CO 450/650, Fall 2017
Combinatorial Optimization

Class time and location: TTh 11:30 — 12:50 in RCH 110

Instructor: Jochen Koenemann

Overview:

Combinatorial Optimization is a field that combines techniques from combinatorics, linear programming, and the theory of algorithms to solve discrete optimization problems. In a typical instance of such a problem, we are given a finite discrete space of solutions, and a rational cost function that assigns a cost to each one of them. The question that interests us is: can we find a minimum-cost solution efficiently (i.e., in time polynomial in the size of the input)?

In order to design an efficient algorithm, we need to understand the structure of the given solution set. For example, consider the problem of finding a shortest path between two nodes u and v in a given weighted graph. The number of such paths may well be exponential in the size of the input, yet efficient algorithms to find a shortest one among them exist; these methods crucially exploit the well-behaved structure of the solution set. Slight variants of the above problem are less well-behaved: the problem of finding a shortest u,v-path via a third node z in some input graph is provably hard to solve!

In this class, we study a large class of discrete optimization problems that are well-behaved, i.e., for which efficient exact algorithms exist. Topics covered in this class include:

- Spanning trees & algorithms
- Maximum flows & minimum cuts
- Matchings
- Matroids & Greedy algorithms

Expectations:

Apart from general mathematical maturity, students are expected to be familiar with the fundamentals of linear optimization and graph theory. Chapter 1-3 (pages 1-70) of [*Combinatorial Optimimization*, Korte & Vygen] (see https://link.springer.com/content/pdf/10.1007/978-3-540-71844-4.pdf) serves as an adequate refresher.