CS 458, Data Structures and Algorithms II
Homework 3

May 21, 2020

Due on 11:00am May 30th, 2020. Collaboration is not allowed

Problem 1 (30 points)

In class, Linear Programming was formulated in the following way:

Inputs:

- n decision variables $x_1, \dots, x_n$.
- m linear inequality constraints. Constraint j is given by $a_{1j}x_1 + \dots + a_{nj}x_n \leq b_j$.
- Linear objective function $c_1x_1 + \dots + c_nx_n$.

Goal: Find values $x_1^*, \dots, x_n^*$ for decision variables such that

- Constraints are satisfied.
- $x_1^*, \dots, x_n^* \geq 0$.
- $c_1x_1^* + \dots + c_nx_n^*$ is maximized.

This statement of the input, with a maximization objective, "less-than" constraints, and nonnegative variables, is often referred to as "standard form" for linear programs. In this problem, you'll show that linear programming can handle a few other things which on face may not even seem "linear" at all. In each part, you'll be given a mathematical program, and your job is to write an equivalent linear program in canonical form. You don't need to offer any proofs, but you should briefly explain why the two programs are equivalent, and how to go translate between solutions to the two programs.

[Hint for all three parts: define extra variables.]

1. *Unconstrained Variables*

$$\begin{array}{lll} \text{maximize} & c_1x_1 + \dots + c_nx_n & \\ \text{subject to} & a_{11}x_1 + \dots + a_{1n}x_n & \leq b_1 \\ & a_{21}x_1 + \dots + a_{2n}x_n & \leq b_2 \\ & \dots & \\ & a_{n1}x_1 + \dots + a_{mn}x_n & \leq b_m \end{array}$$

   [Note the absence of nonnegativity constraints.]

2. *Maximin Objective:*

$$\begin{array}{lll} \text{maximize} & \min(x_1, \ldots, x_n) & \\ \text{subject to} & a_{11}x_1 + \ldots + a_{1n}x_n & \leq b_1 \\ & a_{21}x_1 + \ldots + a_{2n}x_n & \leq b_2 \\ & \cdots & \\ & a_{n1}x_1 + \ldots + a_{mn}x_n & \leq b_m \\ & x_1, \ldots, x_n \geq 0 & \end{array}$$

3. *Absolute Value:*

$$\begin{array}{lll} \text{minimize} & |c_1x_1 + \ldots + c_nx_n| & \\ \text{subject to} & a_{11}x_1 + \ldots + a_{1n}x_n & \leq b_1 \\ & a_{21}x_1 + \ldots + a_{2n}x_n & \leq b_2 \\ & \cdots & \\ & a_{n1}x_1 + \ldots + a_{mn}x_n & \leq b_m \\ & x_1, \ldots, x_n \geq 0 & \end{array}$$

[Note the minimization objective.]

Problem 2 (40 points)

In this part, you are given several computational problems. Your task is to formulate each one as an integer linear program. You don't need to prove that the formulation is correct this time, but you should give an interpretation of the objective function and each constraint. Your formulations do not need to be in canonical form.

a . **(10 points)** *Max 3-SAT:* Given a boolean formula $\Phi(x_1, \ldots, x_n)$ in 3-conjunctive normal form, find a truth assignment for $x_1, \ldots, x_n$ which maximizes the number of clauses satisfied. For clause C_j, let $y_j \in \{0, 1\}$ be a variable indicating whether or not C_j is satisfied, and let $x_i \in \{0, 1\}$ denote the truth value for the boolean variable x_i. Using y_j and x_i as your variables, formulate this problem as an integer program.

b . **(10 points)** *Fair k-Suppliers.* Given a set of customers D, a set of suppliers F, distances d_{ij} between each client $j \in D$ and supplier $i \in F$, and a positive integer k, the goal is to choose a subset of k suppliers $S \subseteq F$ so as to minimize the total distance the customers have to travel to their closest selected supplier, i.e. $\sum_{j \in D} d(j, S)$. For each supplier $i \in F$, let $y_i \in \{0, 1\}$ indicate whether we choose supplier i. For each client $j \in D$ and each supplier $i \in F$, let $x_{i,j}$ indicate whether client j is "assigned" j to supplier i (i.e. if i is j's closest supplier in S). Using these y_i and $x_{i,j}$ as your variables, formulate this problem as an integer program.

c . **(10 points)** *Weighted Independent Set.* Given an undirected graph $G = (V, E)$ with vertex weights $w_v \geq 0$, the goal is to find an independent set of maximum weigh (independent set is a set of vertices such that there is no edge between any pair of vertices in the set). For each vertex v, let $x_v \in \{0, 1\}$ indicate whether v is in the independent set. Using these x_v as your variables, formulate the maximum-weight independent set problem as an integer program.

d . **(10 points)** *Load Balancing.* Given set J of n jobs and a set M of m machines, where job j requires p_j units of consecutive processing time and can only be assigned to a machine from the set $M_j \subseteq M$, the goal is to assign each job j to some machine in $i \in M_j$ such that we minimize the maximum load on any machine (i.e. the *makespan*). (If $J_i \subseteq J$ denotes the jobs assigned to a machine $i \in M$ in a feasible assignment, and using $L_i = \sum_{j \in J_i} p_j$ to denote its resulting load, we seek to minimize $\max_i L_i$). Let $x_{i,j} \in \{0, 1\}$ indicate whether job j is assigned to machine $i \in M_j$. Also let L denote the maximum load. Using these as your variables, formulate the load balancing problem as an integer program.

Problem 3 (35 points)

In class we show how to model maximum flow using variable $f_{u,v}$ for every $u, v \in V$. Additionally we showed how to model the minimum $s - t$ cut using variables $x_{u,v}$ and y_v for every vertex in $u, v \in V$

In this problem you will give an alternative linear program using paths. In all of your answers you have to provide a justification why the linear program correctly describes the problem you are given.

a. (**10 points**) Let $\mathcal{P}$ denote the set of s-t paths in G. For each path $P \in \mathcal{P}$, let x_P denote the flow on P. Using these x_P as your variables, formulate the maximum flow problem as a linear program.

b. (**10 points**) For each edge $(u, v) \in E$, let $y_{uv} \in \{0, 1\}$ indicate whether edge (u, v) crosses the cut (that is, $y_{uv} = 1$ if and only if $u \in S, v \in T$.) Using these y_{uv} as your variables, formulate the minimum cut problem as an (integer) linear program. You are not allowed to use any other variables.

c . (**5 points**) Give the linear programming relaxation of the integer linear program you defined in part b. Show that it is the dual of the linear program you defined in part a.

d. (**5 points**) Assuming $c(u, v)$ are integers, show that the integrality gap of both linear programs defined in part (a) and (c) is 1. You can use any existing result you may know.

e . (**Bonus 5 points**) Show that the integrality gap of the linear programming relaxation you define in part c. is still 1 even for arbitrary capacities $c(u, v) \in \mathbb{R}$ and $c(u, v) \geq 0$.