Math 466/566 — Network Optimization (Fall 2008)

Time	Tu-Thu 12:00–1:15 p.m.
Location	Webster 11
Instructor	Bala Krishnamoorthy
Office	Neill 325
Office Hours	Tue 2:00–4:00 p.m, Wed 1:00–3:00 p.m.
Email	kbala@wsu.edu
Course web page	http://www.wsu.edu/∼kbala/Math566.html
Book	Ahuja, Magnanti, and Orlin — Network Flows: Theory, Algorithms, and Applications. Prentice-Hall. ISBN: 013617549X
References	D. Bertsekas — Network Optimization: Continuous and Discrete models Athena Scientific. ISBN: 1886529027
	Bazaraa, Jarvis, and Sherali — Linear Programming and Network Flows, second edition John wiley and Sons ISBN: 0471636819
	Cormen, Leiserson, Rivest, and Stein — Introduction to Algorithms, second edition MIT Press. ISBN: 0262032937

Description of the Course

Network flow problems is an important class of optimization problems, with applications to several areas including chemistry, computer networking, engineering, public policy, scheduling, telecommunications, transportation, and many others. This course will provide an integrated view of the theory, algorithms, and the applications of key network optimization problems including the shortest path problem, the maximum flow problem, the minimum cost flow problem, the minimum spanning tree problem, and the multi-commodity flow problem. Most of the arguments will be presented from first principles, and we will adopt a network or graphical view point. Limited use of a linear programming approach will be involved, for which the necessary background materials will be reviewed. The only pre-requirement is **mathematical intuition**! Emphasis will be on powerful algorithm strategies, rigorous analysis of the algorithms, and data structures for their implementation. Apart from problems involving proofs (in homework and exams), the student will be produce simple implementations of some of the algorithms (using MATLAB). Animations of algorithms will also be presented in order to help the student in gaining a sound intuition about them. Depending on student interest, we will discuss topics from combinatorial optimization as well.

Organization and Grading

There will be around 10 homework assignments. Each assignment will be handed out at least one week before the day on which it will be due (dates are given in the tentative schedule, but these dates might be subject to change). Some of the problems will involve writing simple MATLAB codes to produce a running version of an algorithm discussed in class. Students should email the programs to me. Discussion of homework problems with others is allowed, but each person should hand in his or her own written solutions and codes.

There will be one mid-term and a final exam. The mid-term will be an in-class, closed-book exam. The final exam will be a take-home, open-book exam. It will be cumulative, but will concentrate more on the topics covered after the mid-term. Apart from the homework assignments and the two exams, there will be a course project. Students could work on the project in teams of two (or individually). The total score for the course will be calculated using the following weights:

- homework - 50 %
- project - 15 %
- mid-term - 15 %
- final exam - 20 %.

The least homework grade **from among those turned in** will be dropped.

In each homework and in the exams, a few problems will be marked as "advanced". Students registered for Math 466 will not be responsible for solving these problems. They are strongly encouraged to attempt the advanced problems, though. Points gained on these problems by Math 466 students will be counted as extra credit. There will be two separate curves for determining the final grades – one each for Math 466 and for Math 566.

Software

Coding assignments should preferably be done using MATLAB. Students registered for the class (466 or 566) can use MATLAB available at http://my.math.wsu.edu. In order to login, one should use his or her WSU student id and **4mymath** as the password. Using the version available in a PC or in the machines in one of the labs in Neill Hall would be more efficient though.

We will discuss several algorithms in the course. Animating the algorithms will make them more clear, intuitive, and easy to analyze. A program called GIDEN will be used for producing the animations discussed in class. This program is consistent with the notation and pseudo-codes used in the text. Students can access GIDEN for free at http://users.iems.northwestern.edu/∼giden/. The use of GIDEN will not be required for doing your homework, project, or in exams.