CS 170 HW 9

Due on 2018-10-28, at 9:59 pm

1 (★) Study Group

List the names and SIDs of the members in your study group.

2 (★★★) A cohort of spies

A cohort of k spies resident in a certain country needs escape routes in case of emergency. They will be travelling using the railway system which we can think of as a directed graph $G = (V, E)$ with V being the cities. Each spy i has a starting point $s_i \in V$ and needs to reach the consulate of a friendly nation; these consulates are in a known set of cities $T \subseteq V$. In order to move undeteced, the spies agree that at most c of them should ever pass through any one city. Our goal is to find a set of paths for each of the spies (or detect that the requirements cannot be met).

Model this problem as a flow network. Specify the vertices, edges and capacities, and show that a maximum flow in your network can be transformed into an optimal solution for the original problem. You do not need to explain how to solve the max-flow instance itself.

Solution: We can think of each spy i as a unit of flow that we want to move from s_i to any vertex $\in T$. To do so, we can model the graph as a flow network by setting the capacity of each edge to ∞, adding a new vertex t and adding an edge (t', t) for each $t' \in T$. We can add a source s and edges of capacity 1 from s to s_i. By doing so, we restrict the maximum flow to be k.

Lastly, we need to ensure that no more than c spies are in a city. We want to add vertex capacities of c to each vertex. To implement it, we do the following: for each vertex v that we want add constraint to, we create 2 vertices v_{in} and v_{out}. v_{in} has all the incoming edges of v and v_{out} has all the outcoming edges. We also put a directed edge from v_{in} to v_{out} with edge capacity constraint c.

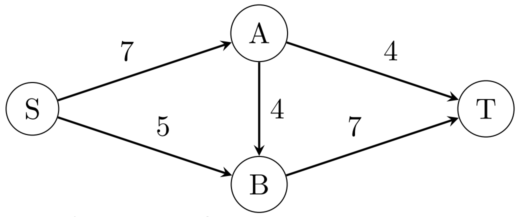
– we can, as we saw in the previous problem set, adapt the graph to express this solely with edge capacities.

If the max flow is indeed k, then as every capacity is an integer, the Ford-Fulkerson algorithm for computing max flow will output an integral flow (one with all flows being integers). Therefore, we can incrementally starting at each spy i follow a path in the flow from s_i to any vertex in T. We iterate through all of spies to reach a solution.

3 (★★★) Max Flow, Min Cut, and Duality

In this exercise, we will demonstrate that LP duality can be used to show the max-flow min-cut theorem.

Consider this instance of max flow:



Let f_1 be the flow pushed on the path $\{S, A, T\}$, f_2 be the flow pushed on the path $\{S, A, B, T\}$, and f_3 be the flow pushed on the path $\{S, B, T\}$. The following is an LP for max flow in terms of the variables f_1, f_2, f_3:

$$\begin{aligned} \max \quad & f_1 + f_2 + f_3 & \\ & f_1 + f_2 \leq 7 & \text{(Constraint for } (S, A)) \\ & f_3 \leq 5 & \text{(Constraint for } (S, B)) \\ & f_1 \leq 4 & \text{(Constraint for } (A, T)) \\ & f_2 \leq 4 & \text{(Constraint for } (A, B)) \\ & f_2 + f_3 \leq 7 & \text{(Constraint for } (B, T)) \\ & f_1, f_2, f_3 \geq 0 & \end{aligned}$$

The objective is to maximize the flow being pushed, with the constraint that for every edge, we can't push more flow through that edge than its capacity allows.

(a) Find the dual of this linear program, where the variables in the dual are x_e for every edge e in the graph.

(b) Consider any cut in the graph. Show that setting $x_e = 1$ for every edge crossing this cut and $x_e = 0$ for every edge not crossing this cut gives a feasible solution to the dual program.

(c) Based on your answer to the previous part, what problem is being modelled by the dual program? By LP duality, what can you argue about this problem and the max flow problem?

Solution:

(a) The dual is:

$$\begin{aligned} \min \quad & 7x_{SA} + 5x_{SB} + 4x_{AT} + 4x_{AB} + 7x_{BT} \\ & x_{SA} + x_{AT} \geq 1 \text{ - Constraint for } f_1 \\ & x_{SA} + x_{AB} + x_{BT} \geq 1 \text{ - Constraint for } f_2 \\ & x_{SB} + x_{BT} \geq 1 \text{ - Constraint for } f_3 \\ & x_e \geq 0 \quad \forall e \in E \end{aligned}$$

(b) Notice that each constraint contains all variables x_e for every edge e in the corresponding path. For any $s-t$ cut, every $s-t$ path contains an edge crossing this cut. So for any cut, the suggested solution will set at least one x_e to 1 on each path, giving that each constraint is satisfied.

(c) The dual LP is an LP for the min-cut problem. By the previous answer, we know the constraints describe solutions corresponding to cuts. The objective then just says to find the cut of the smallest size. By LP duality, the dual and primal optima are equal, i.e. the max flow and min cut values are equal.

4 (★★) Zero-Sum Battle

Two Pokemon trainers are about to engage in battle! Each trainer has 3 Pokemon, each of a single, unique type. They each must choose which Pokemon to send out first. Of course each trainer's advantage in battle depends not only on their own Pokemon, but on which Pokemon their opponent sends out.

The table below indicates the competitive advantage (payoff) Trainer A would gain (and Trainer B would lose). For example, if Trainer B chooses the fire Pokemon and Trainer A chooses the rock Pokemon, Trainer A would have payoff 2.

		Trainer B: ice	grass	fire
Trainer A:	dragon	-10	3	3
	steel	4	-1	-3
	rock	6	-9	2

Feel free to use an online LP solver to solve your LPs in this problem.
Here is an example of a Python LP Solver and its Tutorial.

1. Write an LP to find the optimal strategy for Trainer A. What is the optimal strategy and expected payoff?

2. Now do the same for Trainer B. What is the optimal strategy and expected payoff?

Solution:

1. d = probability that A picks the dragon type
   s = probability that A picks the steel type
   r = probability that A picks the rock type

$$\begin{aligned} \max \quad & z \\ -10d + 4s + 6r &\geq z && \text{(B chooses ice)} \\ 3d - s - 9r &\geq z && \text{(B chooses grass)} \\ 3d - 3s + 2r &\geq z && \text{(B chooses fire)} \\ d + s + r &= 1 \\ d, s, r &\geq 0 \end{aligned}$$

   The optimal strategy is $d = 0.3346,\ s = 0.5630,\ r = 0.1024$ for an optimal payoff of -0.48.

2. i = probability that B picks the ice type
   g = probability that B picks the grass type
   f = probability that B picks the fire type

$$\begin{aligned} &\min \quad z \\ -10i + 3g + 3f &\leq z && \text{(A chooses dragon)} \\ 4i - g - 3f &\leq z && \text{(A chooses steel)} \\ 6i - 9g + 2f &\leq z && \text{(A chooses rock)} \\ i + g + f &= 1 \\ i, g, f &\geq 0 \end{aligned}$$

B's optimal strategy is $i = 0.2677,\ g = 0.3228,\ f = 0.4094$. The value for this is -0.48, which is the payoff for A. The payoff for B is 0.48, since the game is zero-sum.

(Note for grading: Equivalent LPs are of course fine. It is fine for part (b) to maximize B's payoff instead of minimizing A's. For the strategies, fractions or decimals close to the solutions are fine, as long as the LP is correct.)

5 (★★★★) Minimum Spanning Trees

Consider the minimum spanning tree problem, where we are given an undirected graph G with edge weights $w_{u,v}$ for every pair of vertices u, v.

An *integer* linear program that solves the minimum spanning tree problem is as follows:

$$\begin{aligned} \text{Minimize} \quad & \sum_{(u,v)\in E} w_{u,v} x_{u,v} \\ \text{subject to} \quad & \sum_{\{u,v\}\in E : u \in S, v \in V \setminus S} x_{u,v} \geq 1 \quad \text{for all } S \subseteq V \text{ with } 0 < |S| < |V| \\ & \sum_{\{u,v\}\in E} x_{u,v} \leq |V| - 1 \\ & x_{u,v} \in \{0, 1\}, \quad \forall (u,v) \in E \end{aligned}$$

(a) Show how to obtain a minimum spanning tree T of G from an optimum solution of the ILP, and prove that T is indeed an MST. Why do we need the constraint $x_{u,v} \in \{0, 1\}$?

(b) How many constraints does the program have?

(c) Suppose that we *replaced* the binary constraint on each of the decision variables $x_{u,v}$ with the pair of constraints:

$$0 \leq x_{u,v} \leq 1, \quad \forall (u,v) \in E$$

How does this affect the optimum value of the program? Give an example of a graph where the optimum value of the relaxed linear program differs from the optimum value of the integer linear program.

Solution:

(a) $T = \{(u, v) \in E : x_{u,v} = 1\}$. The first constraint ensures that T is connected (there is at least one edge crossing every cut). The second constraint ensures that T is a tree. Moreover, every spanning tree T is a feasible solution of the ILP. The objective is the weight of T, and so the optimum is the MST. We need $x_{u,v} \in \{0, 1\}$ because it's not clear what you'd do with a fractional edge.

(b) There are $2^{|V|} + |E| - 1 = \Theta(2^{|V|})$ constraints.

(c) $v_{LP} \leq v_{ILP}$. The new linear program solution's objective value v_{LP} is at most integer linear program's objective value v_{ILP}, because every feasible solution of the ILP is a feasible solution of the LP.

One example is a cycle with 3 nodes, $w_{u,v} = 1, \quad \forall u, v \in E$. The optimal ILP formulation picks any two of the edges for a total objective cost of 2. The optimal LP formulation picks $x_{u,v} = \frac{1}{2}$ for all edges, for a total objective cost of $\frac{3}{2}$.