):792–807, 1986.

[7] O. Goldreich and A. Kahan. How to construct constant-round zero-knowledge proof systems for NP. *J. Cryptology*, 9(3):167–190, 1996.

[8] O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *J. ACM*, 38(1):691–729, 1991.

[9] S. Goldwasser, S. Micali, and R. L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM J. Computing*, 17(2):281–308, 1988.

[10] I. Haitner, O. Horvitz, J. Katz, C. Koo, R. Morselli, and R. Shaltiel. Reducing complexity assumptions for statistically-hiding commitment. In *EUROCRYPT '05*, vol. 3494 of Springer LNCS.

[11] I. Haitner and O. Reingold. Statistically-hiding commitment from any one-way function. In *Proc. 39th STOC*, 2007.

[12] J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

[13] R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography. In *Proc. 30th FOCS*, 1989.

[14] R. Impagliazzo and S. Rudich. Limits on the provable consequences of one-way permutations. In *Proc. 21st STOC*, 1989.
[15] Y. Lindell. Parallel coin-tossing and constant-round secure two-party computation. *J. Cryptology*, 2003.
[16] M. Naor. Bit commitment using pseudorandomness. *J. Cryptology* 4(2):151–158, 1991.
[17] M. Naor, R. Ostrovsky, R. Venkatesan, and M. Yung. Perfect zero-knowledge arguments for NP using any one-way permutation. *J. Cryptology*, 11(2):87–108, 1998.
[18] M. Naor and M. Yung. Universal one-way hash functions and their cryptographic applications. In *Proc. 21st STOC*, 1989.
[19] M. Nguyen, S. J. Ong, and S. Vadhan. Statistical zero-knowledge arguments for NP from any one-way function. In *Proc. 47th FOCS*, 2006.
[20] R. Ostrovsky and A. Wigderson. One-way functions are essential for non-trivial zero-knowledge. In *Proc. 2nd ISTCS*, 1993.
[21] J. Rompel. One-way functions are necessary and sufficient for secure signatures. In *Proc. 22nd STOC)*, 1990.
[22] C. Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, 28(4):656–715, 1949.
[23] D. Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In *EUROCRYPT '98*, vol. 1403 of Springer LNCS, 1998.

Norms, XOR lemmas, and lower bounds for $GF(2)$ polynomials and multiparty protocols

Avi Wigderson
(joint work with Emanuele Viola)

A natural measure of agreement between two functions is their *correlation*, which measures the agreement on a random input. Formally, the correlation between two functions $f, p \in \{0,1\}^n \to \{-1,1\}$ is defined as

$$\mathrm{Cor}(f,p) := |E_x[f(x)\cdot p(x)]| = \left|\Pr_x[f(x) = p(x)] - \Pr_x[f(x) \neq p(x)]\right| \in [0,1].$$

For a complexity class C (e.g., circuits of size s on n bits), we denote by $\mathrm{Cor}(f,C)$ the maximum of $\mathrm{Cor}(f,p)$ over all functions $p \in C$. In other words, $\mathrm{Cor}(f,C)$ captures how well on average can we compute f using a function from C.

Correlation bounds are fundamental in computational complexity. Proving that $\mathrm{Cor}(f,C) < 1$ is equivalent to establishing that $f \notin C$, but what is far more desired is proving that $\mathrm{Cor}(f,C)$ is very close to zero, for natural functions f and complexity classes C. Such bounds yield pseudorandom generators that "fool" the class C (e.g. [10, 11, 9, 16]), and they also imply lower bounds for richer classes related to C (e.g., if $\mathrm{Cor}(f,C) < 1/t$ then f cannot be computed exactly by any function which is the majority of any t functions from C [4]). For these applications, we would like to prove correlation bounds as close to zero as possible.

A celebrated way of decreasing correlation (a.k.a. amplifying hardness) is via an *XOR lemma*, first suggested by Yao in his seminal paper [18] (cf. [3]). One starts with a function f of nontrivial correlation with C, and constructs a new function $f^{\oplus m}$ (on $n \cdot m$ bits), which is the exclusive-OR of the value of f on m independent inputs. The hope is that the correlation will decay exponentially with m. This idea is best demonstrated in the information-theoretic setting, in which we try to compute the value of a biased coin. In our language, take C to be the class of constant functions, and f any function with $|E_x[f(x)]| = \mathrm{Cor}(f, C) = \epsilon$. Then it is easy to see that $\mathrm{Cor}(f^{\oplus m}, C) = \epsilon^m$ for every m.[1] So the decay of the correlation in this trivial scenario is purely exponential in the number of copies m.

Yao's XOR lemma deals with the most studied computational model, namely polynomial-size circuits, and goes as follows. Let C be the class of Boolean circuits of size s, and let f be any function on n bits with $\mathrm{Cor}(f, C) \leq \epsilon$. Then for any large m and small $\alpha > 0$, if C' is the class of circuits of size $s \cdot (\alpha/nm)^2$ then $\mathrm{Cor}(f^{\oplus m}, C') \leq \epsilon^m + \alpha$. Many proofs of this XOR lemma were given, starting with Levin [8, 5, 3, 6]. All in fact show that that this lemma holds in more general circumstances, namely as long as C can compute the majority of functions in C'. However, none of these proofs can be applied to the computational models for which we actually can establish the existence of functions with non-trivial correlation (i.e. prove lower bounds), such as low-degree $GF(2)$ polynomials, multiparty protocols, or constant-depth circuits (cf. [14, Chapter 6]). Specifically, none of the above proofs can be applied to obtain a correlation bound of $1/n$ for a function on n bits. Another weakness of the results in [8, 5, 3, 6] is their loss in resources (e.g., circuit size) in C' with respect to C (cf. [3]).

Our results. In this paper we prove new XOR lemmas for two models: low-degree polynomials over $GF(2)$, and low-communication multiparty protocols.

We show that if a Boolean function has correlation at most $\epsilon \leq 1/2$ with any of these models, then the correlation of the parity of its values on m independent instances drops exponentially with m. More specifically:

- For $GF(2)$ polynomials of degree d, the correlation drops to $\exp\left(-m/4^d\right)$. No XOR lemma was known even for $d = 2$.
- For c-bit k-party protocols, the correlation drops to $2^c \cdot \epsilon^{m/2^k}$. No XOR lemma was known for $k \geq 3$ parties.

Another contribution in this paper is a general derivation of direct product lemmas from XOR lemmas. In particular, assuming that f has correlation at most $\epsilon \leq 1/2$ with any of the above models, we obtain the following bounds on the probability of computing m independent instances of f correctly:

- For $GF(2)$ polynomials of degree d we again obtain a bound of $\exp\left(-m/4^d\right)$.
- For c-bit k-party protocols we obtain a bound of $2^{-\Omega(m)}$ in the special case when $\epsilon \leq \exp\left(-c \cdot 2^k\right)$. In this range of ϵ, our bound improves on a direct product lemma for two parties by Parnafes, Raz, and Wigderson [12].

We also give improved (or just simplified) lower bounds in these models. In particular we give a new proof that the Mod_m function on n bits, for odd m, has correlation at most $\exp(-n/4^d)$ with degree-d $GF(2)$ polynomials.

Both proofs of our XOR lemmas use a common approach, very different from the one used for circuits. To each of these complexity classes C we associate a real

[1] Strictly speaking, now C denotes the constant functions on $n \cdot m$ bits.

norm N on all Boolean functions which has the following properties (informally stated):

(1) N CAPTURES CORRELATION WITH C. For every function f, $N(f) \approx \mathrm{Cor}(f, C)$.
(2) N COMMUTES WITH XOR. Let f, g be two functions on *disjoint* inputs, then $N(f \cdot g) = N(f) \cdot N(g)$.

Given such a norm N, the proof of an XOR lemma for C is straightforward:

$$\mathrm{Cor}(f^{\oplus m}, C) \approx N(f^{\oplus m}) = N(f)^m \approx \mathrm{Cor}(f, C)^m.$$

Of course, the challenge is to find the appropriate norm and prove their properties. We explain how such norms are related to the basic question of "Property Testing," where the problem being tested is how close is the given function f to the class C.

We also explain how such norms suggest themselves when the class C in question is a linear space. This is the case with $GF(2)$ polynomials, in which case indeed the well-known Gowers' norms are used. For multi-party protocols, which are *not* a linear space, we explain how they can be "approximated" by a linear space, for which a natural norm is implicit in previous papers.

References

[1] L. Babai, N. Nisan, and M. Szegedy. Multiparty protocols, pseudorandom generators for logspace, and time-space trade-offs. *J. Comput. System Sci.*, 45(2):204–232, 1992. Twenty-first Symposium on the Theory of Computing (Seattle, WA, 1989).

[2] A. K. Chandra, M. L. Furst, and R. J. Lipton. Multi-party protocols. In *Proceedings of the 15th Annual ACM Symposium on Theory of Computing*, Boston, Massachusetts, 25–27 Apr. 1983.

[3] O. Goldreich, N. Nisan, and A. Wigderson. On Yao's XOR lemma. Technical Report TR95–050, Electronic Colloquium on Computational Complexity, March 1995.

[4] A. Hajnal, W. Maass, P. Pudlák, M. Szegedy, and G. Turán. Threshold circuits of bounded depth. *J. Comput. System Sci.*, 46(2):129–154, 1993.

[5] R. Impagliazzo. Hard-core distributions for somewhat hard problems. In *36th Annual Symposium on Foundations of Computer Science*, pages 538–545, Milwaukee, Wisconsin, 23–25 Oct. 1995. IEEE.

[6] R. Impagliazzo and A. Wigderson. $P = BPP$ if E requires exponential circuits: Derandomizing the XOR lemma. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, pages 220–229, El Paso, Texas, 4–6 May 1997.

[7] E. Kushilevitz and N. Nisan. *Communication complexity.* Cambridge University Press, Cambridge, 1997.

[8] L. A. Levin. One way functions and pseudorandom generators. *Combinatorica*, 7(4):357–363, 1987.

[9] M. Luby, B. Velickovic, and A. Wigderson. Deterministic approximate counting of depth-2 circuits. In *Proceedings of the 2nd Israeli Symposium on Theoretical Computer Science (ISTCS)*, pages 18–24, 1993.

[10] N. Nisan. Pseudorandom bits for constant depth circuits. *Combinatorica*, 11(1):63–70, 1991.

[11] N. Nisan and A. Wigderson. Hardness vs randomness. *J. Computer & Systems Sciences*, 49(2):149–167, Oct. 1994.

[12] I. Parnafes, R. Raz, and A. Wigderson. Direct product results and the GCD problem, in old and new communication models. In *STOC '97 (El Paso, TX)*, pages 363–372. ACM, New York, 1999.

[13] A. A. Razborov. Lower bounds on the dimension of schemes of bounded depth in a complete basis containing the logical addition function. *Mat. Zametki*, 41(4):598–607, 623, 1987.

[14] E. Viola. *The Complexity of Hardness Amplification and Derandomization*. PhD thesis, Harvard University, 2006. http://www.eccc.uni-trier.de/eccc.

[15] E. Viola. New correlation bounds for GF(2) polynomials using gowers uniformity. *Electronic Colloquium on Computational Complexity*, Technical Report TR06-097, 2006. http://www.eccc.uni-trier.de/eccc.

[16] E. Viola. Pseudorandom bits for constant-depth circuits with few arbitrary symmetric gates. *SIAM Journal on Computing*, 36(5):1387–1403, 2007.

[17] E. Viola and A. Wigderson. Norms, XOR lemmas, and lower bounds for $GF(2)$ polynomials and multiparty protocols. In Proceedings of the 22nd Annual IEEE Conference on Computational Complexity (San Diego, CA), pages 141-154.

[18] A. C. Yao. Theory and applications of trapdoor functions (extended abstract). In *23rd Annual Symposium on Foundations of Computer Science*, pages 80–91, Chicago, Illinois, 3–5 Nov. 1982. IEEE.

[19] A. C.-C. Yao. Some complexity questions related to distributive computing. In *STOC '79: Proceedings of the eleventh annual ACM symposium on Theory of computing*, pages 209–213, New York, NY, USA, 1979. ACM Press.

Towards 3-Query Locally Decodable Codes of Subexponential Length

Sergey Yekhanin

1. Abstract

A q-query Locally Decodable Code (LDC) encodes an n-bit message x as an N-bit codeword $C(x)$, such that one can probabilistically recover any bit x_i of the message by querying only q bits of the codeword $C(x)$, even after some constant fraction of codeword bits has been corrupted.

We give new constructions of three query LDCs of vastly shorter length than that of previous constructions. Specifically, given any Mersenne prime $p = 2^t - 1$, we design three query LDCs of length $N = \exp\left(n^{1/t}\right)$, for every n. Based on the largest known Mersenne prime, this translates to a length of less than $\exp\left(n^{10^{-7}}\right)$, compared to $\exp\left(n^{1/2}\right)$ in the previous constructions.

2. Introduction

Classical error-correcting codes allow one to encode an n-bit string x into in N-bit codeword $C(x)$, in such a way that x can still be recovered even if $C(x)$ gets corrupted in a number of coordinates. For instance, codewords $C(x)$ of length $N = O(n)$ already suffice to correct errors in up to δN locations of $C(x)$ for any constant $\delta < 1/4$. The disadvantage of classical error-correction is that one needs to consider all or most of the (corrupted) codeword to recover anything about x. Now suppose that one is only interested in recovering one or a few bits of x. In such case more efficient schemes are possible. Such schemes are known as locally decodable codes (LDCs). Locally decodable codes allow reconstruction of an arbitrary bit x_i, from looking only at q randomly chosen coordinates of $C(x)$, where q can be as small as 2. Locally decodable codes have found numerous applications in complexity theory and cryptography. See [10], [5] for a survey. Below is a slightly informal definition of LDCs:

A (q, δ, ϵ)-locally decodable code encodes n-bit strings to N-bit codewords $C(x)$, such that for every $i \in [n]$, the bit x_i can be recovered with probability $1 - \epsilon$, by a randomized decoding procedure that makes only q queries, even if the codeword $C(x)$ is corrupted in up to δN locations.

One should think of $\delta > 0$ and $\epsilon < 1/2$ as constants. The main parameters of interest in LDCs are the length N and the query complexity q. Ideally we would like to have both of them as small as possible. The notion of locally decodable codes was explicitly discussed in various places in the early 1990s. Katz and Trevisan [6] were the first to provide a formal definition of LDCs and prove lower bounds on their length. Further work on locally decodable codes includes [2, 7, 11]. The length of optimal 2-query LDCs was settled by Kerenidis and de Wolf in [7] and is $\exp(n)$. The length of optimal 3-query LDCs is unknown. The best upper bound prior to our work was $\exp\left(n^{1/2}\right)$ due to Beimel et al., and the best lower bound is $\tilde{\Omega}(n^2)$ [7, 14]. For general (constant) q the best upper bound was $\exp\left(n^{O(\log\log q/(q \log q))}\right)$ due to Beimel et al. [2] and the best lower bound is $\tilde{\Omega}\left(n^{1+1/(\lceil q/2 \rceil - 1)}\right)$ [7, 14].

3. Our results

We give new families of locally decodable codes whose length is vastly shorter than that of previous constructions. We show that every Mersenne prime p (i.e., a prime of the form $p = 2^t - 1$) yields a family of three query locally decodable codes of length $\exp\left(n^{1/t}\right)$. The largest Mersenne prime known currently has $t = 32,582,657 > 10^7$. Substituting this prime into our theorem we conclude that for every n there exists a three query locally decodable code of length $\exp\left(n^{1/32,582,657}\right)$.

It has often been conjectured that the number of Mersenne primes is infinite. If indeed this conjecture holds, our constructions yield three query locally decodable codes of length $N = \exp\left(n^{O\left(\frac{1}{\log\log n}\right)}\right)$ *for infinitely many* n. Finally, assuming that the conjecture of Lenstra, Pomerance and Wagstaff [12, 9, 13] regarding the density of Mersenne primes holds, our constructions yield three query locally decodable codes of length $N = \exp\left(n^{O\left(\frac{1}{\log^{1-\epsilon}\log n}\right)}\right)$ *for all* n, for every $\epsilon > 0$.

4. Our technique

All previously known constructions of locally decodable codes and private information retrieval schemes are (implicitly or explicitly) centered around the idea of representing a message x by an evaluation of a certain low degree polynomial over a finite field. Our constructions take a completely different approach. We start by reducing the problem of constructing locally decodable codes to the problem of designing certain families of sets with restricted intersections. We use elementary algebra over finite fields to design such families.

The heart of our construction is the design of a set $S \subseteq \mathbb{F}_p^*$ for a prime p that simultaneously satisfies two properties: (1) There exist two large sequences of vectors $u_1, \ldots, u_n$, $v_1, \ldots, v_n$ in some low dimensional space $\mathbb{F}_p^m$, such that the dot products $(u_i, v_i) = 0$ for all i, and the dot products $(u_j, v_i) \in S$ for all $i \neq j$. We refer to this property as the combinatorial niceness of S; (2) For a small integer q there exists a q sparse polynomial $\phi(x) \in \mathbb{F}_2[x]$ such that the common GCD of all polynomials of the form $\phi(x^\beta)$, $\beta \in S$ and the polynomial $x^p - 1$ is non-trivial. We refer to this property as the algebraic niceness of S. Our notion of combinatorial niceness is related to the notion of set families with restricted intersections in [1].

Our construction of locally decodable codes thus comes in three steps: First we show that a set S exhibiting both combinatorial and algebraic niceness leads to good locally decodable codes. In particular the length n of the sequences $u_1, \ldots, u_n$ and $v_1, \ldots, v_n$ corresponds to the number of message bits we can encode, while the length of the codewords we build is $N = p^m$. So the longer the sequence and the smaller the dimension the better. The query complexity of our codes is given by the parameter q from the definition of algebraic niceness of S. This step of our construction is quite general and applies to vectors $u_1, \ldots, v_n$ and subsets S over any field. It leads us to the task of identifying good sets that are both combinatorially and algebraically nice, and these tasks narrow our choice of fields. As our second step we focus on combinatorial niceness. In general big sets tend to be "nicer" (allow longer sequences) than small ones. We show that every multiplicative subgroup of a prime field is combinatorially as nice as its cardinality would allow. This still leaves us with a variety of fields and subsets to work with. Finally as the last step we attempt to understand the algebraic niceness of sets. We focus on the very narrow case of Mersenne primes p and the subgroup generated by the element 2 in $\mathbb{F}_p^*$. We manage to show that this subgroup is nice enough to get 3-query locally decodable codes, leading to our final result. A formal treatment of our constructions can be found in [15].

References

[1] L. Babai and P. Frankl, *Linear algebra methods in combinatorics.* 1998.

[2] A. Beimel, Y. Ishai, E. Kushilevitz, and J. F. Raymond. *Breaking the $O\left(n^{1/(2k-1)}\right)$ barrier for information-theoretic private information retrieval,* In Proc. of the 43rd IEEE Symposium on Foundations of Computer Science (FOCS), pp. 261-270, 2002.

[3] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan. *Private information retrieval,* In Proc. of the 36rd IEEE Symposium on Foundations of Computer Science (FOCS), pp. 41-50, 1995. Also, in Journal of the ACM, **45**, 1998.

[4] Curtis Cooper and Steven Boone, http://www.mersenne.org/32582657.htm

[5] W. Gasarch, *A survey on private information retrieval,* The Bulletin of the EATCS, **82**, pp. 72-107, 2004.

[6] J. Katz and L. Trevisan, *On the efficiency of local decoding procedures for error-correcting codes,* In Proc. of the 32th ACM Sym. on Theory of Computing (STOC), pp. 80-86, 2000.

[7] I. Kerenidis and R. de Wolf, *Exponential lower bound for 2-query locally decodable codes via a quantum argument,* Journal of Computer and System Sciences, **69**, pp. 395-420. Earlier version in STOC'03. quant-ph/0208062.
[8] L. Murata and C. Pomerance, *On the largest prime factor of a Mersenne number,* Number theory, CRM Proc. Lecture Notes of American Mathematical Society, **36**, pp. 209-218, 2004.
[9] C. Pomerance, *Recent developments in primality testing,* Math. Intelligencer, **3**, pp. 97-105, (1980/81).
[10] L. Trevisan, "Some applications of coding theory in computational complexity," *Quaderni di Matematica,* 13:347-424, 2004.
[11] S. Wehner and R. de Wolf, *Improved lower bounds for locally decodable codes and private information retrieval,* In Proc. of 32nd International Colloquium on Automata, Languages and Programming (ICALP'05), LNCS 3580, pp. 1424-1436.
[12] Lenstra-Pomerance-Wagstaff conjecture. (2006, May 22). In Wikipedia, The Free Encyclopedia.
[13] S. Wagstaff, *Divisors of Mersenne numbers,* Math. Comp., **40**, pp. 385-397, 1983.
[14] D. Woodruff, *New lower bounds for general locally decodable codes,* Electronic Colloquium on Computational Complexity, Technical report TR07-006, 2007.
[15] S. Yekhanin, *Towards 3-query locally decodable codes of subexponential length,* In Proc. of the 39th ACM Symposium on Theory of Computing (STOC), pp. 266-274, 2007.

Cryptography with Constant Input Locality

Benny Applebaum
(joint work with Yuval Ishai, Eyal Kushilevitz)

We study the following natural question: Which cryptographic primitives (if any) can be realized by functions with constant input locality, namely functions in which every bit of the *input* influences only a constant number of bits of the output? This continues the study of cryptography in low complexity classes. It was recently shown [1] that, under standard cryptographic assumptions, most cryptographic primitives can be realized by functions with constant *output* locality, namely ones in which every bit of the *output* is influenced by a constant number of bits from the input.

We (almost) characterize what cryptographic tasks can be performed with constant input locality. On the negative side, we show that primitives which require some form of non-malleability (such as digital signatures, message authentication, or non-malleable encryption) *cannot* be realized with constant input locality. On the positive side, assuming the intractability of certain problems from the domain of error correcting codes (namely, hardness of decoding a random linear code or the security of the McEliece cryptosystem), we obtain new constructions of one-way functions, pseudorandom generators, commitments, and semantically-secure public-key encryption schemes whose input locality is constant. Moreover, these constructions also enjoy constant *output locality.* Therefore, they give rise to cryptographic hardware that has constant-depth, constant fan-in and constant *fan-out.* As a byproduct, we obtain a pseudorandom generator whose output and input locality are both optimal (namely, 3).

References

[1] B. Applebaum, Y. Ishai, and E. Kushilevitz. Cryptography in NC^0. *SIAM J. Comput.*, 36(4):845–888, 2006. Preliminary version in Proc. 45th FOCS, 2004.

Lower Bounds on Signatures From Symmetric Primitives

Boaz Barak

(joint work with Mohammad Mahmoody-Ghidary)

We show that every black-box construction of one-time signature schemes from a random oracle achieves security at most $poly(q)2^q$, where q is the total number of queries to the oracle by the generation, signing, and verification algorithms. That is, any such scheme can be broken with probability close to 1 by a (computationally unbounded) adversary making $poly(q)2^q$ queries to the oracle. This is tight up to a constant factor in the number of queries, since a simple modification of Lamport's scheme [2] achieves $2^{(2/3-o(1))q}$ security using q queries. Our result provides the first lower bound on the efficiency of constructing signature schemes using a random oracle. A previous result by Gennaro et al [1] gave a lower bound on such constructions using (highly non-random) one-way functions.

Our results extend (with a loss of a constant factor in the number of queries) also to the random permutation and ideal-cipher oracles, and so can be taken as evidence of an inherent efficiency gap between signature schemes and symmetric primitives such as block ciphers, hash functions, and message authentication codes.

References

[1] R. Gennaro, Y. Gertner, J. Katz, and L. Trevisan. Bounds on the Efficiency of Generic Cryptographic Constructions. *SICOMP: SIAM Journal on Computing*, 35, 2005. Preliminary versions in FOCS' 00 and STOC' 03.

[2] L. Lamport. Constructing Digital Signatures from a One-Way Function. Technical Report CSL-98, SRI International, Oct. 1979.

Lower Bounds for Randomized Read/Write Stream Algorithms

Paul Beame

(joint work with T. S. Jayram, Atri Rudra)

Motivated by the capabilities of modern storage architectures, we consider the following generalization of the data stream model where the algorithm has sequential access to multiple streams. Unlike the data stream model, where the stream is *read only*, in this new model (introduced in [1, 2]) the algorithms can also *write* onto streams. There is no limit on the size of the streams but the number of passes made on the streams is restricted. On the other hand, the amount of internal memory used by the algorithm is scarce, similar to data stream model.

We resolve the main open problem in [3] of proving lower bounds in this model for algorithms that are allowed to have *2-sided error*. Previously, such lower bounds were shown only for deterministic and 1-sided error randomized algorithms [2, 3]. We consider the classical *set disjointness* problem that has proved to be invaluable for deriving lower bounds for many other problems involving data streams and other randomized models of computation. For this problem, we show a near-linear lower bound on the size of the internal memory used by a randomized algorithm with 2-sided error that is allowed to have $o(\log N/\log\log N)$ passes over the streams. This bound is almost optimal since there is a simple algorithm that can solve this problem using logarithmic memory if the number of passes over the streams is allowed to be $O(\log N)$.

Applications include near-linear lower bounds on the internal memory for well-known problems in the literature: (1) approximately counting the number of distinct elements in the input (F_0); (2) approximating the frequency of the *mode* of an input sequence (F^*_∞); (3) computing the join of two relations; and (4) deciding if some node of an XML document matches an XQuery (or XPath) query.

Our techniques involve a novel direct-sum type of argument that yields lower bounds for many other problems. Our results asymptotically improve all previously known bounds for problems in the read/write streams model even in deterministic and 1-sided error models of computation.

References

[1] M. Grohe, C. Koch, and N. Schweikardt. *Tight lower bounds for query processing on streaming and external memory data external memory*, Proceedings of the 32nd International Colloquium on Automata, Languages and Programming (ICALP), LNCS3580 (2005), 1076–1088.

[2] M. Grohe and N. Schweikardt, *Lower bounds for sorting with few random accesses to external memory*, Proceedings of the 24th ACM Symposium on Principles of Database Systems (PODS) (2005), 238–249.

[3] M. Grohe, A. Hernich, and N. Schweikardt, *Randomized computations on large data sets: Tight lower bounds*, Proceedings of the 25th ACM Symposium on Principles of Database Systems (PODS) (2006), 243–252.

On the complexity of graph polynomials

Markus Bläser

(joint work with Holger Dell, Mahmoud Fouz, Christian Hoffmann, Johann A. Makowsky)

A graph polynomial P maps graphs to polynomials P_G over some ring R such that isomorphic graphs are mapped to the same polynomial. If we now fix some point ξ and map G to $P_G(\xi)$, we get a new graph invariant that maps graphs to elements of R. We deal with the following question: What is the complexity of this mapping in dependence on ξ? A famous result by Jaeger, Vertigan and Welsh [2] says that for almost all points (in the Zariski sense), it is #P-hard to evaluate the Tutte polynomial at this point. We show a similar result for cover polynomial, an equivalent of the Tutte polynomial for directed graphs (see also [1]), and for the interlace polynomial. Since all these polynomials are definable in monadic second order logic, this gives rise to the following "difficult point conjecture", cf. [3]: For every graph polynomial that is definable in monadic second order logic and that is #P-hard to evaluate for at least one point, its evaluation is #P-hard for almost all points. Finally, we propose that one should study graph polynomials in an algebraic model a la Blum, Shub, and Smale, since most graph polynomials are defined over $\mathbb{C}$. Prior studies usually looked only at fields that have discrete representations like algebraic field extensions of $\mathbb{Q}$.

References

[1] Markus Bläser, Holger Dell, *Complexity of the cover polynomial*, Proc. 34th Int. Coll. on Automata, Languages, and Programming (ICALP), Springer Lecture Notes in Computer Science 4596 (2007).

[2] F. Jaeger, D.L. Vertigan, D.J. Welsh, *On the computational complexity of the Jones and Tutte polynomials*, Mathematical Proceedings of the Cambridge Philosophical Society , **108** (1990), 35–53.

[3] Johann A. Makowsky, *From a zoo to a zoologoy: towards a general theory of graph polynomials*, Theory of Computing Systems, to appear.

Sampling methods for shortest vectors, closest vectors and successive minima

Johannes Blömer

(joint work with Stefanie Naewe)

We study four problems from the geometry of numbers, the *shortest vector problem* (Svp), the *closest vector problem* (Cvp), the *successive minima problem* (Sivp), and the *shortest independent vectors problem* (Sivp). Extending and generalizing results of Ajtai, Kumar, and Sivakumar we present probabilistic single exponential time algorithms for all four problems for all ℓ_p norms. The results on Smp and Sivp are new for all norms. The results on Svp and Cvp generalize previous results of Ajtai et al. for the Euclidean ℓ_2 norm to arbitrary ℓ_p norms. We achieve our results by introducing a new lattice problem, the *subspace avoiding problem* (Sap). We describe a single exponential time algorithm for Sap. We also describe polynomial time reductions from Svp, Cvp, Smp, and Sivp to Sap, establishing the single exponential time algorithm for the four classical lattice problems. This approach leads to a unified algorithmic treatment of the lattice problems Svp, Cvp, Smp, and Sivp.

Pseudorandom generators for low degree polynomials

Andrej Bogdanov

(joint work with Emanuele Viola)

We present a new approach to constructing pseudorandom generators that fool low-degree polynomials over finite fields, based on the Gowers norm. Using this approach, we obtain the following main constructions of explicitly computable generators $G : \mathbb{F}^s \to \mathbb{F}^n$ that fool polynomials over a prime field $\mathbb{F}$:

(1) a generator that fools degree-2 polynomials to within error $1/n$, with seed length $s = O(\log n)$,
(2) a generator that fools degree-3 polynomials to within error ϵ, with seed length $s = 3 \cdot \log_{|\mathbb{F}|} n + f(\epsilon, \mathbb{F})$ where f depends only on ϵ and $\mathbb{F}$,
(3) assuming the "Gowers inverse conjecture," a generator that fools degree-d polynomials to within error ϵ, with seed length $s = d \cdot \log_{|\mathbb{F}|} n + f(d, \epsilon, \mathbb{F})$ where f depends only on d, ϵ, and $\mathbb{F}$.

The results in (1) and (2) are unconditional, i.e. do not rely on any unproven assumption. Moreover, the results in (3) rely on a special case of the conjecture which may be easier to prove.

Our generator for degree-d polynomials is the component-wise sum of d generators for degree-1 polynomials (on independent seeds).

Prior to our work, generators with logarithmic seed length were only known for degree-1 polynomials [NN90]. In fact, over small fields such as $\mathbb{F}_2 = \{0, 1\}$, our results constitute the first progress on these problems since the celebrated generator by Luby, Veličković and Wigderson [LVW93], whose seed length is much bigger: $s = \exp\big(\Omega(\sqrt{\log n})\big)$, even for the case of degree-2 polynomials over $\mathbb{F}_2$.

References

[NN90] J. Naor and M. Naor. Small-bias probability spaces: efficient constructions and applications. In *Proceedings of the 22nd Annual ACM Symposium on the Theory of Computing*, pages 213–223, 1990.

[LVW93] Michael Luby, Boban Veličković, and Avi Wigderson. Deterministic approximate counting of depth-2 circuits. In *Proceedings of the 2nd Israeli Symposium on Theoretical Computer Science (ISTCS)*, pages 18–24, 1993.

On defining integers and proving arithmetic circuit lower bounds

Peter Bürgisser

Let $\tau(n)$ denote the minimum number of arithmetic operations sufficient to build the integer n from the constant 1. We prove that if there are arithmetic circuits of size polynomial in n for computing the permanent of n by n matrices, then $\tau(n!)$ is polynomially bounded in $\log n$. Under the same assumption on the permanent, we conclude that the Pochhammer-Wilkinson polynomials $\prod_{k=1}^{n}(X-k)$ and the Taylor approximations $\sum_{k=0}^{n} \frac{1}{k!} X^k$ and $\sum_{k=1}^{n} \frac{1}{k} X^k$ of exp and log, respectively, can be computed by arithmetic circuits of size polynomial in $\log n$ (allowing divisions). This connects several so far unrelated conjectures in algebraic complexity.

An extended abstract of this work appeared in *Proc. 24th International Symposium on Theoretical Aspects of Computer Science* (STACS 2007), Lecture Notes in Computer Science 4393, pp. 133–144, Springer, 2007. The full paper is accepted for *Computational Complexity.*

References

[1] P.W. Beame, S.A. Cook, and H.J. Hoover. Log depth circuits for division and related problems. *SIAM J. Comput.*, 15(4):994–1003, 1986.
[2] P. Bürgisser. *Completeness and Reduction in Algebraic Complexity Theory*, volume 7 of *Algorithms and Computation in Mathematics*. Springer Verlag, 2000.
[3] P. Bürgisser, M. Clausen, and M.A. Shokrollahi. *Algebraic Complexity Theory*, volume 315 of *Grundlehren der mathematischen Wissenschaften*. Springer Verlag, 1997.
[4] J. von zur Gathen and V. Strassen. Some polynomials that are hard to compute. *Theoret. Comp. Sci.*, 11:331–336, 1980.
[5] W. Hesse, E. Allender, and D.A. Barrrington. Uniform constant-depth threshold circuits for division and iterated multiplication. *J. Comput. System Sci.*, 65(4):695–716, 2002. Special issue on complexity, 2001 (Chicago, IL).
[6] P. Koiran. Valiant's model and the cost of computing integers. *Comput. Complexity*, 13(3-4):131–146, 2004.
[7] G. Malod. *Polynômes et coefficients*. Phd thesis, Université Claude Bernard - Lyon 1, 2003. `http://tel.ccsd.cnrs.fr/tel-00087399`.
[8] J.H. Reif and S.R. Tate. On threshold circuits and polynomial computation. *SIAM J. Comput.*, 21(5):896–908, 1992.
[9] M. Shub and S. Smale. On the intractability of Hilbert's Nullstellensatz and an algebraic version of "NP $\neq$ P?". *Duke Math. J.*, 81:47–54, 1995.
[10] S. Smale. Mathematical problems for the next century. In *Mathematics: frontiers and perspectives*, pages 271–294. Amer. Math. Soc., Providence, RI, 2000.
[11] V. Strassen. Polynomials with rational coefficients which are hard to compute. *SIAM J. Comp.*, 3:128–149, 1974.
[12] J. Torán. Complexity classes defined by counting quantifiers. *J. Assoc. Comput. Mach.*, 38(3):753–774, 1991.
[13] L.G. Valiant. Completeness classes in algebra. In *Proc. 11th ACM STOC*, pages 249–261, 1979.
[14] L.G. Valiant. Reducibility by algebraic projections. In *Logic and Algorithmic: an International Symposium held in honor of Ernst Specker*, volume 30, pages 365–380. Monogr. No. 30 de l'Enseign. Math., 1982.
[15] K.W. Wagner. The complexity of combinatorial problems with succinct input representation. *Acta Inform.*, 23(3):325–356, 1986.

On subexponentiality of the discrete logarithm problem in elliptic curves over extension fields

Claus Diem

Let us consider the discrete logarithm problem in the groups of rational points of elliptic curves over finite fields: Given an elliptic curve $E/\mathbb{F}_q$ and two points $P, Q \in E(\mathbb{F}_q)$ such that $Q \in \langle P \rangle$, find some $x \in \mathbb{N}$ with $Q = x \cdot P$!

We are concerned with the complexity of this algorithmic problem as a function of $\log(q)$. (Note that $\#E(\mathbb{F}_q) \sim q$, and the input length (of an appropriate representation) is in $\Theta(\log(q))$.)

The problem can obviously be solved in a running time of $\tilde{\mathcal{O}}(q)$ by "brute-force" on a Turing machine (or on a random access machine). No (randomized) algorithm is however known which solves the problem in subexponential time. (On a random access machine the best known running time is $\tilde{\mathcal{O}}(e^{\log(q)/2})$; this result follows from

Shanks' Baby-Step-Giant-Step algorithm which applies to the discrete logarithm problem in every finite group.)

This motivates the following tasks:

Find families of elliptic curves over finite fields such that the discrete logarithm problem can be solved in an expected time which is

- subexponential, that is, in $\mathcal{O}(q^{o(1)})$
- bounded by a subexponentiality function, that is, bounded by a function of the form $e^{\log(q)^{\alpha}}$ for some $\alpha < 1$.

Our result is as follows:

Theorem 1. *There exists some $c > 0$ such that the following holds: The discrete logarithm problem in $E(\mathbb{F}_{q^n})$, where $n \leq c\log(q)$ and E is any elliptic curve over $\mathbb{F}_{q^n}$, can be solved an an expected time which is polynomial in q.*

Corollary 2. *Let c be as above. Then the discrete logarithm problem in $E(\mathbb{F}_{q^n})$, where $\frac{1}{2}\log(q)^c \leq n \leq \log(q)^c$ and E is any elliptic curve over $\mathbb{F}_{q^n}$, can be solved in an expected time which is polynomial in $e^{\log(q^n)^{\frac{1}{1+c}}}$.*

The corollary follows from the theorem because under the assumptions of the corollary $q = e^{(\log(q)^c \log(q))^{\frac{1}{1+c}}} \leq e^{(2n\log(q))^{\frac{1}{1+c}}}$.

Previously some families of elliptic curves for which the discrete logarithm problem is subexponential where already known. Also, using the so-called GHS attack one can prove (unpublished) that there exists a sequence of finite fields (of strictly increasing size) such that the discrete logarithm problem in all elliptic curves over these fields is subexponential. It was however not known if there exists a family of finite fields (of strictly increasing size) such that the discrete logarithm problem in all elliptic curves over these fields is bounded by a subexponentiality function. (See definitions above for the distinction of these two questions).

The algorithm used for the proof of the theorem is essentially an algorithm given by P. Gaudry (P. Gaudry: Index calculus for abelian varieties and the elliptic curve discrete logarithm problem, preprint). As the title indicates, it is an algorithm of "index calculus" type. The relations are thereby collected by solving systems of multivariate polynomial equations over $\mathbb{F}_q$. The main difficulty of the proof of the theorem relies in analyzing the algorithm for varying extension degrees n.

Extractors and Rank Extractors for Polynomial Sources

Zeev Dvir

(joint work with Ariel Gabizon, Avi Wigderson)

In this work we construct explicit deterministic extractors from *polynomial sources*, namely from distributions sampled by low degree multivariate polynomials over finite fields. This naturally generalizes previous work on extraction from affine sources (which are degree 1 polynomials) [BKSSW05, Bou05, GR05]. A direct consequence is a deterministic extractor for distributions sampled by polynomial size arithmetic circuits over exponentially large fields.

The steps in our extractor construction, and the tools (mainly from algebraic geometry) that we use for them, are of independent interest:

The first step is a construction of *rank extractors*, which are polynomial mappings which "extract" the algebraic rank from any system of low degree polynomials. More precisely, for any n polynomials, k of which are algebraically independent, a rank extractor outputs k algebraically independent polynomials of slightly higher degree. The rank extractors we construct are applicable not only over finite fields but also over fields of characteristic zero.

The next step is relating algebraic independence to min-entropy. We use a theorem of Wooley to show that these parameters are tightly connected. This allows replacing the algebraic assumption on the source (above) by the natural information theoretic one. It also shows that a rank extractor is already a high quality *condenser* for polynomial sources over polynomially large fields.

Finally, to turn the condensers into extractors, we employ a theorem of Bombieri, giving a character sum estimate for polynomials defined over curves. It allows extracting all the randomness (up to a multiplicative constant) from polynomial sources over exponentially large fields.

References

[BKSSW05] Boaz Barak, Guy Kindler, Ronen Shaltiel, Benny Sudakov, and Avi Wigderson. Simulating independence: new constructions of condensers, ramsey graphs, dispersers, and extractors. In *STOC '05: Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 1–10, New York, NY, USA, 2005. ACM Press.

[Bou05] J. Bourgain. On the construction of affine extractors. *To Appear in GAFA*, 2005.

[GR05] A. Gabizon and R. Raz. Deterministic extractors for affine sources over large fields. In *FOCS '05: Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 407–418, Washington, DC, USA, 2005. IEEE Computer Society.

Counting reducible and singular bivariate polynomials

Joachim von zur Gathen

We investigate four "accidents" that can happen to a bivariate polynomial over a finite field: it can have a nontrivial factor, or a square factor, or a factor over an extension field (but none over the ground field), or a singular root, where all partial derivatives also vanish. The main results are quantitative versions of the intuition that a random polynomial is unlikely to suffer an accident.

In the set $B_n(F) \subseteq F[x,y]$ of bivariate polynomials with total degree at most n, we have sets $A_n(F) \subseteq B_n(F)$ of such "accidents". We have geometric and combinatorial results, namely bounds on the (minimal) codimension of A_n in B_n (over an algebraically closed field), or functions $\alpha_n(q)$ and $\beta_n(q)$ so that

$$\left| \frac{\# A_n(\mathbb{F}_q)}{\# B_n(\mathbb{F}_q)} - \alpha_n(q) \right| \leq \alpha_n(q) \cdot \beta_n(q).$$

The following hold for large enough n.

accident	codim	$\alpha_n(q)$	$\beta_n(q)$
reducible	$n-1$	$(q+1)q^{-n}$	$2q^{-n+3}$
squareful	$2n-1$	$\frac{q^{-2n+1}(1+q^{-1})(1-q^{-n+1})}{1-q^{-n-1}}$	$6q^{-2n+6}$
rel. irreducible		$\varepsilon_n(q)$	$2q^{-n+l+1}$
singular	1	$(1-q^{-3})^{q^2}$	0

In the third line, we use l for the largest prime divisor of n, and

$$\varepsilon_n(q) = \frac{q^{-n^2(l-1)/2l}(1-q^{-1})}{l(1-q^{-l})(1-q^{-n-1})}.$$

Previous work on this question includes Carlitz (1963, 1965), Cohen (1968, 1970), Wan (1992), Ragot (1997, 1999), Gao & Lauder (2002), Bodin (2006). Ragot's results were improved by Hendrik Lenstra. An Extended Abstract appears in the Proc. ISSAC'07.

References

Arnaud Bodin (2006). Number of irreducible polynomials in two variables over finite fields. Preprint, 7 July 2006.

Leonard Carlitz (1963). The distribution of irreducible polynomials in several indeterminates. *Illinois Journal of Mathematics* **7**, 371–375.

Leonard Carlitz (1965). The distribution of irreducible polynomials in several indeterminates II. *Canadian Journal of Mathematics* **17**, 261–266.

S. D. Cohen (1970). The Distribution of Polynomials over Finite Fields. *Acta Arithmetica* **17**, 255–271.

Stephen Cohen (1968). The distribution of irreducible polynomials in several indeterminates over a finite field. *Proceedings of the Edinburgh Mathematical Society* **16**, 1–17.

Shuhong Gao & Alan G. B. Lauder (2002). Hensel Lifting and Bivariate Polynomial Factorisation over Finite Fields. *Mathematics of Computation* **71**(240), 1663–1676.

Jean-François Ragot (1997). *Sur la factorisation absolue des polynômes.* Thèse, Université de Limoges. URL `http://www.unilim.fr/laco/theses/1997/T1997_02.pdf`. 133 pages.

Jean-François Ragot (1999). Counting polynomials with zeros of given multiplicities in finite fields. *Finite Fields and Their Applications* **5**, 219–231.

Daqing Wan (1992). Zeta Functions of Algebraic Cycles over Finite Fields. *Manuscripta Mathematica* **74**, 413–444.

On the Approximation Resistance of a Random Predicate

Johan Håstad

Consider a predicate P which takes as input k Boolean variables and outputs true/false. Suppose P accepts t_P of the 2^k possible input strings.

For the Max-CSP connected with P an instance is given by a k-tuple of literals. For each assignment we can observe the number of k-tuples of Boolean variables that satisfy P and the goal is to maximize this number.

This problem is NP-hard for almost all P and we look at algorithms that approximate this number. There is a natural algorithm that approximates this number within $t_P 2^{-k}$ which simply picks a random assignment. We say that a predicate is *approximation resistant* if it is hard to getter a significantly stronger approximation guarantee. To be more precise P is approximation resistant if, for any $\epsilon > 0$, it is NP-hard to approximate the maximal number of simultaneously satisfiable constraints within a factor $t_P 2^{-k} + \epsilon$.

We prove that, assuming the unique games conjecture [1], for sufficiently large k a random predicate is approximation resistant with high probability.

The result builds on a recent result by Samorodnitsky and Trevisan [2] that if 2^d is the smallest power of two larger than k, there is predicate P_{ST} of width k that only accepts 2^d strings and is approximation resistant.

Our proof shows that any predicate implied by P_{ST}, or a predicate obtained from P_{ST} by permuting the inputs and negating some input bits also is approximation resistant. Using a second moment method we then show that this criteria applies to a random predicate.

This paper will be published at the Approx07-conference to be held in August 2007.

References

[1] S. Khot, *On the power of unique 2-Prover 1-Round games*, Proceedings of 34th ACM Symposium on Theory of Computation, 2002, pp 767-775.

[2] A. Samorodnitsky and L. Trevisan, *Gowers Uniformity, Influence of Variables and PCPs*, Proceedings of the 38th Annual ACM Symposium on Theory of Computing, 2006, pp 11-20.

The Black-Box Query Complexity of Polynomial Summation

Valentine Kabanets

(joint work with Ali Juma, Charles Rackoff, Amir Shpilka)

For any given Boolean formula $\phi(x_1, \ldots, x_n)$, one can efficiently construct (using arithmetization) a low-degree polynomial $p(x_1, \ldots, x_n)$ that agrees with ϕ over all points in the Boolean cube $\{0,1\}^n$; the constructed polynomial p can be interpreted as a polynomial over an arbitrary field $\mathbb{F}$. The problem $\#SAT$ (of counting the number of satisfying assignments of ϕ) thus reduces to the polynomial summation $\sum_{x \in \{0,1\}^n} p(x)$. Motivated by this connection, we study the *query complexity* of the polynomial summation problem: Given (oracle access to) a polynomial $p(x_1, \ldots, x_n)$, compute $\sum_{x \in \{0,1\}^n} p(x)$. Obviously, querying p at all 2^n points in $\{0,1\}^n$ suffices. Is there a field $\mathbb{F}$ such that, for every polynomial $p \in \mathbb{F}[x_1, \ldots, x_n]$, the sum $\sum_{x \in \{0,1\}^n} p(x)$ can be computed using fewer than 2^n queries from $\mathbb{F}^n$? We show that the simple upper bound 2^n is in fact *tight* in the *black-box model* where one has only *oracle access* to the polynomial p, for any field $\mathbb{F}$. We prove these lower bounds for the *adaptive* query model, where the next query can depend on the values of p at previously queried points. Our lower bounds hold even for polynomials that have degree at most 2 in each variable. In contrast, for polynomials that have degree at most 1 in each variable (i.e., multilinear polynomials), we show that a *single* query is sufficient over any field of characteristic other than 2.

On probabilistic analysis of randomization in hybrid symbolic-numeric algorithms

Erich Kaltofen

(joint work with Zhengfeng Yang, Lihong Zhi)

Algebraic randomization techniques can be applied to hybrid symbolic-numeric algorithms, that is, algebraic algorithms where the scalars in the inputs have numerical errors. We consider the problem of solving highly over- and underdetermined systems of linear equations by essentially optimal randomized algorithms (e.g., solving a linear system with n equations and $p = O((n\log(n))^{0.72})$ variables in $O(pn\log(n))$ field operations) and interpolating a sparse rational multivariate function from noisy values. We show that Zippel's original sparse polynomial interpolation technique applies to numerically perturbed data and we give an exact and hybrid algorithm for interpolating sparse rational functions. We discuss the expected condition numbers of the arising randomized linear systems, and observe that certain randomized projections can lead to ill-conditioned systems [1].

References

[1] Erich Kaltofen, Zhengfeng Yang, and Lihong Zhi. On probabilistic analysis of randomization in hybrid symbolic-numeric algorithms. In *Proc. Internat. Workshop on Symbolic-Numeric Comput. 2007* [2], 11–17.

[2] Jan Verschelde and Stephen Watt, editors. *Proc. Internat. Workshop on Symbolic-Numeric Comput. 2007*, New York, N. Y., 2007. ACM.

Understanding parallel repetition requires understanding foams

Guy Kindler

(joint work with Uri Feige, Ryan O'Donnell)

The parallel repetition theorem, proven by Raz in 1995, is a fundamental result that in addition to its philosophical appeal, plays a key role in complexity theory. The theorem studies the behavior of success probabilities of two prover games, when many copies of such a game are played in parallel. It shows that the success probability decreases exponentially in the number of repetitions, but the parameters given by the theorem do not seem tight. It is natural to ask what are the best parameters for which the theorem holds, and improving them would have complexity theoretic implications.

This talk describes an attempt to improve the parameters in a very special case of the parallel repetition theorem. Our attempt had only limited success, but it turns out that the reason we got stuck was that the following seemingly hard question from the geometry of "foams" was hidden in the special case that we were trying to solve: What is the least surface area of a cell that tiles $\mathbb{R}^d$ by the lattice $\mathbb{Z}^d$? Very little about this foam problem is known. It is interesting to see such a geometric question encoded inside the problem of parallel repetition in two prover games.

Interpolation in Valiant's theory

Pascal Koiran
(joint work with Sylvain Périfel)

The starting point of this work is a question raised by Christos Papadimitriou in a personal communication to Erich Kaltofen:

Question (*)

> If a multivariate polynomial P is computable by a (boolean) polynomial-time algorithm on rational inputs, does that imply that P can be computed by a polynomial-size arithmetic circuit? In such a circuit, the only allowed operations are additions, subtractions, and multiplications.

As pointed out by Papadimitriou, Strassen's "Vermeidung von Divisionen" shows that that for evaluating a low-degree polynomial P, divisions would not increase exponentially the power of arithmetic circuits. It is indeed a natural question whether, more generally, all boolean operations can be replaced efficiently by additions, subtractions and multiplications.

In my talk I explained why it seems difficult to answer this question either way. Obtaining a positive answer seems difficult because it would imply the following transfer theorem: $\mathsf{FP} = \sharp\mathsf{P} \Rightarrow \mathsf{VP} = \mathsf{VNP}$ (assuming that $\mathsf{FP} = \sharp\mathsf{P}$, the permanent must be in FP; a positive answer to question (*) would therefore imply that the permanent is in VP, and that $\mathsf{VP} = \mathsf{VNP}$ by completeness of the permanent). Unfortunately, in spite of all the work establishing close connections between the boolean model of computation and the algebraic models of Valiant and of Blum, Shub and Smale no such transfer theorem is known. In fact, we do not know of any hypothesis from boolean complexity theory that would imply the equality $\mathsf{VP} = \mathsf{VNP}$ (but transfer theorems in the opposite direction were established [1]).

A natural strategy for obtaining a negative answer to question (*) would be to exhibit a family of polynomials that are easy to evaluate on rational inputs but hard to evaluate by arithmetic circuits. Unfortunately, there seems to be a lack of candidate polynomials. Another difficulty is that a negative answer would imply the separation of the algebraic complexity classes VP^0 and VNP^0. This observation is our main contribution to the study of question (*). The classes VP^0 and VNP^0 are constant-free versions of the classes VP (of "easily computable polynomial families") and VNP (of "easily definable polynomial families") defined by Valiant. The separation $\mathsf{VP}^0 \neq \mathsf{VNP}^0$ seems very plausible, but it also seems very difficult to establish.

The full paper will contain a few additional results.

References

[1] P. Bürgisser. *Completeness and Reduction in Algebraic Complexity Theory.* Number 7 in Algorithms and Computation in Mathematics. Springer, 2000.

Subspace polynomials and List Decoding of Reed-Solomon Codes

Swastik Kopparty

(joint work with Eli Ben-Sasson, Jaikumar Radhakrishnan)

We show combinatorial limitations on efficient list decoding of Reed-Solomon codes beyond the Johnson and Guruswami-Sudan bounds [Joh62, GS99]. In particular, we show that for arbitrarily large fields $\mathbb{F}_N, |\mathbb{F}_N| = N$, for any $\delta \in (0,1)$, and $K = N^\delta$:

- **Existence:** there exists a received word $w_N : \mathbb{F}_N \to \mathbb{F}_N$ that agrees with a super-polynomial number of distinct degree K polynomials on $\approx N^{\sqrt{\delta}}$ points each;
- **Explicit:** there exists a polynomial time constructible received word $w'_N : \mathbb{F}_N \to \mathbb{F}_N$ that agrees with a super-polynomial number of distinct degree K polynomials, on $\approx 2^{\sqrt{\log N}} K$ points each.

In both cases, our results improve upon the previous state of the art, which was $\approx N^\delta/\delta$ points of agreement for the existence case [JH01], and $\approx 2N^\delta$ points of agreement for the explicit one [GR05b]. Furthermore, for δ close to 1 our bound approaches the Guruswami-Sudan bound (which is $\sqrt{NK}$) and implies limitations on extending their efficient Reed-Solomon list decoding algorithm to larger decoding radius.

Our proof method is surprisingly simple. We work with polynomials that vanish on subspaces of an extension field viewed as a vector space over the base field. These *subspace polynomials* are a subclass of *linearized polynomials* that were first studied by Ore [Ore33, Ore34] in the 1930s, and later by coding theorists. For us their main attraction is their sparsity and abundance of roots, virtues that recently won them pivotal roles in *probabilistically checkable proofs of proximity* [BSGH+04, BSS05] and sub-linear proof verification [BSGH+05].

References

[BSGH+04] Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil Vadhan. Robust PCPs of proximity, shorter PCPs and applications to coding. In ACM, editor, *Proceedings of the 36th Annual ACM Symposium on the Theory of Computing: Chicago, Illinois, USA, June 13–15, 2004*, pages 1–10, pub-ACM:adr, 2004. pub-ACM.

[BSGH+05] Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan, and Salil P. Vadhan. Short PCPs verifiable in polylogarithmic time. In *IEEE Conference on Computational Complexity*, pages 120–134, 2005.

[BSS05] Eli Ben-Sasson and Madhu Sudan. Simple PCPs with poly-log rate and query complexity. In *STOC '05: Proceedings of the 37th annual ACM Symposium on Theory of Computing*, pages 266–275, New York, NY, USA, 2005. ACM Press.

[GR05b] Venkatesan Guruswami and Atri Rudra. Limits to list decoding Reed-Solomon codes. In *STOC '05: Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 602–609, New York, NY, USA, 2005. ACM Press.

[GS99] Venkatesan Guruswami and Madhu Sudan. Improved Decoding of Reed-Solomon and Algebraic-Geometric Codes. In *IEEE Transactions on Information Theory*, volume 45, pages 1757–1767, 1999.

[JH01] Jørn Justesen and Tom Høholdt. Bounds on list decoding of MDS codes. *IEEE Trans. Inform. Theory*, 47(4):1604–1609, 2001.

[Joh62] S. M. Johnson. A new upper bound for error-correcting codes. *IEEE Trans. on Information Theory*, 8:203–207, 1962.

[Ore33] O. Ore. On a special class of polynomials. *Trans. Amer. Math. Soc.*, 35(3):559–584, 1933.

[Ore34] O. Ore. Contributions to the theory of finite fields. *Trans. Amer. Math. Soc.*, 36(2):243–274, 1934.

[RS60] I. S. Reed and G. Solomon. Polynomial Codes over Certain Finite Fields. *Journal of Society for Industrial and Applied Mathematics*, 8:300–304, 1960.

[Sud97] Madhu Sudan. Decoding of Reed Solomon Codes beyond the Error-Correction Bound. *Journal of Complexity*, 13(1):180–193, 1997.

Constructing Boolean Functions of Maximal Algebraic Immunity

Matthias Krause

(joint work with Hellen Altendorf, Frederik Armknecht)

We construct boolean functions $f : \{0,1\}^n \longrightarrow \{0,1\}^m$, for which the graph $gr(f) = \{(x, f(x)), x \in \{0,1\}\} \subseteq \{0,1\}^{n+m}$ has maximal algebraic immunity. Hereby, the algebraic immunity $AI(S)$ of a subset S of the boolean cube is defined to be the minimal d for which there is a nontrivial degree-d polynomial (over $GF(2)$) which annihilates S, i.e. which outputs 0 for all $x \in S$. Consequently, if the algebraic immunity of a given boolean function is d then nontrivial relations in the input/output bits of degree smaller than d do not exist.

The study of the algebraic immunity of boolean functions in the context of symmetric cryptography was initiated by Meier, Pasalic, Carlet in [10]. It is motivated by the need for appropriate boolean functions serving as building blocks of symmetric ciphers.Such functions should have large algebraic immunity for preventing vulnerability of the cipher against algebraic attacks. For several practically used cryptosystems, building blocks with low algebraic immunity open the door to express the bits of the secret key by a overdefined system of low degree equations (see, e.g. [2], [6], [7], [8], [13]), which then can be solved by nontrivial methods ([4], [5], [12]).

In [3] we completely solve the problem of constructing explicitely defined single-output functions of maximal algebraic immunity. For even number of input bits it can be easily shown that majority has this property. For odd number of input bits the situation is more complicated.

For multi-output functions no explicite construction of a function family of maximal algebraic immunity is known. We present an efficient algorithm, based on the matroid union algorithm of Edmonds [9], which computes for given m, n, d, if existent, the table of a function $h : \{0,1\}^n \longrightarrow \{0,1\}^m$ of algebraic immunity d. To the best of our knowledge, this is the first systematic method for constructing multi-output functions of high algebraic immunity.

A natural upper bound $d^*(n,m)$ for the algebraic immunity of a boolean function $f : \{0,1\}^n \longrightarrow \{0,1\}^m$ is the minimal number d such that $\sum_{i=0}^{d} \binom{n+m}{i} > 2^n$. We conjecture that for all $1 \leq m \leq n$ there are functions $f : \{0,1\}^n \longrightarrow \{0,1\}^m$ of algebraic immunity $d^*(n,m)$. Experiments show that this is true for all $1 \leq m \leq n \leq 20$ [1]. The proof of this conjecture remains as an open problem.

References

[1] H. Altendorf, *Maximal Immune Funktionen*, Diploma Thesis, Universität Mannheim, 2007.

[2] F. Armknecht, M. Krause, *Algebraic attacks on Combiners with Memory*, Proceedings of Crypto, LNCS **2729** (2003), 162–176.

[3] F. Armknecht, M. Krause, *Constructing Single- and Multi-Output Boolean Functions of Maximal Algebraic Immunity*, Proceedings of ICALP, LNCS **4052**, Part II (2006), 180–191.

[4] G. Ars, G., J. Faugère, H. Imai, M. Kawazoe, M. Sugita, *Comparison Between XL and Gröbner Basis Algorithms.*, Proceedings of Asiacrypt, LNCS **3329** (2004), 338–353.

[5] C. Cid, G. Leurent, *An Analysis of the XSL Algorithm*, Proceedings of Asiacrypt, LNCS **3788** (2005), 333–352.

[6] N. Courtois, J. Pieprzyk, *Cryptanalysis of block ciphers with overdefined systems of equations*, Proceedings of Asiacrypt, LNCS **2501** (2002), 267–287.

[7] N. Courtois, W. Meier, *Algebraic attacks on Stream Ciphers with Linear Feedback*, Proceedings of Eurocrypt, LNCS **2656** (2003), 345–359.

[8] N. Courtois, *Fast Algebraic Attacks on Stream Ciphers with Linear Feedback*, Proceedings of CRYPTO, LNCS **2729** (2003), 176–194.

[9] J. Edmonds, *Matroid Partition* Journal of the AMS **11** (1968), 335–345.

[10] W. Meier, E. Pasalic, C. Carlet, *Algebraic attacks and decomposition of Boolean functions*, Proceeding of Eurocrypt 2004, LNCS **3027** (2004), 474-491.

[11] A. Schrijver, *Combinatorial Optimization. Polyhedra and Efficiency*, Springer (2003).

[12] A. Shamir, J. Patarin, N. Courtois, A. Klimov, *Efficient Algorithms for Solving Overdefined Systems of Multivariate Polynomial Equations*, Proceedings of Eurocrypt, LNCS **1807** (2000), 392–407.

[13] T. Shimoyama, T., T. Kaneko, *Quadratic Relation of S-boxes and Its Application to the Linear Attack of Full Round DES*, Proceedings of Crypto, LNCS **1462** (1998), 200–211.

All Natural NPC Problems Have Average-Case Complete Versions

Noam Livne

In 1984 Levin put forward a suggestion for a theory of *average case complexity* [1]. In this theory a problem, called a *distributional problem*, is defined as a pair consisting of a decision problem and a probability distribution over the instances. Introducing adequate notions of "efficiency-on-average", simple distributions and efficiency-on-average preserving reductions, Levin developed a theory analogous to the theory of NP-completeness. In particular, he showed that there exists a simple distributional problem that is complete under these reductions. But since then very few distributional problems were shown to be complete in this sense. In this paper we show a simple sufficient condition for an NP-complete decision problem to have a distributional version that is complete under these reductions (and thus to be "hard on the average" with respect to some simple probability distribution). Apparently all known NP-complete decision problems meet this condition.

References

[1] Leonid A Levin. Average case complete problems. *SIAM J. Comput.*, 15(1):285–286, 1986.

The probability that a small perturbation of a numerical analysis problem is difficult

Martin Lotz

(joint work with Peter Bürgisser, Felipe Cucker)

The *condition number* of a numerical computation problem measures the sensitivity of the output to small perturbations of the input. Condition numbers occur in many instances of round-off analysis, and they also appear as a parameter in complexity bounds for a variety of iterative algorithms for solving numerical problems. In the work underlying this talk [1], we prove a general result providing smoothed analysis estimates for conic condition numbers. Our probability estimates depend only on geometric invariants of the corresponding sets of ill-posed inputs.

A condition number is *conic* if there exists a semi-algebraic cone $\Sigma \subseteq \mathbb{R}^{p+1}$, the set of *ill-posed inputs*, such that for all input data $a \in \mathbb{R}^{p+1} \setminus \{0\}$, $(a) = \frac{\|a\|}{\mathrm{dist}(a,\Sigma)}$ holds. Since Σ is a cone, we may restrict to data lying in the unit sphere S^p, and then the conic condition number be characterized as the inverse distance to ill-posedness on the sphere. Our main result is the following (in the statement, $z \in B(a,\sigma)$ means that z is uniformly distributed in a spherical cap of radius $\arcsin \sigma$ around a).

Theorem 1. *Let be a conic condition number with set of ill-posed inputs Σ, and assume Σ is contained in the zero set of homogeneous polynomials of degree at most d. Then, for all $\sigma \in (0,1]$ and all $t \geq (2d+1)\frac{p}{\sigma}$,*

$$\sup_{a\in S^p} \mathbf{Prob}_{z\in B(a,\sigma)}\{ \quad (z) \geq t\} \leq 26\, dp\, \frac{1}{\sigma t}.$$

and

$$\sup_{a\in S^p} \mathbf{E}_{z\in B(a,\sigma)}(\ln \quad (z)) \leq 2\ln p + 2\ln d + 2\ln\frac{1}{\sigma} + 5.$$

While many condition numbers are not conic themselves, they can often be bounded by such. Several applications to linear and polynomial equation solving show that the estimates obtained in this way are easy to derive and quite accurate.

The main theorem is based on a volume estimate of ε-tubular neighborhoods around a real algebraic subvariety of a sphere, intersected with a disk of radius σ. Besides ε and σ, this bound depends only the dimension of the sphere and on the degree of the defining equations.

References

[1] P. Bürgisser, F. Cucker, and M. Lotz, *The probability that a small perturbation of a numerical analysis problem is difficult*, Mathematics of Computation, to appear (2007).

Combinatorial Construction of Locally Testable Codes

Or Meir

An error correcting code is said to be *locally testable* if there is a test that can check whether a given string is a codeword of the code, or rather far from the code, by reading only a constant number of symbols of the string. Locally Testable Codes (LTCs) were first explicitly studied by Goldreich and Sudan [4] and since then few constructions of LTCs were suggested (see [3] for a survey of those constructions).

LTCs are connected with Probabilistically Checkable Proofs (PCPs) and can be seen as the "Combinatorial counterparts" of PCPs. Since they are simpler objects then PCPs, one might expect that constructing LTCs would be easier than constructing PCPs. However, all the known constructions either use PCP as a building block, or imply directly the existence of a PCP.

In this work we present a new and simpler construction of LTCs that seems to be strictly weaker than PCP. Another important feature of our construction is that it is purely combinatorial, while previous constructions were very algebraic. Finally, our construction matches the parameters of the best known construction of LTCs by Ben-Sasson and Sudan [1] (in both cases, these constructions are combined with Dinur's gap amplification technique [2] in order to achieve the best possible parameters). However, unlike the construction of [1], our construction is not entirely explicit.

References

[1] E. Ben-Sasson and M. Sudan, *Simple PCPs with poly-log rate and query complexity.*, STOC 2005, pp. 266-275 (see ECCC TR04-060).
[2] I. Dinur, *The PCP theorem by gap amplification*, STOC 2006, pp. 241-250 (see ECCC TR05-046).
[3] O. Goldreich, *Short Locally Testable Codes and Proofs (Survey)*, ECCC TR05-014, 2005.
[4] O. Goldreich and M. Sudan, *Locally testable codes and PCPs of almost linear length*, FOCS 2002, pp. 13-22 (see ECCC TR02-050, 2002).

Sub-Constant Error Low Degree Test and *PCP* of Almost-Linear Size

Dana Moshkovitz
(joint work with Ran Raz)

The PCP theorem [2, 1] is one of the most important theorems proven in Theoretical Computer Science. The PCP theorem states that any mathematical proof can be written in a different format, such that the proof can be (probabilistically) verified by querying only a *constant* number of places in it. The PCP theorem implies hardness of approximation problems, as well as yields constructions of codes with local testing and decoding properties. Two parameters of a PCP that play a central role are the size of the PCP and its probability of error.

In 1997 researchers managed to construct PCPs with polynomial size and sub-constant probability of error [11, 3, 7]. In the last 6 years, many researchers got extremely interested in PCPs of almost linear size and managed to construct such PCPs [8, 5, 4, 6]. However, these last constructions of PCPs with almost linear size only achieve constant (and not sub-constant) probability of error. The bottleneck for constructing PCPs that have both sub-constant error and almost linear size was the construction of *low degree tests* that have both sub-constant error and almost linear size. We constructed such tests [9] and proved a corresponding PCP theorem [10].

References

[1] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *JACM*, 45(3):501–555, 1998.

[2] S. Arora and S. Safra. Probabilistic checking of proofs: a new characterization of NP. *JACM*, 45(1):70–122, 1998.

[3] S. Arora and M. Sudan. Improved low-degree testing and its applications. *Combinatorica*, 23(3):365–426, 2003.

[4] E. Ben-Sasson, O. Goldreich, P. Harsha, M. Sudan, and S. Vadhan. Robust PCPs of proximity, shorter pcps and applications to coding. In *Proc. 36th STOC*, pages 1–10, 2004.

[5] E. Ben-Sasson, M. Sudan, S. P. Vadhan, and A. Wigderson. Randomness-efficient low degree tests and short PCPs via epsilon-biased sets. In *Proc. 34th STOC*, pages 612–621, 2003.

[6] I. Dinur. The PCP theorem by gap amplification. In *Proc. 38th STOC*, 2006.

[7] I. Dinur, E. Fischer, G. Kindler, R. Raz, and S. Safra. PCP characterizations of NP: Towards a polynomially-small error-probability. In *Proc. 31st STOC*, pages 29–40, 1999.

[8] O. Goldreich and M. Sudan. Locally testable codes and PCPs of almost-linear length. In *Proc. 43rd FOCS*, pages 13–22, 2002.

[9] D. Moshkovitz and R. Raz. Sub-constant error low degree test of almost-linear size. In *Proc. 38th STOC*, pages 21–30, 2006.

[10] D. Moshkovitz and R. Raz. Sub-constant error probabilistically checkable proof of almost-linear size. Technical Report TR07-026, ECCC, 2007.

[11] R. Raz and S. Safra. A sub-constant error-probability low-degree test and a sub-constant error-probability PCP characterization of NP. In *Proc. 29th STOC*, pages 475–484, 1997.

The Cryptographic Applications of Compressibility With Respect to Solutions

Moni Naor
(joint work with Danny Harnik)

We study compression that preserves the solution to an instance of a problem rather than preserving the instance itself. Our focus is on the compressibility of NP decision problems. We consider NP problems that have long instances but relatively short witnesses. The question is, can one efficiently compress an instance and store a shorter representation that maintains the information of whether the original input is in the language or not. We want the length of the compressed instance to be polynomial in the length of the witness and polylogarithmic in the length of original input. We discuss the differences between this notion and similar notions from parameterized complexity.

Our motivation for studying this issue stems from the vast cryptographic implications such compressibility has. For example, we say that SAT is compressible if there exists a polynomial p, so that given a formula consisting of m clauses over n variables it is possible to come up with an equivalent (w.r.t satisfiability) formula of size at most p(n, log m). Then, given a compression algorithm for SAT we provide a construction of: (i) A one-way function from a distributionally-hard problem. (ii) Collision resistant hash functions from any one-way function. The latter task was shown to be impossible via black-box reductions by Simon [4], and indeed the construction presented is inherently non-black-box. Another application of SAT compressibility is a cryptanalytic result concerning the limitation of everlasting security in the bounded storage model (see [1, 2]) when mixed with (time) complexity based cryptography.

References

[1] Y. Aumann and Y.Z. Ding and M.O. Rabin, *Everlasting Security in the Bounded Storage Model*, IEEE Transactions on Information Theory, 48(6), 2002, 1668–1680. EUROCRYPT 2004, Lecture Notes in Computer Science 3027, Springer, 126–137.

[2] S. Dziembowski and U. Maurer, *On Generating the Initial Key in the Bounded-Storage Model*,

[3] Danny Harnik and Moni Naor, *On the Compressibility of NP Instances and Cryptographic Applications*, FOCS 2006, 719–728.

[4] Dan Simon, *Finding Collisions on a One-Way Street: Can Secure Hash Functions Be Based on General Assumptions?*, EUROCRYPT 1998, Lecture Notes in Computer Science 1403, Springer, 334–345.

Two problems related to the Max-Cut and the Unique Games Conjecture

Ryan O'Donnell

This talk is in two parts.

In the first part we report on recent work with Yi Wu [1]. In this work, we complete a long line of research into SDP algorithms and hardness results for the Max-Cut problem. Specifically, we explicitly identify a certain curve $S : [\frac{1}{2}, 1] \to [\frac{1}{2}, 1]$ with the following properties: For each $c \in [\frac{1}{2}, 1]$, there is a graph with Max-Cut at most $S(c)$ but SDP relaxation at least c. On the other hand, every graph with SDP relaxation at least c has a cut of value at least $S(c)$, and further, this cut is findable via an efficient "RPR^2" SDP algorithm. Furthermore, we connect SDP analysis to Long Code test analysis and show the following: Among all (Max-Cut) Long Code tests with completeness at least c, the lowest possible achievable soundness is $S(c)$. Further consequences of these results for algorithmic hardness are also derived.

In the second part of the talk, we discuss a certain aspect of the Unique Games Conjecture we feel is overlooked. Namely, we do not know any distribution on Unique-Label-Cover instances — natural or not — for which it *seems* harder to approximate solutions better than the extent for which we know NP-hardness. Or more concretely, we do not know a distribution on Max-2Lin(2) instances with value $1-\epsilon$ for which finding $1-\frac{5}{4}\epsilon$ solutions even "seems" hard — whereas the Unique Games Conjecture predicts that even finding $1-\Theta(\sqrt{\epsilon})$ solutions should be hard. This is in contrast to, say, Max-3Lin(2), where for the most natural random planted $1-\epsilon$ instances, finding $\frac{1}{2}+\epsilon$ empirically seems very hard. We propose as an open problem looking for distributions on Max-2Lin(2) or Max-2Lin(q) instances that seem hard to approximate.

References

[1] R. O'Donnell, Y. Wu, *An optimal SDP algorithm for Max-Cut, and an equally optimal Long Code test.* Manuscript, 2007.

Tight Integrality gaps for Vertex Cover SDPs in the Lovasz-Schrijver hierarchy

Toniann Pitassi

(joint work with Konstantinos Georgiou, Avner Magen, Iannis Tourlakis)

Linear and semidefinite programming are highly successful approaches for obtaining good approximations for NP-hard optimization problems. For example, breakthrough approximation algorithms for Max Cut and Sparsest Cut use semidefinite programming.

Perhaps the most prominent NP-hard problem whose exact approximation factor is still unresolved is Vertex Cover. PCP-based techniques of Dinur and Safra show that it is not possible to achieve a factor better than 1.36; on the other hand no known algorithm does better than the factor of 2 achieved by the simple greedy algorithm. Furthermore, there is a widespread belief that SDP techniques are the most promising methods available for improving upon this factor of 2.

Following a line of study initiated by Arora, Bollobas, Lovasz and Tourlakis, our aim is to show that a large family of LP and SDP based algorithms fail to produce an approximation for Vertex Cover better than 2. Lovasz and Schrijver introduced the LS systems that naturally capture large classes of LP and SDP relaxations. The strongest of these systems, LS_+, captures the celebrated SDP-based algorithms for Max Cut and Sparsest Cut mentioned above.

We prove an integrality gap of 2 for Vertex Cover SDPs resulting from tightening the standard LP relaxation with $\Omega(\sqrt{\log n/\log\log n})$ rounds of LS_+. While tight integrality gaps for Vertex Cover were known for the weaker LS system previous results did not preclude a polynomial-time $2-\Omega(1)$ approximation algorithm based on LS_+, even when restricted to only two rounds of LS_+ tightenings.

Quantum Frege proofs and a problem on quantum computing

Pavel Pudlák

In this talk I shall address the question whether quantum circuits could help us prove theorems of the classical propositional calculus faster than conventional devices. I shall propose a class of proof systems, *Quantum Frege Proof Systems.* This is based on a generalization of the concept of a Frege deduction rule to the quantum setting. A *quantum Frege rule* is roughly a linear superposition of classical Frege rules. Given a finite set of quantum deduction rules, a *quantum Frege proof* is a sequence of proof lines, the first line being empty and each next line is obtained from the previous one by applying one of the quantum deduction rules. Thus each proof line is a quantum superposition of strings of formulas. We say that the proof proves a given proposition ϕ if ϕ occurs in the last proof line with amplitude α, $|\alpha|^2 \geq 1/2$ (ie., if we measure the last state we shall see a string of propositions which includes ϕ with probability $\geq 1/2$). We represent a quantum Frege proof by a string of quantum circuits that compute the transitions defined by the quantum Frege rules.

Given a quantum Frege proof P of a tautology ϕ, one can easily show that there exists a classical Frege proof with the same number of steps and the same bound on the size of formulas involved. However, if we are given a representation of P by the string of quantum circuits, we do not know how to construct this classical proof. We can show that a classical proof cannot be constructed in polynomial time, if factoring is not computable in polynomial time. The proof of this result is based on tautologies formalizing a bit commitment schema. What remains open is whether one can construct the classical proof using polynomial size *quantum* circuits. This is closely related to the following problem about histories.

Let $\mathbf{B}$ be the basis of the Hilbert space of n qubits consisting of the 2^n strings of 0-1 bits. Let $K = (U_1, \ldots, U_t)$ be a string of unitary operators. For $a_0, a_1, \ldots, a_t \in \mathbf{B}$, we shall say that $(a_0, a_1, \ldots, a_t)$ is a *history* of K, if for all $i = 1, \ldots, t$, $\langle a_i|U_i|a_{i-1}\rangle \neq 0$. Let $U = U_t \ldots U_1$ denote the product of a string of the unitary transformations. If $a, b \in \mathbf{B}$ are such that $\langle b|U|a\rangle \neq 0$, then there exists a history of the form $(a = a_0, a_1, \ldots, a_t = b)$.

Problem. *Suppose the unitary transformations U_i are given by quantum circuits C_i. Let also $a \in \mathbf{B}$ be given and assume that measuring the first bit of $|U|a\rangle$ gives 0 with probability at least 1/2. Is it possible to construct a history $(a = a_0, a_1, \ldots, a_t)$ such that the first bit of a_t is 0 using polynomial size quantum circuits?*

Related questions have been studied in [1].

References

[1] S. Aaronson, *Quantum Computing and Hidden Variables.* Physical Review A **71:032325,** March 2005.

[2] P. Pudlák, *Quantum deduction rules*, Electronic Colloquium on Computational Complexity **TR07-032** (2007).

Randomness versus Hardness and Lower Bounds for Constant-Depth Arithmetic Circuits

Ran Raz

I gave a short description of the main results in [R].

We present simple-to-describe problems, that seem natural-to-study in the context of pseudorandomness and explicit constructions of combinatorial objects, and are seemingly unrelated to arithmetic circuit complexity, and whose solution would give strong (up to exponential) lower bounds for the size of general arithmetic circuits. We then prove lower bounds of $n^{1+\Omega(1/d)}$ for the size of arithmetic circuits of depth d for explicit polynomials of degree $O(d)$.

Our main results are as follows: Let $\mathbb{F}$ be a field and let n be an integer.

(1) Let $s = s(n), m = m(n), r = r(n)$ be integers s.t. $n^{\omega(1)} \leq s < m = n^r$. (Think of r as relatively small, say $r = \log\log n$).

Can one give an explicit polynomial-mapping $f\colon \mathbb{F}^n \to \mathbb{F}^m$ of total-degree at most 2^n, such that, the image of f is not contained in the image of any polynomial-mapping $\Gamma\colon \mathbb{F}^s \to \mathbb{F}^m$ of total-degree at most r ?

We show that for any $\mathbb{F}$ of characteristic different than 2, and any s, m, r as above, the existence of an explicit f as above (with the right notion of explicitness) implies super-polynomial lower bounds for computing the permanent over $\mathbb{F}$.

(2) Let $s = s(n), m = m(n), r$ be integers s.t. $s < m = n^{r+1}$, and $2 \leq r \leq O(1)$.

Given (as input) a polynomial-mapping $\Gamma\colon \mathbb{F}^s \to \mathbb{F}^m$ of total-degree at most $2r - 1$, can one output (in polynomial time) an explicit polynomial-mapping $f\colon \mathbb{F}^n \to \mathbb{F}^m$ of total-degree at most $\mathrm{poly}(n)$, such that, the image of f is not contained in the image of Γ ?

We show that for any $\mathbb{F}$ and any s, m, r as above, a positive solution for this problem implies an explicit lower bound of $\Omega(\sqrt{s})$ for the size of arithmetic circuits over $\mathbb{F}$.

(3) For any $d = d(n)$, we give an explicit example for an n-variate polynomial of total-degree $O(d)$, with coefficients in $\{0, 1\}$, such that, any depth d arithmetic circuit for this polynomial (over any field) is of size $\geq n^{1+\Omega(1/d)}$.

References

[R] R. Raz. Elusive Functions and Lower Bounds for Arithmetic Circuits, Manuscript 2007.

Flag Algebras and Density of Triangles in Graphs

Alexander Razborov

This talk is devoted to the part of Extremal Combinatorics that, in the asymptotic form, studies with which densities "template" combinatorial structures (like graphs, digraphs, hypergraphs or tournaments) may or may not appear in unknown (large) structures of the same type. It is worth noting that the whole subject of

Extremal Combinatorics originated in the seminal paper by Turán (1941) devoted to problems of exactly this type (this is why the densities in question are also called *Turán densities*). And, although by now the subject has definitely outgrown these boundaries, it would be fair to say that Turán-like problems still make its core.

Consider for example three problems of a very similar flavour. What is the minimal edge density of a graph that guarantees the existence of at least one copy of K_3 in this graph? Supposing the edge density is greater than this critical value, what is the minimal possible *density* of triangles guaranteed to exist in such a graph (as a function of the edge density)? How about the analogous questions for 3-hypergraphs (with K_3 replaced by K_4^3, the complete 3-graph on 4 vertices)?

Of these three questions, the first one was completely solved in the seminal paper by Turán (in fact, for more general case of K_r, where $r \geq 3$ is an arbitrary constant). The second question is answered in our work, but its generalization to K_r is still open for any $r > 3$. The third question is *widely* open: this is one of the most intriguing, famous and notoriously difficult problems in the whole field of Combinatorics.

Except for the above-mentioned concrete result, we also try to understand and explicitly extract the mathematical structure underlying and unifying many common techniques existing in the "asymptotic" Extremal Combinatorics and find for them a common denominator. The backbone of our framework is made by certain associative commutative algebras over the reals that we call "Flag Algebras"; most of the standard ideas in the area can be then expressed as simple computations in these algebras using a small set of pre-defined homomorphisms and linear mappings between them. This framework captures, among other things, various inductive arguments existing in the area, and, after some routine technical work (done once and for all) it becomes completely free of the ϵ/δ stuff. In this sense it can be viewed as an extremely goal-oriented fragment of the non-standard analysis; another related feature is that computations in the flag algebras are very easy to program, which substantially enhances the search for right relations and techniques useful for any given concrete problem.

The talk is based on two preprints: "Flag Algebras" (to appear in *Journal of Symbolic Logic*) and "On the Minimal Density of Triangles in Graphs" (to appear in *Combinatorics, Probability and Computing*); both are available from my home page http://www.mi.ras.ru/˜razborov/.

A Hypercontractive Inequality for Matrix-Valued Functions

Oded Regev
(joint work with Avraham Ben-Aroya and Ronald de Wolf)

The Bonami-Beckner hypercontractive inequality is a powerful tool in Fourier analysis of real-valued functions on the Boolean cube. We present a version of this inequality for *matrix-valued* functions on the Boolean cube. We also present a number of applications of this. In particular, we analyze maps that encode n

classical bits into m qubits, in such a way that each set of k bits can be recovered with some probability by an appropriate measurement on the quantum encoding; we show that if $m < 0.7n$, then the success probability is exponentially small in k. This may be viewed as a direct product version of Nayak's quantum random access code bound. It in turn implies strong direct product theorems for the one-way quantum communication complexity of Disjointness and other problems. We also slightly strengthen and simplify a result about 3-party communication complexity of Disjointness due to Beame et al.

References

[1] A. Ambainis, A. Nayak, A. Ta-Shma, and U. Vazirani. Quantum dense coding and a lower bound for 1-way quantum finite automata. In *Proceedings of 31st ACM STOC*, pages 376–383, 1999. quant-ph/9804043.

[2] P. Beame, T. Pitassi, N. Segerlind, and A. Wigderson. A strong direct product theorem for corruption and the multiparty communication complexity of set disjointness. *Computational Complexity*, 2007. To appear. Earlier version in Complexity'05.

[3] W. Beckner. Inequalities in Fourier analysis. *Annals of Mathematics*, 102:159–182, 1975.

[4] A. Bonami. Etude des coefficients de Fourier des fonctions de $L^p(G)$. *Annales de l'Institute Fourier*, 20(2):335–402, 1970.

[5] E. A. Carlen and E. H. Lieb. Optimal hypercontractivity for Fermi fields and related noncommutative integration inequalities. *Communications in Mathematical Physics*, 155(1):27–46, 1993.

[6] A. Nayak. Optimal lower bounds for quantum automata and random access codes. In *Proceedings of 40th IEEE FOCS*, pages 369–376, 1999. quant-ph/9904093.

Partial Exposure and Correlated Types in Large Games

Omer Reingold

(joint work with Ronen Gradwohl)

In this work we introduce the notion of partial exposure, in which the players of a simultaneous-move Bayesian game are exposed to the realized types and chosen actions of a subset of the other players. We show that in *any* large simultaneous-move game, each player has very little regret even after being partially exposed to other players. Additionally, in any extensive version of the game in which a player is partially exposed to other players, her original strategy is very likely still a best response.

Furthermore, we generalize the recent results of Kalai (2004, 2005) [3, 4], and show that the equilibria of large continuous games with many semi-anonymous players are ex post Nash and structurally robust even when the types are correlated. Two forms of correlation are permitted: local dependencies, in which each player's type can depend arbitrarily on some fixed set of other players, and "peer-pressure" dependencies, in which any set of k or more players may be mutually dependent (but any $k-1$ are independent).

Finally, we combine the above and show a robustness result for all large games, even with correlated types.

In the talk we aimed to discuss a central notion to our work, which is the effect of random variables on a function [1, 2]. We aimed at discussing the similarity and differences between the effect and the influence of random variables.

References

[1] N. I. Al-Najjar and R. Smorodinsky. Pivotal players and the characterization of influence. Journal of Economic Theory 92, 2000. Pages 318-342.

[2] O. Haggstrom, G. Kalai and E. Mossel. A Law of Large Numbers for Weighted Majority. Adv. in Appl. Math. 37 (2006), no. 1: 112-123.

[3] E. Kalai. Large robust games. Econometrica, Vol. 72, No. 6, November 2004. Pages 1631-1665.

[4] E. Kalai. Partially-specified large games. Lecture Notes in Computer Science 3828, 2005. Pages 3-13.

Designing Boolean Sorting Circuits with Optimal Average Delay

Rüdiger Reischuk

(joint work with A. Jakoby, M. Liśkiewicz, C. Schindelhauer)

In previous work we have introduced an average case measure for the time complexity of Boolean circuits – that is the delay between feeding the input bits into a circuit and the moment when the results are ready at the output gates – and analysed this complexity measure for prefix computations. Here we consider the problem to sort large integers that are given in binary notation. Contrary to a *word comparator sorting circuit* C where a basic computational element, a comparator, is charged with a single time step to compare two elements, in a *bit comparator circuit* C' a comparison of two binary numbers has to be implemented by a Boolean subcircuit `CM` called *comparator module* that is built from Boolean gates of bounded fanin. Thus, compared to C, the depth of C' will be larger by a factor up to the depth of `CM`.

Our goal is to minimize the average delay of bit comparator sorting circuits. The worst-case delay can be estimated by the depth of the circuit. For this worst-case measure two topologically quite different designs seem to be appropriate for the comparator modules: a tree-like one if the inputs are long numbers, otherwise a linear array working in a pipelined fashion. Inserting these into a word comparator circuit we get bit level sorting circuits for binary numbers of length m for which the depth is either increased by a multiplicative factor of oder $\log m$ or by an additive term of order m.

We show that this obvious solution can be improved significantly by constructing efficient sorting and merging circuits for the bit model that only suffer a constant factor time loss on the average if the inputs are uniformly distributed. This is done by designing suitable hybrid architectures of tree compaction and pipelining. These results can also be extended to classes of nonuniform distributions if we put a bound on the complexity of the distributions themselves.

References

M. Ajtai, J. Komlos, and E. Szemeredi, *Sorting in $c \log n$ parallel steps*, Combinatorica 3, 1983, 1-19.

M. Al-Hajery and K. Batcher, *On the bit-level complexity of bitonic sorting networks*, Proc. 22. Int. Conf. on Parallel Processing, 1993, III.209 – III.213.

A. Jakoby, *Die Komplexität von Präfixfunktionen bezüglich ihres mittleren Zeitverhaltens,* Dissertation, Universität zu Lübeck, 1998.

A. Jakoby, R. Reischuk, and C. Schindelhauer, *Circuit complexity: from the worst case to the average case*, Proc. 26. ACM STOC, 1994, 58-67.

A. Jakoby, R. Reischuk, and C. Schindelhauer, *Malign distributions for average case circuit complexity*, Proc. 12. STACS, 1995, Springer LNCS 900, 628-639.

A. Jakoby, R. Reischuk, C. Schindelhauer, and S. Weis, *The average case complexity of the parallel prefix problem*, Proc. 21. ICALP, 1994, Springer LNCS 820, 593-604.

T. Leighton and C. G. Plaxton, *A (fairly) simple circuit that (usually) sorts*, Proc. 31. IEEE FOCS, 1990, 264-274.

Some Specific Derandomizations

Nitin Saxena

1. Towards Depth 3 Identity Testing and Lower Bounds

We study depth-3 arithmetic circuits ($\Sigma\Pi\Sigma$ circuits) of the form:

$$C(x_1, \ldots, x_n) = \sum_{i=1}^{k} \ell_{i,1}^{e_{i,1}} \cdots \ell_{i,c}^{e_{i,c}}$$

where, $\ell_{i,1}, \ldots, \ell_{i,c}$ are linear functions over a field $\mathbb{F}$, c is a constant and say $(e_{1,1} + \cdots + e_{1,c}) =: d$ which is the total degree of the polynomial $C(x_1, \ldots, x_n)$. We show that identity testing of such circuits can be done in *poly* $(\max_i\{(e_{i,1}+1)\cdots(e_{i,c}+1)\}, k, n)$ many field operations. This immediately gives a $poly(2^d, k, n)$ time identity test for general depth 3 circuits. We also show exponential lower bounds for determinant and permanent for circuits of the above form. Our lower bounds hold over any field $\mathbb{F}$. Proving such lower bounds for general depth 3 circuits over fields of charactersitic 0 is an important open problem [2].

2. Towards Polynomial Factoring over finite fields (assuming GRH)

Finding a nontrivial factor of a given univariate polynomial over a finite field is a fundamental algebraic problem. It has a randomized polynomial time algorithm but its deterministic complexity is open. There are many partial results known using the Generalized Riemann Hypothesis, see [1] for references. We extend all these approaches and relate the problem of polynomial factoring (assuming GRH) with the existence of some combinatorial objects called *association schemes*. We conjecture that certain *anti-symmetric* association schemes do not exist which would then imply that polynomial factoring is in P (assuming GRH).

References

[1] Shuhong Gao, *On the Deterministic Complexity of Factoring Polynomials*, Journal of Symbolic Computation, **31(1/2)**, (2001), 19–36.
[2] Dima Grigoriev and Alexander A. Razborov, *Exponential Complexity Lower Bounds for Depth 3 Arithmetic Circuits in Algebras of Functions Over Finite Fields*, FOCS (1998), 269–278.

On the Complexity of Counting Components of Algebraic Varieties

Peter Scheiblechner
(joint work with Peter Bürgisser)

We consider complex algebraic varieties $V = \mathcal{Z}(f_1, \ldots, f_r) \subseteq \mathbb{C}^n$ given by finitely many polynomials $f_1, \ldots, f_r \in \mathbb{C}[X_1, \ldots, X_n]$. A standard argument shows that the complexity of the following problems does not essentially change when changing the input data structure from dense to sparse or straight-line program representation.

For simplicity we state our results in the Turing model only, where we restrict ourselves to rational polynomials. We consider the following problems:

#CC Given $f_1, \ldots, f_r$, compute the number of connected components of V.

#IC Given $f_1, \ldots, f_r$, compute the number of irreducible components of V.

#Betti(k) Given $f_1, \ldots, f_r$, compute the kth topological Betti number of V.

Furthermore, we denoty by #IC(r) the problem #IC restricted to a fixed number r of equations. Our main results are summarised in the following table.

	#Betti(k)	#CC	#IC	#IC(r)
PSPACE	hard	complete	contained	
#P			hard	
P				
Random NC				contained

We note that in the algebraic model one can derandomise the result for #IC(r) at the cost of good parallelisation, i.e., the problem can be solved in deterministic sequential polynomial time in the algebraic model.

References

[Bas06] S. Basu. Computing the first few Betti numbers of semi-algebraic sets in single exponential time. *J. Symbolic Comput.*, 41(10):1125–1154, 2006.
[BC03] P. Bürgisser and F. Cucker. Counting complexity classes for numeric computations I: Semilinear sets. *SIAM J. Comp.*, 33:227–260, 2003.
[BC06] P. Bürgisser and F. Cucker. Counting complexity classes for numeric computations II: Algebraic and semialgebraic sets. *J. Compl.*, 22:147–191, 2006.
[BS07] P. Bürgisser and P. Scheiblechner. Differential forms in computational algebraic geometry. In *ISSAC '07: Proceedings of the 2007 international symposium on Symbolic and algebraic computation*, New York, NY, USA, 2007. ACM Press. To appear.

[Can88] J. Canny. Some algebraic and geometric computations in PSPACE. In *Proc. 20th Ann. ACM STOC*, pages 460–467, 1988.

[GH91a] M. Giusti and J. Heintz. Algorithmes -disons rapides- pour la décomposition d'une variété algébrique en composantes irréductibles et équidimensionnelles. In T. Mora C. Traverso, editor, *Effective Methods in Algebraic Geometry (Proceedings of MEGA'90)*, volume 94 of *Progress in Math.*, pages 169–193, New York, NY, USA, 1991. Birkhäuser.

[Rei79] J.H. Reif. Complexity of the mover's problem and generalizations. In *Proc. 20th FOCS*, pages 421–427, 1979.

[Sch07] P. Scheiblechner. On the complexity of deciding connectedness and computing Betti numbers of a complex algebraic variety. *J. Compl.*, 23:359–379, 2007.

Cryptography Based on the Equivalence of Quadratic Forms

Claus Peter Schnorr
(joint work with R.J. Hartung)

Notation. A symmetric matrix $A = A^t \in \mathbb{Z}^{n\times n}$ defines the quadratic form $\vec{x}^t A\vec{x}$. The forms $A_0, A_1 \in \mathbb{Z}^{n\times n}$ are *equivalent* if $T^tA_0T = A_1$ holds for some $T \in \mathrm{GL}_n(\mathbb{Z})$.
References. [Ca78] presents the classical theory of rational quadratic forms, for LLL-reduction of quadratic forms see [S07, Si05] and for lattice based cryptography see [MG02].

We present public key identification and digital signatures based on the computational equivalence problem (**CEP**) of n-ary quadratic forms, $n \geq 3$. We present proofs of knowledge of an equivalence transform $T \in \mathrm{GL}_n(\mathbb{Z})$. Small dimension n yields short private and public keys and efficient protocols.

Lattices correspond to positive definite quadratic forms. However, lattice based cryptography requires lattices of high dimension n because the lattice problems **SVP** and **CVP** are in polynomial time for any fixed dimension n and get slowly harder as n increases. Importantly, solving **CEP** for a small T such that $T^tA_1T = A_0$ is NP-hard for indefinite forms $A_1, A_0 \in \mathbb{Z}^{n\times n}$ for every fixed $n \geq 3$. This follows from the NP-hardness proof of binary quadratic equations over the integers of [MA78]. This NP-hardness proof requires that $\det A$ has a large square factor. However **CEP** is polynomial time for isotropic, ternary forms with odd, squarefree determinant. For isotropic forms A an isotropic vector $\mathbf{y} \neq \mathbf{0}$, such that $\mathbf{y}^tA\mathbf{y} = 0$ can be found in polynomial time given the factorization of $\det A$ [Si05]. Given an isotropic vector $\mathbf{y}$ the equation $\mathbf{x}^tA\mathbf{x} = c$ can be solved in polynomial time for every $c \in \mathbb{Z}$ if $\det A$ is odd and squarefee.

Proof of knowledge. Prover $\mathcal{P}$ proves to verifier $\mathcal{V}$ knowledge of S such that $S^tA_1S = A_0$ by iterating:

1. $\mathcal{P}$ computes and sends an LLL-reduced form $A' := T^tA_0T$ for a randomized $T \in \mathrm{GL}_n(\mathbb{Z})$, see [HS07].
2. $\mathcal{V}$ sends a random one-bit challenge $b \in_R \{0, 1\}$,
3. $\mathcal{P}$ sends the reply $R_b := S^bT \in \mathrm{GL}_n(\mathbb{Z})$, and $\mathcal{V}$ checks that $R_b^tA_bR_b = A'$.

If a fraudulent $\widetilde{\mathcal{P}}$ succeeds with $\widetilde{A}'$ and replies $\widetilde{R}_b$ for both $\widetilde{R}_0, \widetilde{R}_1$ he gets an equivalent private key $S' := \widetilde{R}_1 \widetilde{R}_0^{-1}$ satisfying $S'^t A_1 S' = A_0$. This protocol is statistical zeroknowledge under reasonable heuristics.

Another proof of knowledge uses, long challenges and can be transformed into an efficient public key signature scheme by replacing $\mathcal{V}$ through a cryptographic hash function. This proof represents some $c \in \mathbb{Z}$ as $\mathbf{x}^t A_b \mathbf{x} = c = \mathbf{y}^t A' \mathbf{y}$. Here $\mathbf{x}, \mathbf{y}$ must and can be chosen such that the problem to extend $\mathbf{x}, \mathbf{y}$ to an equivalence transform $T \in \mathrm{GL}_n(\mathbb{Z})$ still requires exponential time by known algorithms. In fact the reconstruction of T requires to represent the determinant of some $(n-1)$-dimensional subform of A' by the $(n-1)$-dimensional, adjoint form $A_b^{\#}$ of A_b. No subexponential algorithm is known for this latter problem for any fixed $n \geq 4$. Most instances of this problem are subexponential for $n = 3$.

References

[Ca78] *J.W.S. Cassels*, Rational Quadratic Forms. L.M.S Monographs, **13**, Academic Press, 1978.

[HS07] *R.J. Hartung and C.P. Schnorr*, Public Key Identification Based on the Equivalence of Quadratic Forms. In Proc. of MFCS, Aug. 26 –Aug. 31, Český Krumlov, Czech Republic, LNCS ??, Springer-Verlag, 2007.

[MA78] *K. Manders and L.M. Adleman*, NP-complete decision problems for binary quadratics, *JCCS*, 1978.

[MG02] *D. Micciancio and S. Goldwasser*, Complexity of Lattice Problems, A Cryptographic Perspective. Kluwer Academic Publishers, London, 2002.

[S07] *C.P. Schnorr*, Progress on LLL and Lattice Reduction. Proc. LLL+25, Caen, 29.06-1.07.2007.

[Si05] *D. Simon*, Solving Quadratic Equations Using Reduced Unimodular Quadratic Forms. *Math. of Comp.* **74** (251), pp. 1531–1543, 2005. Moreover preprint math.unicaen.fr (2005) "... on dimensions 4, 5 and more".

Low-end uniform hardness versus randomness tradeoffs for Arthur-Merlin games

Ronen Shaltiel

(joint work with Chris Umans)

In 1998, Impagliazzo and Wigderson [IW98] proved a hardness vs. randomness tradeoff for BPP in the *uniform setting*, which was subsequently extended to give optimal tradeoffs for the full range of possible hardness assumptions by Trevisan and Vadhan [TV02] (in a slightly weaker setting). In 2003, Gutfreund, Shaltiel and Ta-Shma [GSTS03] proved a uniform hardness vs. randomness tradeoff for AM, but that result only worked on the "high-end" of possible hardness assumptions.

In this work, we give uniform hardness vs. randomness tradeoffs for AM that are near-optimal for the full range of possible hardness assumptions. Following [GSTS03], we do this by constructing a hitting-set-generator (HSG) for AM with "resilient reconstruction." Our construction is a recursive variant of the Miltersen-Vinodchandran HSG [MV99], the only known HSG construction with this required property. The main new idea is to have the reconstruction procedure operate implicitly and locally on superpolynomially large objects, using tools from PCPs (low-degree testing, self-correction) together with a novel use of extractors that are built from Reed-Muller codes [SU01] for a sort of locally-computable error-reduction.

As a consequence we obtain gap theorems for AM (and AM $\cap$ coAM) that state, roughly, that either AM (or AM $\cap$ coAM) protocols running in time $t(n)$ can simulate all of EXP ("Arthur-Merlin games are powerful"), or else all of AM (or AM $\cap$ coAM) can be simulated in nondeterministic time $s(n)$ ("Arthur-Merlin games can be derandomized"), for a near-optimal relationship between $t(n)$ and $s(n)$. As in [GSTS03], the case of AM $\cap$ coAM yields a particularly clean theorem that is of special interest due to the wide array of cryptographic and other problems that lie in this class.

References

[GSTS03] D. Gutfreund, R. Shaltiel, and A. Ta-Shma. Uniform hardness versus randomness tradeoffs for Arthur-Merlin games. *Computational Complexity*, 12(3-4):85–130, 2003.

[IW98] R. Impagliazzo and A. Wigderson. Randomness vs. time: De-randomization under a uniform assumption. In *39th Annual Symposium on Foundations of Computer Science*, pages 734–743, 1998.

[MV99] P. B. Miltersen and N. V. Vinodchandran. Derandomizing Arthur-Merlin games using hitting sets. In *40th Annual Symposium on Foundations of Computer Science*, pages 71–80, 1999.

[SU01] R. Shaltiel and C. Umans. Simple extractors for all min-entropies and a new pseudo-random generator. In *Proceedings of the 42nd Symposium on Foundations of Computer Science*, pages 648–657, 2001.

[TV02] L. Trevisan and S. Vadhan. Pseudorandomness and average-case complexity via uniform reductions. In *Proceedings of the 17th Annual Conference on Computational Complexity*, 2002.

Towards Universal Semantic Communication

Madhu Sudan
(joint work with Brendan Juba)

Consider the following fantastic scenario: Earth has just started receiving some signals from outer space. These signals don't seem like usual cosmic noise. Potentially an intelligent alien civilization is trying to make contact. How should Earth respond? How can we (earthlings) tell if the aliens are receiving our response and reacting to it? Are they really intelligent, or are we talking to sunspots? If they are intelligent, will we ever be able to achieve meaningful interaction in this setting?

Can these questions be tacked mathematically? The classical theory of communication, while founded solidly in mathematics, typically ignores the issue of semantics of communication, and has focussed principally on quantitative measures in syntactic settings. Increasingly, however, it is becoming clear that practical challenges to communication arise due to semantic gaps between senders and receivers. The fictional problem above, merely, carries this gap to the extreme.

In this work, we attempt to describe how the theory of computational complexity can shed light on such interactions. The principal goal is to figure out how some of the nebulous notions, such as intelligence and understanding, should be defined in this setting. We assert that in order to communicate "successfully", the communicating players should be explicit about their goals - what the communication should achieve. We show that when the goals are explicit the communicating players can achieve *meaningful* interaction, provided the players are *capable* of satisfying the goals, and *cooperative*, under reasonable, mathematical, definitions of these notions.

References

[1] B. Juba and M. Sudan, *Towards Universal Semantic Communication*, Preliminary Manuscript, available from `http://people.csail.mit.edu/madhu/papers/juba.pdf`, February 2007.

Integrality Gaps for Vertex Cover in Lovasz-Schrijver Hierarchies

Luca Trevisan
(joint work with Grant Schoenebeck, Madhur Tulsiani)

We study linear and semidefinite programming relaxations of Vertex Cover arising from repeated applications of the "lift-and-project" method of Lovasz and Schrijver [5] starting from the standard linear programming relaxation.

For linear programs (LS), Arora, Bollobas, Lovasz and Tourlakis [1] prove that the integrality gap remains at least $2-\epsilon$ after $\Omega_\epsilon(\log n)$ rounds, where n is the number of vertices, and Tourlakis [6] proves that integrality gap remains at least $1.5-\epsilon$ after $\Omega_\epsilon((\log n)^2)$ rounds. We prove that the integrality gap remains at least $2-\epsilon$ after $\Omega_\epsilon(n)$ rounds.

For semidefinite programs (LS+), Goemans and Kleinberg [4] prove that after one round the integrality gap remains arbitrarily close to 2. Charikar [2] proves an integrality gap of 2 for a stronger relaxation that is, however, incomparable with two rounds of LS+ and is strictly weaker than the relaxation resulting from a constant number of rounds. Georgiou et al. [3] show that the integrality gap remains $2-o(1)$ after $\Omega(\sqrt{\log n/\log\log n})$ rounds. We prove that the integrality gap remains at least $7/6-\epsilon$ after $\Omega_\epsilon(n)$ rounds.

References

[1] Sanjeev Arora, Béla Bollobás, László Lovász, and Iannis Tourlakis. Proving integrality gaps without knowing the linear program. *Theory of Computing*, 2(2):19–51, 2006.

[2] Moses Charikar. On semidefinite programming relaxations for graph coloring and vertex cover. In *Proceedings of the 13th ACM-SIAM Symposium on Discrete Algorithms*, pages 616–620, 2002.
[3] Konstantinos Georgiou, Avner Magen, Toniann Pitassi and Iannis Tourlakis. Tight integrality gaps for Vertex Cover SDPs in the Lovasz-Schrijver hierarchy ECCC Report TR06-152, 2006.
[4] Jon M. Kleinberg and Michel X. Goemans. The Lovász Theta function and a semidefinite programming relaxation of vertex cover. *SIAM Journal on Discrete Mathematics*, 11:196–204, 1998.
[5] L. Lovasz and A. Schrijver. Cones of matrices and set-functions and 0-1 optimization. *SIAM J. on Optimization*, 1(12):166–190, 1991.
[6] Iannis Tourlakis. New lower bounds for vertex cover in the Lovasz-Schrijver hierarchy. In *Proceedings of the 21st IEEE Conference on Computational Complexity*, 2006.

Evolvability

Leslie G. Valiant

We suggest a quantitative model of evolution for the purpose of studying how representations of complex functions can evolve from simpler ones within realistic population sizes and numbers of generations [1]. Evolution is treated as a form of computational learning, in which the course of learning depends only on the fitness of the hypothesis on the aggregate of the examples, and not otherwise on the examples. We formulate a notion of evolvability for different classes of functions. It is shown that in any one phase of evolution monotone Boolean conjunctions and disjunctions are evolvable over the uniform distribution, while Boolean parity functions are not. The framework also suggests how a wider range of issues in evolution might be quantified. We also suggest that the process of biological evolution over multiple phases should be viewed as *evolvable target pursuit*, which consists of a series of evolutionary phases, each one pursuing a target that is evolvable in our technical sense, each target being rendered evolvable by the serendipitous combination of the environment and the outcome of previous evolutionary phases.

References

[1] L.G. Valiant, *Proc. 32nd International Symposium on Mathematical Foundations of Computer Science*, Cesky Krumlov, Czech Republic, August 26-31, 2007, pp 22-43.

One-way multi-party communication lower bound for pointer jumping with applications

Emanuele Viola
(joint work with Avi Wigderson)

In this paper we study the one-way multi-party communication model, in which every party speaks exactly once in its turn. For every fixed k, we prove a tight lower bound of $\Omega\left(n^{1/(k-1)}\right)$ on the probabilistic communication complexity of pointer jumping in a k-layered tree, where the pointers of the i-th layer reside on the forehead of the i-th party to speak. The lower bound remains nontrivial even for $k = (\log n)^{1/3}$ parties. Previous to our work a lower bound was known only for $k = 3$, and in very restricted models for $k > 3$. Our results have the following consequences to other models and problems, extending previous work in several directions.

The one-way model is strong enough to capture *general* (non one-way) multi-party protocols of bounded rounds. Thus we generalize to this multi-party model results on two directions studied in the classical 2-party model (e.g. [PS, NW]). The first is a round hierarchy: We give an exponential separation between the power of r and $2r$ rounds in general probabilistic k-party protocols, for any fixed k and r. The second is the relative power of determinism and nondeterminism: We prove an exponential separation between nondeterministic and deterministic communication complexity for general k-party protocols with r rounds, for any fixed k, r.

The pointer jumping function is weak enough to be a special case of the well-studied disjointness function. Thus we obtain a lower bound of $\Omega\left(n^{1/(k-1)}\right)$ on the probabilistic complexity of k-set disjointness in the one-way model, extending a similar lower bound for the weaker simultaneous model, in which parties simultaneously send one message to a referee [BPSW].

Finally, we infer an exponential separation between the power of different orders in which parties send messages in the one-way model, for every fixed k. Previous to our work such a separation was only known for $k = 3$ [NW].

Our lower bound technique, which handles functions of high discrepancy, may be of independent interest. It provides a "party-elimination" induction, based on a restricted form of a direct-product result, specific to the pointer jumping function.

This work will appear in *FOCS 2007*.

References

[BPSW] P. Beame, T. Pitassi, N. Segerlind, and A. Wigderson. A Direct Sum Theorem for Corruption and the Multiparty NOF Communication Complexity of Set Disjointness. In *Proceedings of the Twentieth Annual Conference on Computational Complexity*, pages 52–66. IEEE, June 12–15 2005.

[NW] N. Nisan and A. Wigderson. Rounds in communication complexity revisited. *SIAM J. Comput.*, 22(1):211–219, 1993.

[PS] C. H. Papadimitriou and M. Sipser. Communication complexity. *J. Comput. System Sci.*, 28(2):260–269, 1984.

Nechiporuk Bounds for the Middle Bit of Multiplication

Ingo Wegener

(joint work with Philipp Woelfel)

Other results on restricted branching programs for MM_N, the middle bit of multiplication, have revealed a lot about the subfunction structure of this function. This leads to the aim to investigate the best possible bounds on the branching program size and the formula size of MM_N obtainable by Nechiporuk's lower bound technique. We prove bounds of size $\Omega(n^{3/2}/\log n)$ and $\Omega(n^{3/2})$ respectively and prove that these bounds can be improved by not more than an $n^{1/6}$-factor. The results have been presented at the conference Computational Complexity (2005) and the full version is accepted for publication in the journal Computational Complexity.

Network Extractor Protocols and Three-Source Extractors

David Zuckerman
(joint work with Xin Li, Anup Rao)

We design several efficient one-round *network extractor protocols*, which extract private randomness over a network with faulty players when each player has a single, weak random source of sufficient min-entropy. As a corollary, we derive efficient protocols for Byzantine agreement and leader election (and hence the equivalent collective coin-flipping) in the full information model. Our robust protocols run in just one more round than the corresponding protocols with perfect randomness. Our results significantly improve those of Goldwasser, Sudan, and Vaikuntanathan [1].

In a synchronous network, if each of p players has a weak source with min-entropy rate greater than $1/2$, then we essentially match the bounds for perfect randomness: Byzantine agreement tolerating a $1/3 - \alpha$ fraction faulty players in $O(\log p)$ rounds, and leader election tolerating a $1/2 - \alpha$ fraction faulty players in $\log^* p + O(1)$ rounds, for any constant $\alpha > 0$. In a synchronous network, if each player's n-bit source of randomness has $n^{\Omega(1)}$ min-entropy, then the bounds drop to $1/4-\alpha$ and $1/3-\alpha$, respectively. In an asyncrounous network, if each player has access to a source with polynomial min-entropy (though $1/3$ of the players need shorter sources than the others), then our Byzantine agreement protocol tolerates a $1/18 - \alpha$ fraction of faulty players.

Extractors for independent sources are crucial to our results. In particular, our results for asynchronous protocols rely on a new extractor for three independent sources. Two of the sources must have n bits with min-entropy at least n^γ; the third must have $n^{\gamma^2/c}$ bits with min-entropy at least $\log^{10} n$. (Here c is an absolute constant and $\gamma > 0$ is arbitrary.) Previously, extractors for independent sources with min-entropy n^γ required $O(1/\gamma)$ sources [2].

References

[1] Shafi Goldwasser, Madhu Sudan, and Vinod Vaikuntanathan. Distributed computing with imperfect randomness. In Pierre Fraigniaud, editor, *DISC*, volume 3724 of *Lecture Notes in Computer Science*, pages 288–302. Springer, 2005.

[2] Anup Rao. Extractors for a constant number of polynomially small min-entropy independent sources. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*, pages 497–506, 2006.

Reporter: Peter Scheiblechner