Lecture 8 – Integer Solutions in Network Flow Problems

Module 5 – Special Cases of Linear Programs
CS/ISyE/ECE 524

A note on integer solutions

- Some minimum-cost problems require integer solutions
  - Remember the binary variables in assignment problems and shortest/longest path problems
- So far, we've just been ignoring that requirement and "hoping for the best"
  - And so far, it's worked out!

Question: Is there a way of *guaranteeing* we get an integer solution? Or have we just been super lucky?

Yes! We can *guarantee* integer solutions when the incidence matrix A is **totally unimodular.**

A matrix A is **totally unimodular (TU)** if *every* square submatrix of A has a determinant of $1, -1$, or 0.

Note that this definition includes 1×1 submatrices, so every entry of A must be $1, 0$, or -1

Some Totally Awesome Facts about Totally Unimodular Matrices

Theorem
If A is TU and b is an integer vector, the vertices of $\{x \mid Ax \leq b\}$ have integer coordinates. The optimal solution of an LP is always at a vertex. Therefore, under these conditions, we can solve an LP and get an integer solution!

Theorem
Every incidence matrix is TU.

Examples:
$\begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$ is TU. $\begin{bmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \end{bmatrix}$ is not.
(determinant of 1st and 3rd column is 2).

- So what? (A lot!)
  - If a MCNF problem has integer demands, supplies, and edge capacities, there exists a min cost ***integer*** solution
  - Every assignment problem is an LP
  - Every shortest path problem is an LP
- When we talk about integer programs, we'll see why this matters so much

Special Cases of Linear Programs Module Learning Outcomes

Now you should be able to…

- Recognize the multi-period planning problem structure
  - Build a linear programming model of a multi-period planning problem
  - Give an example of a multi-period planning problem
- Reformulate a model with piecewise linear objective into a standard form linear programming model
- Recognize the "minimax" (maximin) problem structure
  - Build a linear programming model of a minimax problem
  - Give an example of a minimax problem

Special Cases of Linear Programs Module Learning Outcomes

Now you should be able to…

- Give basic definitions of combinatorial optimization problems and be able to describe how they relate to network flow problems
  - Graph
  - Node (vertex)
  - Arc (edge)
  - Flow
- Recognize when a problem falls into the min-cost network flow structure and build a linear programming model of the problem
- Give examples of different types of min-cost network flow problem (e.g., shortest path, assignment, max flow, matching)
- Explain the concept of total unimodularity
  - Understand the relationship between TU matrices and MCNF problems