IEORE4004: Optimization Models and Methods 11/21/2019

Assignment 5

Deadline: 12/03/2018, h: 11:40am *Instructor: Christian Kroer*

- This assignment sheet has 4 exercises. You must submit your solution via Canvas before the deadline. The time constraint is strict. Late submissions will not be accepted. Each time you are requested to write an LP and solve it with a solver, report the formulation and the screenshot of the output of the solver.
- You are allowed to discuss the assignment with others but the write-up must be individual work. Please mention in your write-up all the people you have discussed the solution with.

Problem 1:

a. The data set "t2.dat" describes a minimum-cost flow problem that can be read by the supplied script "singlecomm.py". You can use this script to create an LP that solves the minimum-cost flow problem (or you can just type it yourself). Check using Gurobi that this problem is infeasible.

b. Consider the following optimization problem that adds capacities to the links of the network, at minimum cost, so as to make the problem feasible. If you add $\Delta_{ij} \geq 0$ units of capacity to arc (i, j), then you pay $c(i, j)\Delta(i, j)$, where $c(i, j)$ is the original cost. We want to make the problem feasible by spending as little as possible [Note: you do not need to consider the cost of the flow for this part]. Solve this problem using Gurobi (or your favourite solver).

c. Add the optimal Δ^* computed at the previous step and solve the min-cost flow problem using Gurobi (or your favourite solver).

d. Decompose the flow you obtained at the previous step as seen in class.

Problem 2: Suppose we have a maximum flow problem (from node s to node t), where, in addition, we are told that every node different from s and t has a restriction: the flow through that node cannot exceed ten units. Show how to formulate the problem as a max flow problem in a generic network.

Problem 3: Five baseball teams compete with each other over a season of 70 games. Hence, each team plays 7 times against each other team. Suppose that each team has played with each other team 5 times, and overall results are reported below (recall that in baseball there is no tie).

Team	# games won
A	10
B	10
C	11
D	7
E	12

We say that a team X *wins the tournament* if there is no team that wins strictly more games than X.

a. Formulate as a maximum flow problem the question of deciding if team D can still win the tournament. Solve it (either by hand or using a software).

b. Formulate as a maximum flow problem the question of deciding if team D can be *the only team* that wins the tournament. Solve it (either by hand or using a software).

Problem 4:

a. You are given a 4×7 board and a robot initially placed at the upper left corner. On each step, the robot can either move one cell to the right (R), or one cell down (D). We are interested in the number of ways there are for the robot to move to the destination, which is the lower right corner.

						dest

For example, two possible routes the robot can take are:

$$Route1 : R, R, R, R, R, R, D, D, D$$
$$Route2 : R, D, R, D, R, D, R, R, R$$

Use Dynamic Programming to calculate how many distinct ways there are for the robot to move to the destination. (Hint: You can use $f_i(j)$ to represent the number of ways there are if the board is of size $i \times j$.)

** Remark: from combinatorics' perspective, this number is equivalent to the number of ways we can arrange* 6 *R's and* 3 *D's in a sequence, which is* $\frac{9!}{3!6!}$.

b. Now assume that coins are placed in some cells (shown in the table below). The goal of the robot is to collect as many coins as possible and bring them to the destination. As in the previous part, the robot can only move one cell right or one cell down each time. Use Dynamic Programming to figure out which route the robot should take to maximize the amount of coins collected.

					$\$$	
		$\$$	$\$$			$\$$
	$\$$			$\$$		
		$\$$			$\$$	dest