Ryan O'Donnell
Assistant Professor
Computer Science Department
SCS

August 29th, 2007 3:20pm

Research Topic
Polynomial-time Algorithms

Research Problem
What level of efficiency can we achieve from known NP-hard problems and still achieve polynomial time?

Problem Statement
Given an algorithm for an NP-hard problem, identify the maximum level of efficiency we can achieve by modifying the algorithm while still performing in polynomial time.

NP-hard problem: a mathematical problem (such as the traveling salesman problem) to which there is either no known solution (other than trial and error), or there is an algorithm that provides a solution, but the solution algorithm increases in size proportional to the particular problem at a rate that is greater than a polynomial rate.

Maximum level of efficiency: given a mathematical problem asking for the "best possible" answer, the maximum level of efficiency is "how close" you can get to that answer using an algorithm which performs in polynomial time.

Modifying the algorithm: taking a given algorithm and changing the parameters or arguments in such a way that the resulting algorithm performs better in some way.

Polynomial time: An algorithm performs in polynomial time if the ratio from the size of the algorithm to the size of a specific problem is polynomial, i.e. of the form $ax + bx^2 + cx^3 + \ldots$

Problem Description
After many years of investigation, we've discovered quite a few algorithms which can be performed in polynomial time. However, have we discovered them all? There are several problems which we know are NP-hard, and so they probably cannot be; however, there is a small subset which are unknown. Additionally, there are several optimization problems which are only NP-hard in the exact version; i.e., what is the MOST efficient solution for problem X? In reality, we are ok with a solution which is maybe 85% as efficient as the theoretical most-efficient solution, because it is close; and we know that up to about 87% efficiency, determining a solution of this fidelity can actually be done in polynomial time. We also know that 94% and up requires non-polynomial time; what we don't know is the highest efficiency that can be attained in polynomial time (i.e. can 87-94% efficiency be achieved in polynomial time).

Computer Science Perspective
This is very much an algorithms problem; what are the most efficient algorithms we can use to solve problems in reasonable (polynomial) time. By answering these questions, we provide computer science with more tools to use in the creation of practical solutions.

Disciplines actively involved
Complexity theory; Mathematics; Algorithms.

Actively Involved Discipline: a discipline in which further research can result as a result of the successful completion of this research.

Description of Disciplines Involved
This problem clearly involves Mathematics and complexity theory as the foundation for this research. Algorithms are the subject of the research (finding algorithmic approximations) .

References
Presenter web page:
http://www.cs.cmu.edu/~odonnell/
Optimal inapproximability results for MAX-CUT and other two-variable CSPs?:
http://www.cs.cmu.edu/~odonnell/papers/maxcut.pdf

By mnovakou
Updated mnovakou