

Executive Summary for the Dagstuhl Seminar 09511

Parameterized Complexity and Approximation Algorithms

Erik Demaine	MohammadTaghi HajiAghayi	Daniel Marx
M.I.T.	AT&T Research	Tel Aviv University
edemaine@mit.edu	Hajiagha@research.att.com	dmarx@cs.bme.hu

Many of the computational problems that arise in practice are optimization problems: the task is to find a solution where the cost, quality, size, profit, or some other measure is as large or small as possible. The NP-hardness of an optimization problem implies that, unless $P = NP$, there is no polynomial-time algorithm that finds the exact value of the optimum. Various approaches have been proposed in the literature to cope with NP-hard problems. When designing approximation algorithms, we relax the requirement that the algorithm produces an optimum solution, and our aim is to devise a polynomial-time algorithm such that the solution it produces is not necessarily optimal, but there is some worst-case bound on the solution quality.

In parameterized complexity the running time is analyzed in finer detail: instead of expressing it as a function of the input size, one or more parameters of the input instance are defined, and we investigate the effect of these parameters on the running time. The goal is to design algorithms that work efficiently if the parameters of the input instance are small (even if the size of the input is large). More precisely, we say that a problem is fixed-parameter tractable (FPT) with parameter k if the problem can be solved in time $f(k)n^c$ for some function f and constant c . That is, our goal is to design algorithms that are polynomial in n and exponential only in the parameter k . The motivation behind this definition is that in practice we do not have to be able to solve the problem for any possible input: we might be able to define some parameter k that is typically small for the instances we encounter.

Until very recently, approximation algorithms and parameterized complexity have been considered to be two different approaches that have very little to do with each other. Indeed, the methodology of the two fields are very different: the design of approximation algorithms has its own distinctive set of tools such as linear programming, greedy algorithms, probabilistic methods, while parameterized complexity uses a different set of techniques: kernelization, bounded search trees, and extremal combinatorics. However, in the past few years, several connections between the two fields were identified that are worth investigating.

During the 4 days of the conference, 23 talks were given by the participants. Five of these talks were 60-minute surveys on various topics: Dániel Marx talked about several existing connections between approximation algorithms and fixed-parameter algorithms; Gregory Gutin talked about the rapidly growing area of parameterization above guaranteed values; Erik Demaine talked about the recent

area of bidimensionality relevant to both approximation and fixed-parameter algorithms; Guy Kortsarz talked about relevant problems in wireless network design; and Daniel Lokshantov talked about lower bounds on kernelization. As an additional highlight of the seminar, Holger Dell presented in detail his exciting new result about sparsification and its applications to kernel lower bounds.

It is becoming increasingly clear that kernelization—both upper bounds and lower bounds—are becoming a central focus of fixed-parameter algorithms. The talks of Lokshantov and Dell have shown that kernelization lower bounds are a rich topic with much deep work to be done, and the talks of Demaine and Bodlaender have shown that “metakernelization” results are possible for wide ranges of problems. It is expected that in the next few years there will be substantial further progress on the topic of kernelization.

The seminar successfully brought together both experts and newcomers from the two fields of approximation algorithms and fixed-parameter algorithms, with many interesting interactions. The talks left plenty of time for discussion in the afternoon. An open problem session was held on Monday, and problems raised there were discussed by different groups throughout the seminar.