TDA206/DIT370: Discrete Optimization Final Exam

Instructor: Devdatt Dubhashi (dubhashi@chalmers.se)

March 16, 2022

Instructions:

- Write your answers to the point. Credit will only be assigned based on the correctness of what is requested e.g. if it is a LP, credit will be assigned based on the choice of decision variables, objective and constraints only. Anything else you write will not receive any credit. So use extra sheets to work out the answer and then only write the requested answer in your solution.
- You must upload **one pdf file** to Canvas as per the instructions on the course page. The pdf can be generated via LaTeX (preferred option) but it can also be handwritten and then scanned as a pdf (or photo image converted to pdf). Make sure to **join scanned images into one pdf** containing all your pictures/scans. The scanned images must be easily readable so make sure that you try and test out your scanning or imaging beforehand
- You may use the textbook and all other reading material mentioned together with the lectures on the course page.
- If you have questions, send an email at the address given above, they will be promptly answered between 8:30 AM and 12:30 PM.

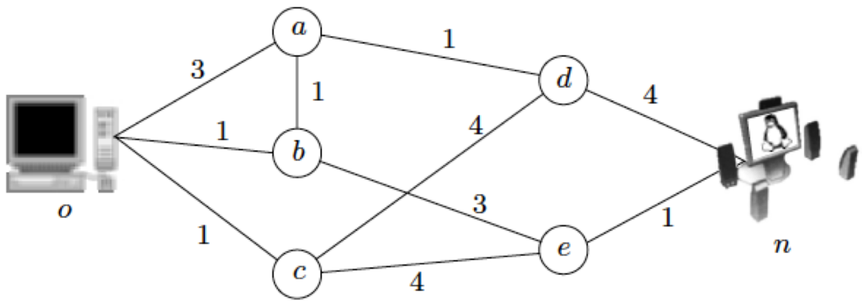

Figure 1: Flow network

1. (12 points) Consider the network flow problem discussed in the first lecture in the course: Figure 1 shows an *undirected* graph over which we'd like to solve for the maximum flow from o to n. In class and in the textbook, we used variables for the flow that could take positive or negative values. However Kalle doesn't like this and wants to write a formulation using only flow variables that take non-negative values. So, he introduces two non-negative flow variables corresponding to each edge. For example, in the network in Figure 1, $f_{a,d} \geq 0$ denotes the the flow *from* a to d while $f_{d,a} \geq 0$ stands for the flow in the opposite direction.

   (a) Write the objective function i.e. the flow leaving o in terms of the these variables.

   (b) Write the flow conservation constraint at vertex a and at vertex e in terms of these variables.

   (c) Write the capacity constraints for flow variables corresponding to the edge $\{c, e\}$.

   (d) Now consider a general undirected graph $G = (V, E)$ with capacities $c_{i,j}, \{i, j\} \in E$, source and target vertices $s, t \in V$. Write the general LP for this graph in terms of the variables introduced by Kalle.

2. (12 points) Solve the following LP using the Simplex algorithm. Show all tableaus, the BFS at each step, the objective function value and the incoming and outgoing variables of the pivot step.

$$
\begin{aligned}
\max \quad z \quad = \quad & 2x_1 + 3x_2 + 3x_3 \\
\text{s.t.} \quad & 3x_1 + 2x_2 \leq 60 \\
& -x_1 + x_2 + 4x_3 \leq 10 \\
& 2x_1 - 2x_2 + 5x_3 \leq 5 \\
& x_1, x_2, x_3 \geq 0
\end{aligned}
$$

3. (12 points) Consider the LP:

$$
\begin{aligned}
\max \quad z \quad = \quad & 2x_1 + 16x_2 + 2x_3 \\
\text{s.t.} \quad & 2x_1 + x_2 - x_3 \leq -3 \\
& -3x_1 + x_2 + 2x_3 \leq 12 \\
& x_1, x_2, x_3 \geq 0
\end{aligned}
$$

   Write the dual and Check whether each of the following is an optimal solution, using complementary slackness, giving justification:

   (a) $x_1 = 6, x_2 = 0, x_3 = 12.$

(b) $x_1 = 0, x_2 = 2, x_3 = 5$.

(c) $x_1 = 0, x_2 = 0, x_3 = 6$

4. (12 points) Apply the Primal Dual algorithm for the vertex cover problem to the following graphs. Show the dual variables at each step, the dual objective value, the final value of the solution and compare it to the optimal solution giving the approximation factor.

   (a) The cycle C_6 on 6 vertices: vertex i is connected to vertex $i+1$ for $i = 1 \cdots 5$ and vertex 6 is connected to vertex 1. All vertex weights are 1.

   (b) The cycle C_6 but even vertices have weight 1 and odd vertices have weight 2.

   (c) The complete graph K_6 on 6 vertices: all vertices are connected to each other. All vertex weights are 1. Generalize to K_n the complete graph on n vertices and give the limit of the approximation factor as $n \to \infty$.

5. (12 points) In class we wrote a SDP for MAXCUT, the problem of computing a maximum cut in an undirected graph $G = (V, E)$ with weights $w_{i,j}, i, j \in E$. Suppose now there are two distinguished vertices $s, t \in V$ and we would like to compute the maximum cut that puts these two vertices on different sides of the cut. So the MAX-st-CUT problem requires that we compute a cut $(S, V \setminus S)$ such that $s \in S$ and $t \in V \setminus S$ such that the weight of the cut $\sum_{i \in S, j \notin S} w_{i,j}$ is maximized.

   (a) Show that if you can solve MAX-st-CUT in polynomial time, then you can also solve MAXCUT in polynomial time.

   (b) Write an exact formulation of MAX-st-CUT using variables $x_i \in \{-1, +1\}, i \in V$.

   (c) Pass to the vector program relaxation of the exact formulation by introducing vectors on the unit sphere.

   (d) Write the SDP corresponding to the vector relaxation.

   (e) Give an algorithm to round the SDP solution to produce a cut that is guaranteed to separate s and t.