Combinatorial Algorithms

Rationale

Combinatorial algorithms is a core part of algorithms, which is a core part of computer science, Many of the optimization problems that are most fundamental to computer science and have had the greatest "broader impact" outside of computer science and indeed within the wider world – shortest paths for travel, network flow for business and transportation, maximum matching for resource allocation, linear programming for myriad operations research problems – are among the topics covered in this course.

Course Description

This is a course on combinatorial algorithms covering topics (far) beyond the scope of the first-year algorithms class. More precisely, this is an advanced course in algorithms for optimization problems concerning discrete objects, principally graphs. In such problems, we search a finite but typically exponentially large set of valid solutions—e.g., all matchings in a graph for maximizing or minimizing some objective function. Nonetheless, most of the problems we study in this course are optimally solvable in polynomial time. The fundamental topics here are matchings, flows and cuts, shortest paths and spanning trees, and matroids. An overarching theme is that many such problems have traditionally been studied both a) by computer scientists, using discrete, combinatorial algorithms (greedy, dynamic programming, etc.), and b) in the operations research optimization community, where they are treated as continuous optimization problems (solved by linear programming, etc.). We will often compare the two approaches, and we will find that it can be fruitful to combine them. In particular, we will repeatedly use linear programming throughout the course.

Topic List

Topics can include but are not limited to:

- shortest paths and spanning trees

  - single-source (Dijkstra, Bellman-Ford)
  - all-pairs (Floyd-Warshall)
  - spanning trees (Prim, Kruskal)
  - arborescences (Edmonds)
  - NP-Complete extensions (CDS, Steiner, etc.)

- linear programming
  - convex hulls and polyhedra
  - simplex,
  - LP-rounding for approximation
  - duality
  - packing and covering LPs
  - primal-dual method
  - totally unimodular matrices

- convex optimization
  - convex programming
  - Lagrangian duality
  - semi-definite programming

- matchings
  - unweighted bipartite (augmenting paths)
  - weighted bipartite (Hungarian)
  - unweighted non-bipartite (Edmonds)
  - weighted non-bipartite (more Edmonds)
  - NP-Complete extensions (GAP, 3DM, etc.)

- network flows
  - disjoint paths (Menger)
  - max flow / min cut

- augmenting path algorithms
- push-relabel algorithms
- min-cost max flow
- multi-commodity flow and multicut

Learning goals

Learn some of the canonical algorithms (and algorithm schemata) for solving fundamental matching, flow, and path problems; become able to apply and extend these techniques to new problem variations; come to see how many of these problems are mutually reducible to one another; gain an appreciation of the conceptual foundations of duality and matroid theory, and of polyhedral combinatorics as mathematical technology.

Assessment

The theoretical concepts and proofs will be assessed by bi-weekly homeworks. The algorithms will be assessed by a final project.