The work of Dan A. Spielman

Gil Kalai

ICM 2010, Hyderabad, India.

Dan Spielman, our new Nevanlinna prize laureate, has made ground breaking contributions in theoretical computer science and mathematical programming and his work has profound connections to the study of polytopes and convex bodies, to error correcting codes, expanders, and numerical analysis.

Many of Spielman's achievements came with a beautiful collaboration spanned over two decades with Shanghua Teng.

In the lecture we mainly describe smoothed analysis of algorithms, which is a new paradigm for the analysis of algorithms introduced by Spielman and Teng followed by their explanation to the excellent practical performance of the simplex algorithm via smoothed analysis.

We will also briefly describe Spielman's works on error-correcting codes and touch on his recent work on linear equation solvers, sparsifiers, and spectral graph theory.

I. Smoothed Analysis

"Shang-Hua Teng and I introduced smoothed analysis to provide a means of explaining the practical success of algorithms and heuristics that have poor worst-case behavior and for which average-case analysis was unconvincing."

Worst-Case Analysis

Start with a problem like: Color a map with four colours; Find the zero of a function; Solve a system of linear equations.

- Worst case analysis of an algorithm - The maximum number of operations required by the program as a function of the input length.

Extremely effective tool to study the difficulty of a problem (at least in rough terms, (based on $P \neq NP$.))

Not an effective tool to study the practical performance of a specific algorithm.

Average-Case Analysis

- Average case analysis - The expected number of operations required by the program as a function of the input length for a "random" problem.

(Not to be confused with randomized algorithm and with Levin's notion of average-case complexity.)

Not an effective tool to study the difficulty of a problem, since usually, average case analysis is very difficult.

Gives some understanding regarding practical performance of a specific algorithm.

What is Smoothed Analysis?

Smoothed analysis is a hybrid of worst-case and average-case analysis that inherits advantages from both. The smoothed complexity of an algorithm is the maximum over its inputs of the expected running time of the algorithm under slight random perturbations of that input.

Smoothed Analysis 2000-2010

- The simplex algorithm (Spielman and Teng, followed by others)
- Interior point algorithms
- Graph algorithms
- Integer programming
- k-means and clustering
- and many more...

II. Linear Programming and the Simplex Algorithm

In 1950 George Dantzig introduced the *simplex algorithm* for solving linear programming problems. Linear programming and the simplex algorithm are among the most celebrated applications of mathematics.

A linear programming problem is the problem of finding the maximum of a linear functional (called *a linear objective function*) on d variables subject to a system of n inequalities. The set of solutions to the inequalities is called the *feasible polyhedron* and the simplex algorithm consists of reaching the optimum by moving from one vertex to a neighboring vertex of the feasible polyhedron. The precise rule for this move is called *the pivot rule*.

Understanding the complexity of linear programming and of the simplex algorithm is a major problem in mathematical programming and in theoretical computer science.

Early Thoughts, and Counterexamples

The performance of the simplex algorithm is extremely good in practice. In the early days of linear programming it was believed that the common pivot rules reach the optimum in a number of steps that is polynomial or perhaps even close to linear in d and n. As we will see shortly, this belief turned out to be false.

A related conjecture by Hirsch asserts that for d-polytopes (bounded polyhedra) defined by n inequalities in d variables there is always a path of length at most $n - d$ between every two vertices. The Hirsch conjecture was recently disproved by Francisco Santos.

Klee and Minty found that one of the most common variants of the simplex algorithm is exponential in the worst case. In fact, the number of steps was quite close to the total number of vertices of the feasible polyhedron. Similar results for other pivot rules were subsequently found by several authors.

Understanding LP and the Simplex Algorithm 1950-2000: LP $\in$ P

What can explain the excellent practical performance? In 1979 Khachian proved that $LP \in P$; namely, there is a polynomial time algorithm for linear programming. This had been a major open problem since the complexity classes P and NP were described in the late sixties, and the solution led to the discovery of polynomial algorithms for many other optimization problems. Khachian's proof was based on Nemirovski and Shor's ellipsoid method, which is not practical. For a few years there was a feeling that there is a genuine tradeoff between being good in theory and being good in practice. This feeling was shattered with Karmarkar's 1984 interior point method and subsequent theoretical and practical discoveries.

Understanding LP and the Simplex Algorithm 1950-2000: Average Case Behavior

We come now to developments that are most closely related to Spielman and Teng's work. Borgwardt and Smale pioneered the study of average case complexity for linear programming. It turns out that a certain pivot rule first introduced by Gass and Saaty called the *shadow boundary rule* is most amenable to average-case study. Borgwardt was able to show polynomial average-case behavior for a certain model that exhibits rotational symmetry. In the mid-80s, three groups of researchers were able to prove *quadratic* upper bound for the simplex algorithm for very general random models that exhibit certain sign invariance.

Smoothed Analysis of the Simplex Algorithm

Consider a linear programming (LP) problem:

$\mathbf{max} < \mathbf{c}, \mathbf{x} >, \quad \mathbf{x} \in \mathbf{R}^d$

$\mathbf{subject\ to\ Ax} \leq \mathbf{b}$

Here, $\mathbf{A}$ is an n by d matrix, $\mathbf{b}$ is a column vector of length n, and $\mathbf{c}$ is a column vector of length d.

Next, add independently a Gaussian random variable with variance σ to each entry of the matrix A.

Theorem (Spielman and Teng)

For the shadow-boundary pivot rule, the average number of pivot steps required for a random Gaussian perturbation of variance σ of an arbitrary LP problem is polynomial in d, n, and σ^{-1}.

Smoothed Analysis of the Simplex Algorithm: The Proof

Spielman and Teng's proof is truly a *tour de force*. It relies on a very delicate analysis of random perturbations of convex polytopes. I will mention two ideas.

1. The number of vertices in a 2-dimensional projection of the feasible polytope is only polynomial in d, n, and σ^{-1}.

2. The angles at vertices of the feasible polytopes for the perturbed problem are not "flat". (Flatness is expressed by an important parameter called the "condition number" of a linear system.)

Further Developments

- Smoothed analysis of interior point methods (Dunagan, Spielman and Teng)
- Simplified proofs; better bounds; more general stochastic models (S&T; Vershynin; Tao and Vu;...)
- Towards a strongly polynomial time algorithm for LP (Kelner & Spielman).

III. Error Correcting Codes

Error-correcting codes are among the most celebrated applications of mathematics. Error-correcting codes are eminent in today's technology from satellite communications to computer memories.

Codes have important theoretical aspects in mathematics and CS. The classical geometric problem of densest sphere packing is closely related to the problem of finding error-correcting codes. Codes play a role in statistical testing, and in the area of "Hardness of approximations and PCP." Quantum error correcting codes are expected to be crucial in the engineering and building of quantum computers.

What are Codes?

A binary code C is simply a set of 0-1 vectors of length n. The minimal distance $d(C)$ is the minimal Hamming distance between two elements $x, y \in C$. The same definition extends when the set $\{0, 1\}$ is replaced by a larger alphabet Σ. When the minimal distance is d the code C is capable of correcting $[d/2]$ arbitrary errors. The rate of a code C of vectors of length n is defined as $R(C) = \log |C|/n$.

High Rate Codes Admitting Linear Time Encoding and Decoding

Theorem (Spielman 1995)

There is a construction of positive rate linear codes which correct a positive fraction of errors and which admit a linear time algorithm for decoding and for encoding.

From Expanders to Codes!

There are three steps in the proofs.

- Moving from graphs to codes (Gallager, 1965)
- Expander graphs lead to linear-time encodable codes (Sipser and Spielman, 1995)
- Sort of imitating for codes the construction of superconcentrators from expanders (Spielman, 1995).

Tornado Codes

Dan Spielman has made other important contributions to the theory of error-correcting codes and to connections between codes and computational complexity. Some of his subsequent work on error-correcting codes took a more practical turn.

Spielman, together with Michael G. Luby, Michael Mitzenmacher and M. Amin Shokrollahi constructed codes that approach the capacity of erasure channels. Their constructions, which are now called *tornado codes*, have various practical implications. For example, they can be useful for compensating for packet loss in Internet traffic.

IV. Linear System Solvers, Sparsifiers, Spectral Graph Theory, and Numerical Analysis

Dan Spielman recently focused his attention to one of the most fundamental problems in computing: the problem of solving a system of linear equations. Solving large-scale linear systems is central to scientific and engineering simulation, mathematical programming, and machine learning. Dan found remarkable nearly linear time algorithms for several important classes of linear systems.

This has led to considerable theoretical advances as well as to practically good algorithms.

Conclusion

The beautiful interface between theory and practice, be it in mathematical programming, error-correcting codes, the search for Ramanujan-quality sparsifiers, the analysis of algorithms, computational complexity theory, or numerical analysis, is characteristic of Dan Spielman's work.