College of Computer Science — Fall 1999
Northeastern University — Handout 1
COM 3391: Advanced Algorithms — 29 September 1999

Topics in Approximation Algorithms

Course Information

Instructor. Rajmohan Rajaraman; Email: (rraj@ccs.neu.edu); Phone: 617–373–2075; Office: 113 Cullinane Hall.

Class Homepage. http://www.ccs.neu.edu/home/rraj/Courses/3391/F99/.

Class Meeting Time. W: 5:15–8:15 PM.

Office Hours. M: 5:00–6:00 PM and W: 2:00–3:00 PM.

Prerequisites. The only prerequisite for this course is COM3390 (Analysis of Algorithms). The student must be familiar with greedy algorithms, dynamic programming, graph algorithms for minimum spanning trees, shortest paths. Familiarity with the theory of NP-completeness will be an asset, although not required. A review of NP-completeness will be done in the first class meeting. It is recommended that the student review the material on NP-completeness given in Chapter 36 of *Introdution to Algorithms*, Cormen, Leiserson, and Rivest, MIT Press, 1990.

Course Description. This course will cover some basic topics in the design and analysis of approximation algorithms. The study of approximation algorithms has developed from the seeming intractability of a number of widely-applicable *NP-hard* optimization problems. These optimization problems are unlikely to admit efficient (polynomial-time computable) optimal solutions. Consequently, a number of techniques have been designed to provide approximate (near-optimal) solutions that can be obtained efficiently. Of course, we would like to sacrifice as little optimality as possible, while gaining as much as possible in efficiency. Trading off optimality in favor of efficiency is the paradigm of approximation algorithms.

Approximation algorithms have recently gained significant attention with the development a number of new techniques this decade. These techniques range from simple ones like greedy algorithms, randomization, and dynamic programming, to more sophisticated techniques based on linear programming. Moreover, these techniques have been applied to a variety of different problems in diverse areas including databases, operating systems, and networking.

The goal of this course is to present and analyze the various techniques in approximation algorithms in the context of different applications. Topics covered will be selected from the following:

- Brief review of complexity theory: the classes P and NP, the notion of NP-hardness, polynomial-time reductions.
- The framework for analyzing approximation algorithms: approximation ratio, different notions of approximations.

- Techniques: greedy algorithms, dynamic programming, randomization, design and analysis techniques based on linear programming.
- Classic optimization problems such as the set cover problem and the traveling salesperson problem.
- Network design problems such as the network survivability problem and minimum Steiner Tree problem (that arises in multicast routing).
- Facility location problems that arise in a wide range of applications in operations research and computer science; for example, the location of base stations in wireless networks, the location of fire stations in a city.
- Online algorithms for load balancing, scheduling, and virtual circuit routing.
- A brief introduction to the hardness of approximations.

Textbook: The material covered in the course will be taken from:

- *Introdution to Algorithms*, Cormen, Leiserson, and Rivest, MIT Press, 1990.
- *Approximation Algorithms for NP-Hard Problems*, edited by Dorit Hochbaum, PWS Publishing Company, 1997.
- Lecture notes on approximation algorithms by Michel Goemans, Rajeev Motwani, and Vijay Vazirani.
- Recent research papers that will be handed out in class.

Grading. The course grade will be based on two problem sets (each worth 15%), a midterm (30%), and a project (40%). Furthermore, each student is required to scribe one lecture. We will put together the lecture notes as a manuscript at the end of the course.

Projects. Students will be required to do a course project which could be any one of the following types:

- An implementation project evaluating an approximation technique or different approximation algorithms for an optimization problem.
- Presentation of a survey article or a research paper.
- Work on a research problem.