On the interaction of TCS and Math[*]

Avi Wigderson

June 29, 2016

1 Introduction

Theoretical Computer Science (TCS) is the study of the mathematical foundations of Computer Science. I am often asked if TCS is a branch of Mathematics, or of Computer Science. It is clearly both. Ever since Turing's 1936 definition of the "Turing machine", we have had a formal mathematical model of computation that enables the rigorous mathematical study of computational tasks, algorithms to solve them, and the resources these require. At the same time, the simple description of the Turing machine allowed its simple logical structure to be implemented in hardware, and its universal applicability fueled the rapid development of computer technology, which now dominates our life. So, TCS researchers (or "theorists," as we call them internally) operate sociologically just like any other mathematicians—they prove theorems. They often follow standard mathematical tendencies to generalize, simplify, and follow their noses based solely on aesthetics and beauty. At the same time, their motivation for defining new models and notions arises from the quest to understand that rich and mysterious notion of computation. This naturally draws inspiration from technological developments and concerns, as well as from trying to understand the computational aspects of various natural (and mathematical) processes.

There are numerous connections and collaborations between TCS and Math, and they can be coarsely divided into a few (overlapping) categories. The first two categories arise from one field trying to use the expertise of the other, and the interactions in this case are typically one-sided. One arises from the need of TCS to use general techniques and results from a host of different mathematical areas. Another arises from the fact that mathematicians, like all other scientists, are great consumers of efficient algorithms and software to simply quickly compute things they need for their research. Today such libraries of algorithms and software exist in most areas of mathematics, including algebra, topology, group theory, geometry, and statistics. The next two categories are more subtle. One arises from the question which is by now almost reflexive given any existence theorem: can the object(s) guaranteed to exist be efficiently found? Even without any direct, practical need for such algorithms, seeking them invariably leads to a deeper understanding of the mathematical field at hand. Another category arises from the fact that TCS research produces new mathematical results, theories and problems, which may not be exclusively or even primarily algorithmic in nature. This body of ideas inspires collaboration and potential further development of other mathematical fields. In both of these categories, the methodology of computational modeling, asymptotic analysis, reductions and completeness often play an important role in the developments.

In this survey we will meet examples of all these types. However, I will try to focus on the last two, and on relatively new developments. I will also aim for variety. This survey will show that hardly any area of classical and modern mathematics is untouched by this computational connection, which in some cases is quite surprising. (I will omit discussion of the extensive connections between TCS and combinatorics, a sister field also focusing on discrete structures, since the many strong interactions are to be expected in this case.)

Due to space limitations, I have chosen to focus on essentially one problem or development within each mathematical field. Typically this touches only a small subarea, and does not even begin to do justice to a

[*] This survey will be a chapter in a book I am writing about Math and Computation

wealth of connections, and so should be viewed as merely a demonstration of a huge potential. Indeed, while in some areas the collaborations are reasonably well established, in others they are just budding, with lots of exciting problems waiting to be solved and theory to be developed. The selection of fields and foci is affected strongly by my personal taste and limited knowledge. I have tried to compensate the deficiencies above by giving some background and intuition, as well as further reading material. Indeed, the short vignettes in this survey will hopefully tempt the reader to explore deeper.

The sections below can be read in any order. Here is a list of the covered areas and topics chosen.

- Number Theory: *primality testing*
- Combinatorial Geometry: *point-line incidence*
- Operator Theory: *the Kadison-Singer problem*
- Metric Geometry: *distortion of embeddings*
- Group Theory: *generation and random generation*
- Mathematical Physics: *Monte-Carlo Markov chains*
- Analysis and Probability: *Noise stability*
- Lattice Theory: *the Lenstra-Lenstra-Lovasz (LLL) algorithm*
- Invariant Theory: *Noether normalization*

General background on the Theory of Computation can be found in the survey (aimed at mathematicians) [**?**] and in the standard textbooks on computational complexity [**?**, **?**, **?**].

We will use the following asymptotic notation. For integer functions, f, g we will use $f = O(g)$ if for some positive constant C we have for all n, $f(n) \leq C \cdot g(n)$. Similarly, we will use $f = \Omega(g)$ if for some positive constant c we have for all n , $f(n) \geq c \cdot g(n)$.

2 Number Theory

Here is an except from C. F. Gauss' appeal[1] to the mathematics community of his time (in article 329 of Disquisitiones Arithmeticae (1801)), regarding the computational complexity of testing primality and factorization of integers.

The problem of distinguishing prime numbers from composite numbers, and of resolving the latter into their prime factors is known to be one of the most important and useful in arithmetic. It has engaged the industry and wisdom of ancient and modern geometers to such an extent that it would be superfluous to discuss the problem at length. Nevertheless we must confess that all methods that have been proposed thus far are either restricted to very special cases or are so laborious and difficult that even for numbers that do not exceed the limits of tables constructed by estimable men, they try the patience of even the practiced calculator. And these methods do not apply at all to larger numbers ... the dignity of the science itself seems to require that every possible means be explored for the solution of a problem so elegant and so celebrated.

A remarkable response was found recently, by Agrawal, Kayal, and Saxena [**?**].

Theorem 2.1. *An integer $N \geq 2$ is prime if and only if*

- *N is not a perfect power,*
- *N does not have any prime factor $\leq (\log N)^4$,*

[1] Which is of course in Latin. I copied this English translation from a wonderful survey of Granville [**?**] on the subject matter of this section

- *For every $r, a < (\log N)^4$ we have the following equivalence of polynomials over $\mathbb{Z}_N[X]$:*

$$(X+a)^N \equiv X^N + a \mod (X^r - 1)$$

It is not hard to see that this characterization gives rise to a simple algorithm for testing primality that is deterministic, and runs in time that is *polynomial* in the binary description length of N. Previous deterministic algorithms either assumed the Riemann hypothesis [**?**] or required slightly superpolynomial time [**?**]. The AKS deterministic algorithm came after a sequence of efficient *probabilistic* algorithms [**?**, **?**, **?**, **?**], many requiring sophisticated use and development of number theoretic techniques. These probabilistic algorithms were motivated by, and important to the field of cryptography. What is less well-known is that AKS developed their deterministic algorithm by carefully "derandomizing" a previous probabilistic algorithm of [**?**]. We note that *derandomization*, the conversion of probabilistic algorithms into deterministic ones, is by now a major area in computational complexity with a rich theory, many other such successes as well as challenges.

Gauss' second challenge, of whether efficiently factoring integers is possible, remains open. Indeed, the assumed hardness of factoring is the main guarantee of security in almost all cryptographic and e-commerce systems around the world. More generally, cryptography is an avid consumer of number theoretic notions, including elliptic curves, Weil pairings, and more, which are critical to a variety of cryptographic primitives and applications. As noted by E. Bombieri, these developments shatter Hardy's view of number theory as a completely useless intellectual endeavor.

There are, of course, several problems whose very definitions depend on integer factoring, that can nevertheless be solved efficiently. Perhaps the earliest algorithm ever formally described is Euclid's algorithm for computing the GCD (greatest common divisor) of two given integers[2] m and n. Another famous one is for computing their Legendre-Jacobi symbol $(\frac{m}{n})$ via Gauss' law of quadratic reciprocity.

A fast algorithm for factoring may come out of left-field with the new development of quantum computing [**?**], the study of computers based on quantum-mechanical principles. Shor has shown in [**?**] that such computers are capable of factoring integers in polynomial time. This result led governments, companies, and academia to invest billions in developing technologies which will enable building large-scale quantum computers, and the jury is still out on the feasibility of this project. There is no known theoretical impediment for doing so, but one possible reason for failure of this project is the existence of yet-undiscovered principles of quantum mechanics.

Other central computational problems include solving polynomial equations in finite fields, for which one of the earliest efficient (probabilistic) algorithm was developed by Berlekamp [**?**]. Many other examples can be found in [**?**].

3 Combinatorial geometry

What is the smallest area of a planar region which contains a unit length segment in *every* direction? This is the Kakeya needle problem (and such sets called "Kakeya sets"), which was solved surprisingly by Besicovich [**?**] who showed that this area can be arbitrarily close to zero! Replacing area (namely, Lesbegue measure) by the more robust Hausdorff and Minkowski dimensions, Davies [**?**] proved that both must be 2 (namely, while possibly a sparse set, a Kakeya set in the plane is fully dimensional).

It is natural to extend this problem to higher dimensions. However, obtaining analogous results (namely, that the Hausdorff and Minkowski dimensions are full) turns out to be extremely difficult. Despite the seemingly recreational flavor, this problem has significant importance in a number of mathematical areas (Fourier analysis, Wave equations, analytic number theory, and randomness extraction), and has been attacked through a considerable diversity of mathematical ideas (see [**?**]).

The following finite field analogue of the above Euclidean problem was suggested by Wolff [**?**]. Let $\mathbb{F}$ denote a finite field of size q. A set $K \subseteq \mathbb{F}^n$ is called Kakeya if it contains a line in every direction. More precisely, for every direction $b \in \mathbb{F}^n$ there is a point $a \in \mathbb{F}^n$ such that the line $\{a + bt : t \in \mathbb{F}\}$ is contained

[2] It extends to polynomials, and allows efficient way of computing multiplicative inverses.

in K. As above, we would like to show that any such K must be large (think of the dimension n as a large constant, and the field size q as going to infinity).

Conjecture 3.1. Let $K \subseteq \mathbb{F}^n$ be a Kakeya set. Then $|K| \geq C_n q^n$, where C_n is a constant depending only on the dimension n.

The best exponent of q in such a lower bound intuitively corresponds to the Hausdorff and Minkowski dimensions in the Euclidean setting. Using sophisticated techniques from arithmetic combinatorics Bourgain, Tao and others improved the trivial bound of $n/2$ to about $4n/7$.

Curiously, the exact same conjecture arose, completely independently, within TCS, from work [**?**] on *randomness extractors*, an area which studies the "purification" of "weak random sources" (see e.g. the survey [**?**]). With this motivation, Dvir [**?**] brilliantly proved the Wolff conjecture, using the (algebraic-geometric) "polynomial method" (which is inspired by techniques in decoding algebraic error-correcting codes). Many other applications of this technique to other geometric problems quickly followed, including the Guth-Katz[**?**] resolution of the famous Erdős distance problem, as well as for optimal randomness extraction and more (some are listed in Dvir's survey [**?**]).

Subsequent work determined the exact value of the constant C_n above (up to a factor of 2) [**?**].

Theorem 3.2. *Let $K \subseteq \mathbb{F}^n$ be a Kakeya set. Then $|K| \geq (q/2)^n$. On the other hand, there exist Kakeya sets of size $\leq 2 \cdot (q/2)^n$.*

Many other problems regarding incidences of points and lines (and higher-dimensional geometric objects) have been the source of much activity and collaboration between geometers, algebraists, combinatorialists and computer scientists. The motivation for these questions in the computer science side come from various sources, e.g. problems on local correction of errors [**?**] and derandomization [**?**, **?**]. Other incidence theorems, e.g. Szemeredi-Trotter [**?**] and its finite field version of Bourgain-Katz-Tao [**?**] have been used e.g. in randomness extraction [**?**] and compressed sensing [**?**].

4 Operator theory

The following basic mathematical problem of Kadison and Singer from 1959 [**?**] was intended to formalize a basic question of Dirac concerning the "universality" of measurements in quantum mechanics. We need a few definitions. Consider $B(\mathcal{H})$, the algebra of bounded linear operations on a Hilbert space $\mathcal{H}$, and define a *state* to be a linear, functional f on $B(\mathcal{H})$, normalized to $f(I) = 1$, which takes non-negative values on non-negative-definite operators). The states form a convex set, and a state is called *pure* if it is not a convex combination of other states. Finally, let D be the sub-algebra of diagonal operators in $B(\mathcal{H})$.

Kadison and Singer asked if every pure state on D has a *unique* extension to $B(\mathcal{H})$. This problem on infinite-dimensional operators found a host of equivalent formulations in finite dimensions, with motivations and intuitions from operator theory, discrepancy theory, Banach space theory, signal processing, and probability. All of them were solved affirmatively in recent work of Marcus, Spielman, and Srivastava [**?**] (which also surveys the many related conjectures). Here is one statement they prove, which implies the others.

Theorem 4.1. *For every $\epsilon > 0$, there is an integer $k = k(\epsilon)$ so that the following holds. Fix any n and any $n \times n$ matrix A with zeros on the diagonal and of spectral norm 1. Then there is a partition of $\{1, 2, \cdots, n\}$ into k subsets, $S_1, S_2, \cdots, S_k$, so that each of the principal minors A_i (namely A restricted to rows and columns in S_i) has spectral norm at most ϵ.*

This statement clearly implies that one of the minors has linear size, at least n/k. This consequence is known as the *Restricted Invertibility* Theorem of Bourgain and Tzafriri [**?**], itself an important result in operator theory.

How did computer scientists get interested in this problem? Without getting into too many details, here is a sketchy description of the meandering path which led to this spectacular result.

A central computational problem, at the heart of numerous applications, is solving a linear system of equations. While Gaussian elimination does the job quite efficiently (the number of arithmetic operations is about n^3 for $n \times n$ matrices), for large n this is still inefficient, and faster methods are sought, hopefully nearly linear in the number of non-zero entries of the given matrix. For Laplacian linear systems (arising in many graph theory applications, such as computing electrical flows and random walks), Spielman and Teng [?] achieved precisely that. A major notion they introduced was *spectral sparsifiers* of matrices (or weighted graphs). A sparsifier of a given matrix is another matrix with far fewer (indeed, linear) non-zero entries which nevertheless has essentially the same spectrum as the original (a very special case of this notion are *expander graphs*, which by definition are sparsifiers of complete graphs). Optimal constructions of sparsifiers in [?] lead in [?] to a new proof (with better analysis) of the Restricted Invertibility theorem above, making the connection to the Kadison-Singer problem.

However, the solution to Kadison-Singer seems to require another detour. The same team [?] first resolved a bold conjecture of Bilu and Linial [?] on the spectrum of signings of matrices. This conjecture was part of a plan for a simple, iterative construction of Ramanujan graphs, the best[3] possible expander graphs. Ramanujan graphs were introduced and constructed in [?, ?], but rely on deep results in number theory and algebraic geometry (believed by some to be essential for *any* such construction). To prove the Bilu-Linial conjecture (and indeed produce Ramanujan graphs of every possible degree—something the algebraic constructions couldn't provide), [?] developed a theory of *interlacing polynomials* that turned out to be the key technical tool for resolving Kadison-Singer. In both cases, the novel view is to think of these conjectures probabilistically, and analyze the norm of a random operator by analyzing the average characteristic polynomial. That this method makes sense and actually works is deep and mysterious, and moreover provides a new kind of existence proofs for which no efficient way of finding the desired objects is known.

5 Metric Geometry

How close one metric space is to another is captured by the notion of *distortion*, measuring how distorted distances of one become when embedded into the other. More precisely,

Definition 5.1. Let (X, d) and (X', d') be two metric spaces. An embedding $f : X \to X'$ has distortion $\leq c$ if for every pair of points $x, y \in X$ we have

$$d(x, y) \leq d'(f(x), f(y)) \leq c \cdot d(x, y).$$

When X is finite and of size n, we allow $c = c(n)$ to depend on n.

Understanding the best embeddings between various metric and normed spaces has been a long endeavor in Banach space theory and metric geometry. An example of one major result in this area is Bourgain's embedding theorem [?].

Theorem 5.2. *Every metric space of size n can be embedded into Euclidean space L_2 with distortion $O(\log n)$.*

As computations are more efficient in some spaces than others, approximation algorithms gave rise to many new embedding questions. However, a real shift in the evolution of this field and the level of interactions between geometers and TCS researchers came from trying to prove "hardness of approximation" results. We exemplify this with the Goemans-Linial conjecture [?, ?] about the relation between L_1 and the "negative type" metric space L_2^2 (in which Euclidean distances are squared).

Conjecture 5.3. L_2^2 can be embedded into L_1 with constant distortion.

This conjecture was proved false by Khot and Vishnoi [?], who proved

Theorem 5.4. *For every n there are n-point subsets of L_2^2 for which every embedding to L_1 requires distortion $\Omega(\log \log n)$.*

[3] With respect to the spectral gap. This is one of a few important expansion parameters to optimize

Far more interesting than the result itself is its origin. Khot and Vishnoi were trying to prove that the "max-cut" problem[4] is hard to approximate. They managed to do so under an assumption, known as the *Unique Games* conjecture of Khot [?]. However, applying the reduction used in this proof to a particular instance of the max-cut problem, they obtained the hard-to-embed spaces above *unconditionally.*

This distortion lower bound above was recently improved by Cheeger, Kleiner, and Naor [?] to $\Omega(\log n)^{\delta}$ for some constant $\delta > 0$. We note that a highly nontrivial $\sqrt{\log n}$ upper bound on the distortion follows from the algorithm of Arora, Rao, and Vazirani [?].

Another powerful connection between such questions and TCS is through (again) expander graphs. A basic example is that the graph metric of any constant-degree expander proves that Bourgain's embedding theorem above is optimal! Much more sophisticated examples arise from trying to understand (and perhaps disprove) the Novikov and the Baum-Connes conjectures (see [?]). This program relies on a another, much weaker notion of *coarse* embedding.

Definition 5.5. (X, d) has a coarse embedding into (X', d') if there is a map $f : X \to X'$ and two increasing, unbounded real functions α, β such that for every two points $x, y \in X$,

$$\alpha(d(x,y)) \le d'(f(x), f(y)) \le \beta(d(x,y)).$$

Gromov [?] was the first to construct a metric (the word metric of a group) which cannot be coarsely embedded into a Hilbert space. His construction uses an infinite family of Cayley expanders (graphs defined by groups). This result was greatly generalized recently, by Mendel-Naor [?] and by Lafforgue [?], who constructed graph metrics that cannot be coarsely embedded into any *uniformly convex* space. It is interesting that while Lafforgue's method is algebraic, the Mendel-Naor construction follows the combinatorial *zig-zag* construction of expanders [?].

6 Group Theory

Group theorists, much like number theorists, were intrinsically interested in computational problems since the origin of the field. For example, the *word problem* (given a word in the generators of some group, does it evaluate to the trivial element?) is so fundamental to understanding any group one studies, that as soon as language was created to formally discuss the computational complexity of this problem, hosts of results followed trying to pinpoint that complexity. These include decidability and undecidability results once Turing set up the theory of computation and provided the first undecidable problems, and these were followed with NP-completeness results and efficient algorithms once P and NP were introduced around 1970. Needless to say, these *algorithmic results* encode pure *structural* complexity of the groups at hand. And the word problem is but the first example. A huge body of work today is devoted to finding efficient algorithms for computing commutator subgroups, Sylow subgroups, centralizers, bases, representations, characters, and a host of other important substructures of a group from some natural description of it. A thorough textbook is [?].

Here we focus on two related problems, the *generation* and *random generation* problems, and new conceptual notions borrowed from computational complexity which are essential for studying them. Before defining them formally (below), let's consider an example. Assume I hand you 10 invertible matrices, say 100×100 in size, over the field of size 3. Can you tell me if they generate another such given matrix? Can you even produce convincing evidence of this before we both perish? How about generating a random matrix in the subgroup spanned by these generators? The problem, of course, is that this subgroup will have size far larger than the number of atoms in the known universe, so its elements cannot be listed, and typical words generating elements in the group may need to be prohibitively long. Indeed, even the extremely special cases, for elements in $\mathbb{Z}_p^*$ (namely one, 1×1 matrix), the first question is related to the *discrete logarithm* problem, and for $\mathbb{Z}_{p \cdot q}^*$ it is related to the *integer factoring* problem, both currently requiring exponential time to solve (as a function of the description length).

[4] A "clustering-type" problem in which the vertices of a graph should be partitioned so as to maximize the number of edges between them

Let us consider any finite group G and let $n \approx \log|G|$ be roughly the length of a description of an element of G. Assume we are given k elements in G, $S = \{s_1, s_2, \cdots, s_k\}$. It would be ideal if the procedures we describe would work in time polynomial in n and k (which prohibits enumerating the elements of G, whose size is exponential in n).

The *generation problem* asks if a given element $g \in G$ is generated by S. How does one prove such a fact? A standard certificate for a positive answer is a *word* in the elements of S (and their inverses) which evaluates to g. However, even if G is cyclic, the shortest such word may be exponential in n. An alternative, computationally motivated description, is to give a *program* for g. Its definition shows that the term "program" suits it perfectly, as it has the same structure as usual computer programs, only that instead of applying some standard Boolean or arithmetic operations, we use the group operations of multiplication and inverse.

Definition 6.1. A program (over S) is a finite sequence of elements $g_1, g_2, \cdots, g_m$, where every element g_i is either in S, or is the inverse of a previous g_j, or is the product of previous g_j, g_ℓ. We say that it computes g simply if $g = g_m$.

In the cyclic case, programs afford exponential savings over words in description length, as a program allows us to write large powers by repeatedly squaring elements. What is remarkable is that such savings are possible for *every* group. This discovery of Babai and Szemeredi [?] says that every element of every group has an extremely succinct description in terms of any set of elements generating it.

Theorem 6.2. *For every group G, if a subset of elements S generates another element g, then there is a program of length at most $n^2 \approx (\log|G|)^2$ which computes g from S.*

It is interesting to note that the proof uses a structure which is very combinatorial and counterintuitive for group theorists: that of a *cube*, which we will see again later. For a sequence $(h_1, h_2, \cdots, h_t)$ of elements from G, the cube $C(h_1, h_2, \cdots, h_t)$ is the (multi)set of 2^t elements $\{h_1^{\epsilon_1}, h_2^{\epsilon_2}, \cdots, h_t^{\epsilon_t}\}$, with $\epsilon_i \in \{0,1\}$. Another important feature of the proof is that it works in a very general setting of "black-box" groups—it never needs an explicit description of the host group, only the ability to multiply elements and take their inverses. This is a very important paradigm for arguing about groups, and will be used again below.

How does one prove that an element g is *not* generated by S? It is possible that there is no short "classical" proof! This question motivated Babai to define Arthur-Merlin games—a new notion of probabilistic, interactive proofs (simultaneously with Goldwasser, Micali, and Rackoff [?], who proposed a similar notion for cryptographic reasons), and showed how non-membership can be certified in this new framework. The impact of the definition of interactive proofs on the theory of computation has been immense, but is beyond the scope of this survey.

Returning to the generation problem, let us now consider the problem of *random generation*. Here we are given S, and would like a randomized procedure which will quickly output an (almost) uniform distribution on the subgroup H of G generated by S. This problem, besides its natural appeal, is often faced by computational group theorists. It is clear that sufficiently long random words in the elements of S and its inverses will do the job, but just as with certificates, sufficiently long is often prohibitively long. In a beautiful paper, Babai [?] describes a certain process generating a random program which computes a nearly-uniform element of H, and runs in time $n^5 \approx (\log|G|)^5$ steps. It again uses cubes, and works in the full generality of black-box groups. This paper was followed by even faster algorithms with simpler analysis by Cooperman and by Dixon [?, ?], and the state-of-art is an algorithm whose number of steps is remarkably the same as the length of proofs of generation above—in other words, randomness achieves the efficiency of non-determinism for this problem.

7 Mathematical Physics

The field of mathematical physics is huge, and we focus here mainly on connections of statistical mechanics with the theory of computation. Numerous mathematical models exist of various physical and chemical systems, designed to understand basic properties of different materials and the dynamics of basic processes.

These include such familiar models as Ising, Potts, Monomoer-Dimer, Spin-Glass, Percolation, etc. A typical example explaining the connection of such mathematical models to physics and chemistry, and the basic problems studied is the seminal paper of Heilmann and Lieb [**?**].

Many of the problems studied can be viewed in the following general setting. We have a huge (exponential) space of objects called Ω (these objects may be viewed as the different configurations of a system). Each object is assigned a nonnegative weight (which may be viewed as the "energy" of that state). Scaling these weights gives rise to a probability distribution (often called the Gibbs distribution) on Ω, and to study its properties (phase transitions, critical temperatures, free energy, etc.) one attempts to generate samples from this distribution (if the description of a state takes n bits, then listing all probabilities in question is exponentially prohibitive).

As Ω may be highly unstructured, the most common approach to this sampling problem is known as "Monte Carlo Markov Chain" (or "MCMC") method. The idea is to build a graph on the objects of Ω, with a pair of objects connected by an edge if they are similar in some sense (e.g. sequences which differ only in a few coordinates). Next, one starts from any object, and performs a biased random walk on this graph for some time, and the object reached is the sample produced. In many settings it is not hard to set up the random walk (often called Glauber dynamics or the Metropolis algorithm) so that the *limiting* distribution of the Markov chain is indeed the desired distribution. The main question in this approach is *when* to stop the walk and output a sample; *when* are we close enough to the limit? In other words, how long does it take the chain to converge to the limit? In most cases, these decisions were taken on intuitive, heuristic grounds, without rigorous analysis of convergence time. The exceptions where rigorous bounds were known were typically structured, e.g. where the chain was a Cayley graph of a group (e.g. [**?**, **?**]).

This state of affairs has changed considerably since the interaction in the past couple of decades with the theory of computation. Before describing it, let us see where computational problems even arise in this field. The two major sources are *optimization* and *counting*. That the setting above suits many instances of optimization problems is easy to see. Think of Ω as the set of solutions to a given optimization problem (e.g. the values of certain parameters designed to satisfy a set of constraints), and the weights representing the quality of a solution (e.g. the number of constraints satisfied). So, picking at random from the associated distribution favors high quality solutions. The counting connection is more subtle. Here Ω represents a set of combinatorial objects one wants to count or approximate (e.g. the set of perfect matchings in a graph, or satisfying assignments to a set of constraints). It turns out that for many such settings, sampling an object (approximately) at random allows a recursive procedure to approximate the size of the set [**?**]. Moreover, viewing the finite set as a fine discretization of a continuous object (e.g. lattice points in a convex set) allows one to compute volumes and more generally integrate functions over such domains.

Around 1990, rigorous techniques were introduced [**?**, **?**, **?**, **?**] to analyze the convergence rates of such general Markov chains arising from different approximation algorithms. They establish *conductance* bounds on the Markov chains, mainly via *canonical paths* or *coupling* arguments (a survey of this early work is [**?**]). Collaborative work was soon able to formally justify the physical intuition behind some of the suggested heuristics for many models, and moreover drew physicists to suggest such ingenious chains for optimization problems. The field drew in probabilists and geometers as well, and by now is highly active and diverse. We mention two results to illustrate rigorous convergence bounds for important problems of this type.

Theorem 7.1 ([**?**]). *The permanent of any nonnegative $n \times n$ matrix can be approximated, to any precision, in polynomial time in n.*

Here the importance stems from the seminal result of Valiant [**?**] that the permanent[5] function is *universal* for essentially all counting problems (in particular those arising in the statistical physics models and optimization and counting problems above).

Theorem 7.2 ([**?**]). *The volume of any convex set in n dimensions can be approximated, to any precision, in polynomial time in n.*

[5] The notorious sibling of the determinant, in which no signs appear

Another consequence of this collaboration was a deeper understanding of the underlying physical models. For example, we now know that in many of them *spacial* properties (such as long-term correlations between distant sites) are directly related to the *temporal* speed of convergence of the underlying chain. The close similarity between statistical physics models and optimization problems, especially on random instances, has unraveled the fine geometric structure of the space of solutions, e.g. in [**?**]. At the same time, physics intuition based on such ideas as renormalization, annealing, and replica symmetry breaking, has led to new algorithms for optimization problems. For example, the fastest (yet unproven) heuristics for such problems as Boolean Satisfiability (which is NP-complete in general) use a physics method of "survey propagation" of [**?**].

8 Analysis and Probability

This section gives a taste of a growing number of families of inequalities—large deviation inequalities, isoperimetric inequalities, etc.—that have been generalized beyond their classical origins due to a variety of motivations in the theory of computing and discrete mathematics. Further, the applications sometimes call for *stability* versions of these inequalities, namely an understanding of the structures which make an inequality nearly sharp. Here too these motivations pushed for generalizations of classical results and many new ones.

The following exemplifying story can be told from several angles. One is *noise sensitivity* of functions. We restrict ourselves to the Boolean cube endowed with the uniform probability measure, but many of the questions and results extend to arbitrary product probability spaces. Let $f : \{-1, 1\}^n \to \mathbb{R}$, which we assume is balanced, namely $E[f] = 0$. When the image of f is $\{-1, 1\}$, we can think of f as a voting scheme, translating the binary votes of n individuals into a binary outcome. One natural desire from such a voting scheme may be *noise stability*—that typically very similar inputs (vote vectors) will yield the same outcome. While natural in this social science setting, such questions also arises in statistical physics settings, where natural functions such as bond percolation turn out to be extremely sensitive to noise [**?**]. Let's formally define noise stability.

Definition 8.1. Let $\rho \in [0, 1]$ be a correlation parameter. We call two vectors $x, y \in \{-1, 1\}^n$ are ρ-correlated if they are distributed as follows. The vector x is drawn uniformly at random, and y is obtained from x by flipping each bit x_i independently with probability $(1 - \rho)/2$. Note that for every i the correlation $E[x_i y_i] = \rho$. The *noise sensitivity* of f at ρ, $S_\rho(f)$, is simply defined as the correlation of the outputs, $E[f(x)f(y)]$.

It is not hard to see that the function maximizing noise stability is any *dictatorship* function, e.g. $f(x) = x_1$, for which $S_\rho(f) = \rho$. But another natural social scientific concern is the *influence* of players in voting schemes [**?**], which prohibits such solutions (in democratic environments). Influence of a voter is the probability with which it can change the outcome given that all other votes are uniformly random (so, in a dictatorship it is 1 for the dictator and 0 for all others). A fair voting scheme should have no voter with high influence. As we define influence for Real-valued functions, we will use the (conditional) *variance* to measure a player's potential effect given all other (random) votes.

Definition 8.2. A function $f : \{-1, 1\}^n \to \mathbb{R}$ has influence τ if for every i, $\mathrm{Var}[x_i | x_{-i}] \leq \tau$ for all i (where x_{-i} denotes the vector x without the ith coordinate).

For example, the majority function has influence $O(1/\sqrt{n})$. The question of how small the influence of a balanced function can be is extremely interesting, and leads to a highly relevant inequality for our story (both in content and techniques). As it turns out, ultimate fairness (influence $1/n$ per player) is impossible—[**?**] show that every function has a player with unproportional influence, at least $\Omega(\log n/n)$. At any rate, one can ask which of the functions with *small* influence is most stable, and it is natural to guess that majority should be the best[6].

[6] This noise sensitivity tends to, as n grows, to $S_\rho(Majority_n) = \frac{2}{\pi} \arcsin \rho$.

The conjecture that this is the case, called the *Majority is Stablest* conjecture, arose from a completely different and surprising angle - the field of optimization, specifically "hardness of approximation". A remarkable paper [?] has shown that it implies[7] the optimality of a certain natural algorithm for approximating the maximum cut of a graph (the partition of vertices so as to maximize the number of edges between them—a basic optimization problem whose exact complexity is NP-complete). This connection is highly non-trivial, but by now we have many examples showing how the analysis of certain (semidefinite programming-based) approximation algorithms for a variety of optimization problems raise many new isoperimetric questions, enriching this field.

The Majority is Stablest conjecture was proved in a strong form by [?] shortly after it was posed. Here is a formal statement (which actually works for bounded functions).

Theorem 8.3. *For every (positive correlation parameter) $\rho \geq 0$ and $\epsilon > 0$ there exists (an influence bound) $\tau = \tau(\rho, \epsilon)$ such that for every n and every $f : \{-1,1\}^n \to [-1,1]$ of influence at most τ, $S_\rho(f) \leq S_\rho(Majority_n) + \epsilon$.*

The proof reveals another angle on the story—large deviation inequalities and invariance principles. To see the connection, recall the Berry-Esseen theorem [?], generalizing the standard central limit theorem to *weighted* sums of independent random signs. In this theorem, influences arise very naturally. Consider $\sum_{i=1}^n c_i x_i$. If we normalize the weights c_i to satisfy $\sum_i c_i^2 = 1$, then c_i is the influence of the ith voter, and $\tau = \max_i |c_i|$. The quality of this central limit theorem deteriorates linearly with the influence τ. Lindeberg's proof of Berry-Esseen uses an invariance principle, showing that for linear functions, the cumulative probability distribution $Pr[\sum_{i=1}^n c_i x_i \leq t]$ (for every t) is unchanged (up to τ), *regardless* of the distribution of the variables x_i, as long as they are independent and have expectation 0 and variance 1. Thus, in particular, they can be taken to be standard Gaussian, which trivializes the problem, as the weighted sum is a Gaussian as well!

To prove their theorem, [?] first observed that also in the noise stability problem, the Gaussian case is simple. If the x_i, y_i are standard Gaussians with correlation ρ, the stability problem reduces to a classical result of Borell [?]: that noise stability is maximized by any hyperplane through the origin. Note that here the rotational symmetry of multidimensional Gaussians, which also aids the proof, does not distinguish "dictator" functions from majority—both are such hyperplanes. Given this theorem, an invariance principle whose quality depends on τ would do the job. They next show that it is sufficient to prove the principle only for *low degree* multilinear polynomials (as the effect of noise decays with the degree). Finally, they prove this non-linear extension of Berry-Esseen for such polynomials, a form of which we state below. They also use their invariance principle to prove other conjectures, and since the publication of their paper, quite a number of further generalizations and applications were found.

Theorem 8.4. *Let x_i be any n independent random variables with mean 0, variance 1 and bounded 3rd moments. Let g_i be n independent standard Gaussians. Let Q be any degree d multilinear n-variate polynomial of influence τ. Then for any t,*

$$|Pr[Q(x) \leq t] - Pr[Q(g) \leq t]| \leq O(d\tau^{1/d}).$$

Most of the material above, and much more on the motivations and developments in this exciting area of the analysis of Boolean functions, is in this book [?] by O'Donnell.

9 Lattice Theory

Lattices in Euclidean space are among the most "universal" objects in mathematics, in that besides being natural (e.g. arising in crystalline structures) and worthy of study in their own right, they capture a variety of problems in different fields such as number theory, analysis, approximation theory, Lie algebras, convex geometry, and more. Many of the basic results in lattice theory, as we shall exemplify, are *existential* (namely

[7] Assuming another, complexity-theoretic, conjecture called the "Unique Games" conjecture

supply no efficient means for obtaining the objects whose existence is proved), which in some cases limited progress on these applications.

This section tells the story of one algorithm, of Lenstra, Lenstra, and Lovasz [**?**], often called the LLL algorithm, and some of its implications on these classical applications as well as modern ones in cryptography, optimization, and more. But we had better define a lattice[8] first.

Let $B = \{b_1, b_2, \ldots, b_n\}$ be a basis of $\mathbb{R}^n$. Then the *lattice* $L(B)$ denotes the set (indeed, Abelian group) of all *integer* linear combinations of these vectors, i.e. $L(B) = \{\sum_i z_i b_i \ : \ z_i \in \mathbb{Z}\}$. B is also called the basis of the lattice. Naturally, a given lattice can have many different bases, e.g. the standard integer lattice in the plane, generated by $\{(0,1),(1,0)\}$, is equally well generated by $(999,1),(1000,1)$. A basic invariant associated with a lattice L is its determinant $d(L)$, which is the absolute value of $\det(B)$ for any basis B of L (this is also the volume of the fundamental parallelpiped of the lattice). For simplicity and without loss of generality, we will assume that B is normalized so that we only consider lattices L of $d(L) = 1$.

The most basic result about lattices, was proved by Minkowski (who initiated this field, and with it, Geometry of Numbers) [**?**].

Theorem 9.1. *Consider an arbitrary convex set K in $\mathbb{R}^n$ which is centrally symmetric[9] and has volume $> 2^n$. Then, every lattice L (of determinant 1) has a nonzero point in K.*

This innocent and easy to prove theorem has only an existential (pigeonhole) proof, but already implies (for appropriate norms and lattices) such fundamental results as the finiteness of class numbers of number fields (see e.g. [**?**]).

The most basic questions one can ask about a lattice are about its short vectors. Of course, "short" depends on the norm used. Another basic result of Minkowski [**?**] concerns (the most natural) Euclidean norm.

Theorem 9.2. *Every lattice L has a nonzero point of Euclidean norm at most $\sqrt{n}$.*

Again, this proof is existential, and the obvious algorithm for finding such a point would require exponential time in n. Described in the breakthrough paper above [**?**], the LLL algorithm is an efficient, polynomial-time algorithm, which produces a relatively "short" vector of length at most 2^n. This may seem excessive at first, but the number and diversity of applications is staggering. First, in many problems, the dimension n is a small constant (so the actual input length arises from the bit-size of the given basis). This leads, for instance, to Lenstra's algorithm for (exactly solving) Integer Programming [**?**] in constant dimensions, Odlyzko and Riele's refutation of Mertens' conjecture about cancellations in the Möbius function [**?**], and the long list of number theoretic examples in [**?**]. But it turns out that even when n is arbitrarily large, many problems can be solved in poly(n)-time as well. Here is a list of examples of old and new problems representing this variety, some going back to the original paper [**?**]. In all, it suffices that real number inputs are approximated to poly(n) digits in dimension n.

- Diophantine approximation. While the best possible approximation of one real number by rationals with bounded denominator is readily solved by its (efficiently computable) continued fraction expansion, no such procedure is known for *simultaneous* approximation. Formally, given a *set* of real numbers, say $\{r_1, r_2, \ldots r_n\}$, and a bound Q and $\epsilon > 0$, find integers $q \leq Q$ and $p_1, \ldots, p_n$ such that all $|r_i - p_i/q| \leq \epsilon$. Existentially, the Dirichlet "box-principle" shows that $\epsilon < Q^{1/n}$ is possible. Using LLL, one efficiently obtains $\epsilon < 2^{n^2} Q^{1/n}$ which is meaningful for Q described by poly(n) many bits.

- Minimal polynomials of algebraic numbers. Here we are given a single real number r and a degree bound n, and are asked if there is a polynomial $g(x)$ with integer coefficients, of degree at most n of which r is a root (and actually produce such a g if it exists). Indeed, this is a special case of the problem above with $r_i = r^i$. While the algorithm only outputs g for which $g(r) \approx 0$, it is often easy to check that it actually vanishes. Note that by varying n we can find the minimal such polynomial.

[8] We only define full-rank lattices here, which suffice for this exposition.

[9] Namely, $x \in K$ implies that also $-x \in K$. Such sets are precisely balls in arbitrary norms.

- Polynomial factorization over Rationals. Here the input is an integer polynomial h of degree n, and we want to factor it over $\mathbb{Q}$. The high level idea is to first find an (approximate) root r of h (e.g. using Newton's method), feed it to the problem above, which will return a minimal g having r as a root, and thus divides h. We stress that this algorithm produces the exact factorization, not an approximate one!

- Small integer relations between reals. Given reals $r_1, r_2, \ldots r_n$, and a bound Q, determine if there exist integers $|z_i| < Q$ such that $\sum_i z_i r_i = 0$ (and if so, find these integers). As a famous example, LLL can find an integer relation among $\arctan(1) \approx 0.785398, \arctan(1/5) \approx 0.197395$ and $\arctan(1/239) \approx 0.004184$, yielding Machin's formula

$$\arctan(1) - 4\arctan(1/5) + \arctan(1/239) = 0$$

- Cryptanalysis. Note that a very special case of the problem above (in which the coefficients z_i must be Boolean) is the "Knapsack problem," a famous NP-complete problem. The point here is that in the early days of cryptography, some systems were based on the assumed "average case" hardness of Knapsack. Many such systems were broken by using LLL, e.g. [**?**]. LLL was also used to break some versions of of the RSA cryptosystem (with "small public exponents").

It is perhaps a fitting epilogue to the last item that lattices cannot only destroy cryptosystems, but also create them. The problem of efficiently approximating short vectors up to polynomial (as opposed to exponential, as LLL produces) factors is assumed computationally hard. Ajtai showed in a remarkable paper [**?**] that such hardness is preserved "on average", over a cleverly-chosen distribution of random lattices. This led to a new public-key encryption scheme [**?**] based on this hardness, which is essentially the only one known that can potentially sustain quantum attacks (Shor's efficient quantum algorithms can factor integers and compute discrete logarithms [**?**]). In a recent breakthrough of Gentry [**?**], such hardness leads to *fully homomorphic* encryption, which allows to compute on secret data.

10 Invariant Theory

The problems we discuss here are more generally concerned with connections between computational complexity with algebraic geometry and representation theory. We will mention these later, and proceed with the "invariant-theoretic" aspect, which is the simplest to describe in an elementary way.

Invariants are familiar enough. In high school physics we learn that energy and momentum are preserved (namely, are *invariants*) in the dynamics of general physical systems. A classical puzzle asks when a plane polygons can be "cut and pasted" along straight lines to another polygon - here the obvious invariant, *area*, is the only one. However in generalizing this puzzle to 3-dimensional polyhedra, it turns out that besides the obvious invariant, *volume*, there is another (discovered by Dehn). Questions about the equivalence of two surfaces under homeomorphism, whether two groups are isomorphic, or whether two points are in the same orbit of a dynamical system are naturally treated in this way.

We will focus on invariants of linear group actions. Fix a field $\mathbb{F}$ (while problems are interesting in every field, results mostly work for infinite fields only, and sometimes just for characteristic zero or algebraically closed ones). Let G be a group, and V a representation of G, namely an $\mathbb{F}$-vector space on which G acts; for every $g, h \in G$ and $v \in V$ we have $gv \in V$ and $g(hv) = (gh)v$. When G acts on V, it also acts on $\mathbb{F}[V]$, the polynomial functions on V, also called the *coordinate ring* of V. In our setting V will have finite dimension (say m), and so $\mathbb{F}[V]$ is simply $\mathbb{F}[x_1, x_2, \ldots, x_m] = \mathbb{F}[X]$, the polynomial ring over $\mathbb{F}$ in m variables. We will by denote gp the action of a group element g on a polynomial $p \in \mathbb{F}[V]$.

A polynomial $p(X) \in \mathbb{F}[X]$ is *invariant* if it is unchanged by this action, namely for every $g \in G$ we have $gp = p$. All invariant polynomials clearly form a subring of $\mathbb{F}[X]$, denoted $\mathbb{F}[X]^G$, called the ring of invariants of this action. Understanding the invariants of group actions is the main subject of Invariant Theory. In our setting, all these rings will be *finitely generated*, in the sense that there will be a finite set of polynomials

$\{q_1, q_2, \ldots, q_t\}$ in $\mathbb{F}[X]^G$ so that for every polynomial $p \in \mathbb{F}[X]^G$ there is a t-variate polynomial r over $\mathbb{F}$ so that $p = r(q_1, q_2, \ldots, q_t)$. Finding the "simplest" such generating set of invariants is our concern.

Two familiar examples of perfect solutions to this problem follow. In the first, $G = S_m$, the symmetric group on m letters, acting on the set of m formal variables X (and hence the vector space they generate) by simply permuting them. Then a set of generating invariants are simply the first m *elementary* symmetric polynomials. In the second, $G = SL_n(\mathbb{F})$ acting on the vector space of $n \times n$ matrices in $M_n(\mathbb{F})$, simply by matrix multiplication. In this case all polynomial invariants are generated by the determinant.

In these two cases, there are few invariants, they have low degree, and they are easy to compute—all these quantities are bounded by a polynomial in m, the dimension of the space. If this is the case, one can quickly tell if two polynomials are in the same orbit, and more importantly for algebraic geometric situations, also tell (with some more work) if the orbit *closures* of two polynomials intersect.

Indeed, it is this last problem which supplies one computational complexity interest, that we now briefly explain. Valiant [?] defined arithmetic analogs of P and NP, and showed (via surprising completeness results) that to separate them it is sufficient to prove that the *permanent* polynomial on $n \times n$ matrices does not project to the *determinant* polynomial on $m \times m$ matrices for any $m = \text{poly}(n)$. In a series of papers, Mulmuley and Sohoni (see e.g. [?, ?] for surveys) introduced Geometric Complexity Theory (GCT) to tackle this major open problem. They cast it, changing the permanent a bit, as an orbit closure intersection problem of the orbits of these two fundamental polynomials, under the action of the linear group SL_{m^2}. To date, the tools of algebraic geometry and representation theory do not seem strong enough.

One limitation to their program is the current efficiency of basic algebraic-geometric procedures, most of which use the Gröbner basis algorithm, which is typically prohibitively slow. A particular problem of interest is exactly the search for simple (in all these respects above) generators of the invariant ring $\mathbb{F}[X]^G$. We restrict ourselves to actions of linear groups G, where finite generation is a classical result of Hilbert [?]. Indeed, Hilbert proved the so-called "Noether's normalization lemma"[10], showing how to compute a minimal set of generators for any such representation (geometrically, this translates to embedding an algebraic variety of dimension m into $(m+1)$-dimensional affine space, as a finite cover). However, even in simple specific cases, it is infeasible to describe the map (or list a small set of generators).

This problem was resolved for "explicit varieties" in the paper of Mulmuley [?] (who lists many examples of natural such varieties) and the subsequent [?]. In particular, a natural general setting is that of *quivers* and their representations [?]. We take the simplest quiver, namely the action of $SL_n(\mathbb{F})$ on d-tuples of $n \times n$ matrices. So we have $m = dn^2$ variables $X = (X_1, X_2, \ldots, X_d)$. The action of a matrix $A \in SL_n$ on this tuple is by simultaneous conjugation, by transforming it to the tuple $(A^{-1}X_1A, A^{-1}X_2A, \cdots, A^{-1}X_dA)$. Now, which polynomials in X are invariant under this action? The work of Procesi, Formanek, Razmyslov, and Donkin [?, ?, ?, ?] provides a good set of generating invariants.

The generators are simply the traces of products of length at most n^2 of the given matrices. Namely the set

$$\{Tr(X_{i_1}X_{i_2}\cdots X_{i_t}) \,:\, t \leq n^2\}.$$

These polynomials are explicit, have small degree and are easily computable. The one shortcoming is the *exponential* size of this set. By Hilbert we know that it can, in principle, be reduced to dn^2. Using the best Gröbner basis algorithm of [?] (another collaborative effort) this will take double exponential time in n. The works above [?, ?] reduce it to polynomial time! Indeed, [?] gave a probabilistic polynomial time algorithm, and [?] derandomized a variant of it to yield a deterministic one.

The field of invariant theory, and indeed commutative algebra, are rife with computational problems and algorithms – one excellent text on this subject is [?].

Acknowledgements

I am grateful to Andy Drucker, Peter Sarnak, Amir Shpilka and Yuval Wigderson for careful reading of earlier drafts of this paper.

[10] He used it for his Nullstellensatz and basis theorems.

References

[1] Aaronson, S., *Quantum Computing since Democritus*, Cambridge University Press, 2013.

[2] Achlioptas, D., Coja-Oghlan, A., and Ricci-Tersenghi, F., On the solution-space geometry of random constraint satisfaction problems. *Random Structures and Algorithms* **38** (3) (2011), 251–268.

[3] Adleman, L., and Huang, M., *Primality testing and Abelian varieties over finite field*, Lecture Notes in Mathematics, Volume 1512, Springer-Verlag, 1992.

[4] Aldous, D., Random walks on finite groups and rapidly mixing Markov chains, *Se?minaire de Probabilite?s XVII*, Springer Lecture Notes in Mathematics **986** (1981/82), 243–297.

[5] Aldous, D., The random walk construction for spanning trees and uniform labeled trees, *SIAM Journal on Discrete Mathematics* **3** (1990), 450–465.

[6] Adleman, L., Pomerance, C., and Rumely, R., On distinguishing prime numbers from composite numbers. *Annals of Mathematics* **117** (1) (1983), 173–206.

[7] Agrawal, M., and Biswas, S., Primality and Identity Testing via Chinese Remaindering. *Journal of the ACM* **50** (4) (2003), 429–443.

[8] Agrawal, M., Kayal, N., and Saxena, N., Primes is in P. *Ann. of Math.* **160** (2) (2004), 781–793.

[9] Ajtai, M., Generating hard instances of lattice problems (extended abstract). *Proc. of the 28th Annual ACM Symposium on Theory of Computing (STOC)*, (1996), 99–108.

[10] Ajtai, M., and Dwork, C., A public-key cryptosystem with worst-case/average-case equivalence. *Proc. 29th ACM STOC* (1997), 284–293.

[11] Arora, S., and Barak, B., *Computational Complexity: A Modern Approach*, Cambridge University Press, 2009.

[12] Arora S., Rao S., and Vazirani U., Expander rows, geometric embeddings and graph partitioning. *Proc. of the 36th Annual Symposium on Theory of Computing*, ACM Press, New York 2004, 222-231.

[13] Babai, L., and Szemerédi., On the complexity of matrix group probelms, I. *Proc. 25th IEEE Symposium on Foundations of Computer Science* (1984), 229–240.

[14] Babai, L., Local expansion of vertex-transitive graphs and random generation in finite groups. *23rd ACM Symposium on Theory of Computing (STOC)*, New York 1991, 164–174.

[15] Bach, E., and Shallit, J., Algorithmic Number Theory, Efficient Algorithms. Volume 1 (2nd Ed.), MIT Press, 1997.

[16] Barak, B., Dvir, Z., Wigderson, A., and Yehudayoff, A., Fractional Sylvester-Gallai theorems. *Proc. of the National Academy of Sciences of the United States of America* (2012).

[17] Barak, B., Impagliazzo, R., and Wigderson, A., Extracting randomness using few independent sources. *SICOMP* **36** (4) (2006), 1095–118.

[18] Broder, A.Z., Generating random spanning trees, *Proceedings of the 30th Annual IEEE Symposium on Foundations of Computer Science*, (1989) 442–447.

[19] Batson, J., Spielman D., and Srivastava N., Twice-Ramanujan Sparsifiers. *SIAM Review* **56** (2) (2014), 315–334.

[20] Ben-Or, M., and Linial, N., Collective Coin Flipping, Robust Voting Schemes and Minima of Banzhaf Vlaues. *FOCS 1999* (1988), 68–80.

[21] Benjamini, I., Kalai, G., and Schramm, O., Noise Sensitivity of Boolean Functions and Applications to Percolation. *Publications Mathématiques de l'Institut des Hautes Études Scientifiques* **90** (1999), 5–43.

[22] Berlekamp, E. R., Factoring Polynomials Over Finite Fields. *Bell System Technical Journal* **46** (1967), 1853–1859.

[23] Besicovitch, A., Sur deux questions d'integrabilite des fonctions. *J. Soc. Phys. Math.* **2** (1919), 105–123.

[24] Bilu, Y., and Linial, N., Lifts, discrepancy and nearly optimal spectral gap. *Combinatorica* (**26**) (5) (2006), 495–519

[25] Borell, C., Positivity improving operators and hypercontractivity. *Math. Zeitschrift* **180** (2) (1982), 225–234.

[26] Borell, C., Geometric bounds on the Ornstein-Uhlenbeck velocity process. *Z. Wahrsch. Verw. Gebiete* **70** (1) (1985), 1–13.

[27] Bourgain J., On Lipschitz embedding of finite metric spaces in Hilbert space. *Israel J. Math.*, **52** (1-2) (1985), 46–52.

[28] Bourgain, J., Katz, N., and Tao T., A sum-product estimate in finite fields, and applications. *Geometric and Functional Analysis* **14** (1) (2004), 27–57.

[29] Bourgain, J., and Tzafriri, L., On a problem of Kadison and Singer. *J. Reine. Angew. Math* **420** (1991), 1–43.

[30] Cheeger J., Kleiner B., and Naor A., A $(\log n)^{\Omega(1)}$ integrality gap for the Sparsest Cut SDP. *FOCS 2009* (2009).

[31] Cooperman, G., *Towards a practical, theoretically sound algorithm for random generation in finite groups.* May 2002, available at arXiv:math/0205203.

[32] Cox, D., Little, J., and O'Shea, D. *Ideals, varieties, and algorithms*, Undergraduate Texts in Mathematics. Springer, New York, third edition, 2007.

[33] Davies, R. Some remarks on the Kakeya problem. *Proc. Cambridge Philos. Soc.* **69** (1971), 417–421.

[34] Diaconis, P.,. *Group representations in probability and statistics*, Lecture Notes Monograph Series Vol. 11, Institute of Mathematical Statistics, Hayward, CA, (1988).

[35] Dixon, J. D., Generating random elements in finite groups. *Electronic J. Comb.* **15** (2008), paper R94.

[36] Dyer, M., Frieze, A., and Kannan, R., A random polynomial time algorithm for approximating the volume of convex bodies, *Journal of the ACM* **38** (1991) 1–17.

[37] Dvir, Z., Kopparty, S., Saraf, S., and Sudan, M., Extensions to the Method of Multiplicities, with Applications to Kakeya Sets and Mergers. *SIAM J. on Computing* **42** (6) (2013), 2305–2328.

[38] Donkin, S. Invariants of several matrices. *Invent. Math.*, **110** (2) (1992), 389–401.

[39] Dvir, Z., On the size of Kakeya sets in finite fields. *J. Amer. Math. Soc.* **22** (2009), 1093–1097.

[40] Dvir, Z., From randomness extraction to rotating needles. *SIGACT News* **40** (4) (2009), 46–61.

[41] Dvir, Z., and Shpilka, A., Locally decodable codes with 2 queries and polynomial identity testing for depth 3 circuits. *SIAM J. on Computing* **36** (5) (2006), 1404–1434.

[42] Dyer, M., Frieze, A., and Kanna, R., A random polynomial-time algorithm for approximating the volume of convex bodies. *J. of the ACM* **38** (1) (1991), 1–17.

[43] Feller, W., An introduction to probability theory and its applications, Volume 2, Second edition. John Wiley and Sons Inc., New York, 1971.

[44] Forbes, M. and Shpilka, A., Explicit Noether Normalization for Simultaneous Conjugation via Polynomial Identity Testing. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, Springer Berlin Heidelberg, 2013, 527–542.

[45] Formanek, E. Invariants and the ring of generic matrices. *J. Algebra* **89** (1984), 178–223.

[46] Gabriel, P. Unzerlegbare Darstellungen I. *Manuscripta Mathematica* **6** (1) (1972), 71–103.

[47] Gentry, C., Fully homomorphic encryption using ideal lattices. *STOC 2009* (2009), 169–178.

[48] Goemans M. X., Semidefinite programming in combinatorial optimization. *Mathematical Programming* 79 (1997), 143–161.

[49] Goldreich, O., *Computational Complexity: A Conceptual Perspective*, Cambridge University Press (2008).

[50] Goldwasser, S., and Kilian, J., Almost All Primes Can Be Quickly Certified. *Proc. 18th STOC* (1986), 316-329.

[51] Goldwasser, S., Micali, S., and Rackoff, C., The Knowledge Complexity of Interactive Proof Systems. *SIAM J. Comput.* **18** (1) (1989), 186–208.

[52] Gromov, M., Hyperbolic groups. *Essays in Group Theory*, Math. Sci. Res. Inst. Publ. **8** (1987), 75–264.

[53] Guruswami, V., Lee, J., and Razborov, A., Almost Euclidean sections of L_1^N via expander codes. *Combinatorica* **30** (2010), 47–68.

[54] Guth, L., and Katz, N. H., *On the Erdos distinct distance problem in the plane*, June 2011, available at http://arxiv.org/abs/1011.4105.

[55] Heilmann, O., and Lieb, E., Theory of monomer-dimer systems. *Comm. Math. Phys.* **25** (3) (1972), 190–232.

[56] Hilbert, D. Uber die vollen Invariantensysteme. *Math. Ann.* **42** (1893), 313–370.

[57] Holt, D. F., Eick, B., and O'Brien, E. A. *Handbook of Computational Group Theory*, CRC Press, 2005.

[58] Jerrum, M., and Sinclair, A., The Markov chain Monte Carlo method: an approach to approximate counting and integration. *Approximation Algorithms for NP-hard Problems*, D.S. Hochbaum ed., PWS Publishing, Boston, 1996.

[59] Jerrum, M., Sinclair, A., and Vigoda E., A polynomial-time approzximation algorithm for the permanent of a matrix with non-negative entries. *J. of the ACM* **51** (2005), 671–697.

[60] Jerrum, M., Valiant L., and Vazirani V., Random Generation of Combinatorial Structures from a Uniform Distribution. *Theoretical Computer Science* **43** (1986), 169–201.

[61] Kadison, R. V., and Singer, I. M., Extensions of pure states. *American J. Math.* 81 (1959), 383–400.

[62] Kahn, J., Kalai, G., and Linial, N., The influences of Variables on Boolean Functions (Extended Abstract). *FOCS 1988* (1988), 68–80.

[63] Kasparov, G., and Yu G., The coarse geometric Novikov conjecture and uniform convexity. *Adv. Math.* **206** 1 (2006), 1–56.

[64] Kayal, N., and Saraf, S., Blackbox polynomial identity testing for depth-3 circuits. *Proc. of FOCS 2009*, (2009), 280–291.

[65] Khot S., and Vishnoi N., The unique games conjecture, integrality gap for cut problems and embeddability of negative type metrics into $\ell 1$. *Proc. of the 46th Annual IEEE Symposium on Foundations of Computer Science* (2005), 53–62.

[66] Kindler, G., Khot, S., O'Donnell, R., and Mossel, E., Optimal inapproximability results for MAX-CUT and other two-variable CSPs?. *SIAM J. Comput.* **37** (1) (2007), 319–357.

[67] Lafforgue V., Un renforcement de la propriété (T). *Duke Math. J.* **143** (3) (2008), 559–602.

[68] Lagarias J. C., Knapsack Public Key Cryptosystems and Diophantine Approximation. Advances in Cryptology, Plenum Press, New York 1984, 3–24.

[69] Lenstra H. W., Integer programming with a fixed number of variables. *Mathematics of Operations Research* **8** (4) (1983), 538–548.

[70] Lenstra, A. K., Lenstra Jr., H. W., and Lovász,L. Factoring polynomials with rational coefficients. *Math. Ann.* **261** (1982), 515–534.

[71] Linial, N., Finite metric spaces - combinatorics, geometry and algorithms. *Proc. of the International Congress of Mathematicians III* (2002), 573–586.

[72] Lu, C-J., Reingold, O., Vadhan, S., and Wigderson, A., Extractors: Optimal up to Constant Factors. *35th Annual ACM Sypmosium, STOC 2003*, (2003), 602–611.

[73] Lubotzky, A., Phillips R., and Sarnak P., Ramanujan graphs. *Combinatorica* **8** (3) (1988), 261–277.

[74] Marcus, A., Spielman D., and Srivastava N., Interlacing Families I: Bipartite Ramanujan Graphs of All Degrees. *FOCS 2013* (2013), 529–537.

[75] Marcus, A., Spielman D., and Srivastava N., *Interlacing Families II: Mixed Characteristic Polynomials and the Kadison-Singer Problem*, June 2013, available at arxiv.org/abs/1306.3969.

[76] Margulis, G. A., Explicit group-theoretic constructions of combinatorial schemes and their applications in the construction of expanders and concentrators. *Problems Information Transmission* **24** (1988), 39–46.

[77] Mayr, E. and Ritscher, S. Space effcient Gröbner basis computation without degree bounds. *Proceedings of ISAAC* (2011), 257–264.

[78] Mendel, M., and Naor, A., Nonlinear spectral calculus and super-expanders. *Publications mathématiques de l'IHÉS 119* **1** (2014), 1–95.

[79] Mezard, M., Parisi, G., and Zecchina R., Analytic and Algorithmic Solution of Random Satisfiability Problems. *Science* **297** (5582) (2002), 812–815.

[80] Miller, G.L., Riemann's Hypothesis and Tests for Primality. *J. Comput. System Sci.* **13** (3) (1976), 300–317.

[81] Minkowski, H., Geometrie der Zahlen (Erste Lieferung), Teubner, Leipzig, 1896.

[82] Mossel, E., O'Donnell, R., and Oleszkiewicz, K., Noise stability of functions with low influences: invariance and optimality. *Annals of Mathematics* **171** (1) (2010), 295–341.

[83] Mulmuley, K. D., On P vs. NP and geometric complexity theory. *J. of the ACM* **58** (2) (2011), 5:1–5:26.

[84] Mulmuley, K. D., The GCT program toward the P vs. NP problem. *Communications of the ACM* **55** (6) (2012), 98–107.

[85] Mulmuley, K. D. Geometric Complexity Theory V: Equivalence between blackbox derandomization of polynomial identity testing and derandomization of Noether's Normalization Lemma. *FOCS 2012* (2012), 629–638.

[86] O'Donnell, R. *Analysis of Boolean Functions*, Cambridge University Press, 2014.

[87] Odlyzko A. M., and Riele, H., Disproof of the Mertens conjecture. *J. Reine Angew. Math.* **357** (1985), 138–160.

[88] Papadimitriou, C. H., Computational Complexity, Addison-Wesley, 1993.

[89] Papadimitriou, C. H., NP-completeness: A retrospective. *Automata, languages and programming*, Lecture Notes in Comput. Sci. **1256** (1997), 2–6.

[90] Pohst, M., and Zassenhaus, H., Algorithmic Algebraic Number Theory No. 30, Cambridge University Press, 1989, pp. 384.

[91] Procesi, C. The invariant theory of $n \times n$ matrices. *Adv. Math.* **19** (1976), 306–381.

[92] Rabin, M. O., Probabilistic algorithm for testing primality. *J. Number Theory* **12** (1980), 128–138.

[93] Razborov, A. A., Lower bounds of monotone complexitybof the logical permanent function. *Mat. Zametki* **37** (6) (1985), 887–900; English transl. *Math. Notes* **37** (1985), 485–493.

[94] Reingold, O., Vadhan S., and Wigderson, A., Entropy waves, the zig-zag graph product, and a new constant–degree expanders. *Ann. of Math.* **155** (1) (2002), 157–187.

[95] Shor. P. W., Algorithms for quantum computation: Discrete logarithms and factoring. *Proc. 32nd Annual Symposium on Foundations of Computer Science* (Shafi Goldwasser, ed.) (1994), 124–134.

[96] Simon, D., Selected applications of LLL in number theory. *The LLL Algorithm* Information Security and Cryptography (2010), 265–282.

[97] Sinclair, A.J., and Jerrum, M.R., Approximate counting, uniform generation and rapidly mixing Markov chains, *Information and Computation* **82** (1989), 93–133.

[98] Solovay, R. M., and Strassen, V., A fast Monte-Carlo test for primality. *SIAM J. Comput.* **6** (1) (1977), 84–85.

[99] Spielman, D., and Srivastava N., An Elementary Proof of the Restricted Invertibility Theorem. *Israel J. Math* **190** (2012).

[100] Spielman, D. and Teng, S., Spectral Sparsification of Graphs. *SIAM J. on Computing* **40** (2011), 981–1025.

[101] Szemerédi, E., Trotter, W. T., Extremal problems in discrete geometry. *Combinatorica* **3** (3-4) (1983), 381–392.

[102] Tao, T., *Recent progress on the Kakeya conjecture*, May 2009, available at http://terrytao.wordpress.com/2009/05/11/recent-progress-on-the-kakeya-conjecture/.

[103] Vadhan, S. P., Pseudorandomness. *Foundations and Trends in Theoretical Computer Science* **7** (1-3) (2011), 1–336.

[104] Valiant, L., The Complexity of Computing the Permanent. *Theoretical Computer Science (Elsevier)* **8** (2) (1979), 189–201.

[105] Wigderson A., P, NP and Mathematics - A computational complexity perspective. *Proceedings of the ICM 2006* **1** (2007), EMS Publishing House, Zurich, 665–712.

[106] Wolff, T., Recent work connected with the Kakeya problem. *Prospects in Mathematics*, AMS, Princeton, NJ, 1999, 129–162