Exact Algorithms and Complexity

Ramamohan Paturi*

Department of Computer Science and Engineering
University of California, San Diego
La Jolla, CA 92093-0404, USA

Over the past couple of decades, a series of exact exponential-time algorithms have been developed with improved run times for a number of problems including IndependentSet, k-SAT, and k-colorability using a variety of algorithmic techniques such as backtracking, dynamic programming, and inclusion-exclusion. The series of improvements are typically in the form of better exponents compared to exhaustive search. These improvements prompt several questions, chief among them is whether we can expect continued improvements in the exponent. Is there a limit beyond which one should not expect improvement? If we assume $\mathbf{NP} \neq \mathbf{P}$ or other appropriate complexity statement, what can we say about the likely exact complexities of various **NP**-complete problems?

Besides the improvement in exponents, there are two other general aspects to the algorithmic developments. Problems seem to differ considerably in terms of the improvements in the exponents. Secondly, different algorithmic paradigms seem to work best for different problems. These aspects are particularly interesting given the well-known fact that all **NP**-complete problems are equivalent as far as polynomial-time solvability is concerned. How do the best possible exponents differ for different problems? Can we explain the difference in terms of the structural properties of the problems? Are the likely complexities of various problems related? What is relative power of various algorithmic paradigms?

One approach would be to consider natural, though restricted, computational models. For example, consider the class **OPP** of one-sided error probabilistic polynomial-time algorithms. **OPP** captures a common design paradigm for randomized exact exponential-time algorithms: to repeat sufficiently many times a one-sided error probabilistic polynomial-time algorithm that is correct with an exponentially small probability so that the overall algorithm finds a witness with constant probability. This class includes Davis-Putnam-style backtracking algorithms developed in recent times to provide improved exponential-time upper bounds for a variety of **NP**-hard problems. While the original versions of some of these algorithms are couched as exponential-time algorithms, one can observe from a formalization due to Eppstein that these algorithms can be converted into probabilistic polynomial-time algorithms whose success probability is the reciprocal of the best exponential-time bound. The class is interesting not just because of ubiquity, but because such algorithms are ideal from the point

* This research is supported by NSF grant CCF-0947262 from the Division of Computing and Communication Foundations.

O. Strichman and S. Szeider (Eds.): SAT 2010, LNCS 6175, pp. 8–9, 2010.

of view of space efficiency, parallelization, and speed-up by quantum computation. What are the limitations of such algorithms for deciding **NP**-complete problems? Could the best algorithm for specific **NP**-complete problems be in this class?

On the other hand, the recent algorithms for k-colorability for $k \geq 3$ use inclusion-exclusion principle in combination with dynamic programming to achieve the bound of 2^n. This raises a natural question whether we can expect an **OPP** algorithm for k-colorability whose success probability is at least 2^{-n}. More generally, can we expect **OPP**-style optimal algorithms for k-colorability? Does there exist any **OPP** algorithm for k-colorability whose success probability is at least c^{-n} where c is independent of k? Negative answers (or evidence to that effect) for these questions would provide convincing proof (or evidence) that exponential-time inclusion-exclusion and dynamic programming paradigms are strictly more powerful than that of **OPP**. On the other hand, algorithmic results that would place k-colorability in the class **OPP** with c^{-n} success probability would be exciting.

The current state of the art in complexity theory is far from being able to resolve these questions, especially the question of best exponents, even under reasonable complexity assumptions. However, recent algorithmic and complexity results are interesting and they provide food for thought. In this talk, I will present key algorithmic results as well as our current understanding of the limitations.