Approximation Algorithms and Hardness of Approximation

EECS 598 - Special Topics

Instructor: Euiwoong Lee

Most interesting optimization problems are NP-hard, so unless P = NP, there are no polynomial time algorithms to find optimal solutions to such problems. Therefore, it is natural and important to study approximation algorithms that compute approximately optimal solutions with provable guarantees. Computational complexity can be also extended to approximate optimization, and for many optimization problems, it is possible to prove that even computing an approximately optimal solution remains NP-hard.

Approximation algorithms have been actively studied in both algorithms and complexity theory, culminating in optimal approximation algorithms for some fundamental problems; they achieve some approximation guarantees and no polynomial time algorithm can do better under some complexity conjectures. The theory of approximation algorithms also leads to beautiful connections between algorithms, complexity, and some areas of mathematics. This course will provide an overview of these connections, stressing techniques and tools required to prove both algorithms and complexity results.

A tentative list of topics: linear programming (LP) and semidefinite programming (SDP) relaxations, primal-dual method, randomized rounding, metric embedding, probabilistically checkable proofs, discrete Fourier analysis, and unique games conjecture.

Prerequisites: Graduate or senior standing, or permission of the instructor.

Recommended: EECS 203, EECS 281, EECS 376, Math 217, or equivalents.