Exercise Set III, Advanced Algorithms 2022

These exercises are for your own benefit. Feel free to collaborate and share your answers with other students. **This exercise set contains many problems.** So solve as many problems as you can and ask for help if you get stuck for too long. Problems marked * are more difficult but also more fun :).

These problems are taken from various sources at EPFL and on the Internet, too numerous to cite individually.

1 *(*)* Consider the linear programming relaxation for minimum-weight vertex cover:

$$\begin{aligned} \text{Minimize} \quad & \sum_{v \in V} x_v w(v) \\ \text{Subject to} \quad & x_u + x_v \geq 1 \quad \forall \{u, v\} \in E \\ & 0 \leq x_v \leq 1 \qquad \forall v \in V \end{aligned}$$

In class, we saw that any extreme point is integral when considering bipartite graphs. For general graphs, this is not true, as can be seen by considering the graph consisting of a single triangle. However, we have the following statement for general graphs:

Any extreme point x^* satisfies $x^*_v \in \{0, \frac{1}{2}, 1\}$ for every $v \in V$.

Prove the above statement.

Solution: Consider an extreme point x^*, and suppose for the sake of contradiction that x^* is not half-integral, i.e., that there is an edge e such that $x^*_e \notin \{0, \frac{1}{2}, 1\}$. We will show that x^* is a convex combination of feasible points, contradicting that x^* is an extreme point. Let $V^+ = \{v : \frac{1}{2} < x^*_v < 1\}$ and $V^- = \{v : 0 < x^*_v < \frac{1}{2}\}$. Note that $V^+ \cup V^- \neq \emptyset$, since x^* is assumed to not be half-integral. Take $\epsilon > 0$ to be tiny, and define:

$$y_v^+ = \begin{cases} x^*_v + \epsilon & \text{if } v \in V^+ \\ x^*_v - \epsilon & \text{if } v \in V^- \\ x^*_v & \text{otherwise} \end{cases}$$

$$y_v^- = \begin{cases} x^*_v - \epsilon & \text{if } v \in V^+ \\ x^*_v + \epsilon & \text{if } v \in V^- \\ x^*_v & \text{otherwise} \end{cases}$$

Note that $x^* = \frac{1}{2}y^+ + \frac{1}{2}y^-$.
It remains to verify that y^+ and y^- are feasible solutions.

1. By selecting ϵ small enough, the boundary constraints $(0 \leq y^+_v \leq 1, 0 \leq y^-_v \leq 1)$ are satisfied.

2. Consider the constraints for the edges $e = \{u, v\} \in E$. If $x^*_u + x^*_v > 1$, the constraint remains satisfied by picking $\epsilon > 0$ small enough. If $x^*_u + x^*_v = 1$, then consider the following cases:

   - $u, v \notin V^+ \cup V^-$. In this case, $y^+_u + y^+_v = x^*_u + x^*_v = 1$.
   - $u \in V^+$; then $v \in V^-$. In this case, $y^+_u + y^+_v = x^*_u + \epsilon + x^*_v - \epsilon = 1$.
   - $u \in V^-$; then $v \in V^+$. In this case, $y^+_u + y^+_v = x^*_u - \epsilon + x^*_v + \epsilon = 1$.

So y^+ is a feasible solution. The same argument holds for y^-.

2 Write the dual of the following linear program:

$$\begin{aligned} \text{Maximize} \quad & 6x_1 + 14x_2 + 13x_3 \\ \text{Subject to} \quad & x_1 + 3x_2 + x_3 \leq 24 \\ & x_1 + 2x_2 + 4x_3 \leq 60 \\ & x_1, x_2, x_3 \geq 0 \end{aligned}$$

Hint: How can you convince your friend that the above linear program has optimum value at most z?

Solution: We convince our friend by taking $y_1 \geq 0$ multiples of the first constraints and $y_2 \geq 0$ multiplies of the second constraint so that

$$6x_1 + 14x_2 + 13x_3 \leq y_1(x_1 + 3x_2 + x_3) + y_2(x_1 + 2x_2 + 4x_3) \leq y_1 24 + y_2 60\,.$$

To get the best upper bound, we wish to minimize the right-hand-side $24y_1 + 60y_2$. However, for the first inequality to hold, we need that $y_1x_1 + y_2x_1 \geq 6x_1$ for all non-negative x_1 and so $y_1 + y_2 \geq 6$. The same argument gives us the constraints $3y_1 + 2y_2 \geq 14$ for x_2 and $y_1 + 4y_2 \geq 13$ for x_3. It follows that we can formulate the problem of finding an upper bound as the following linear program (the dual):

$$\begin{aligned} \text{Minimize} \quad & 24y_1 + 60y_2 \\ \text{Subject to} \quad & y_1 + y_2 \geq 6 \\ & 3y_1 + 2y_2 \geq 14 \\ & y_1 + 4y_2 \geq 13 \\ & y_1, y_2 \geq 0 \end{aligned}$$

3 Consider the min-cost perfect matching problem on a bipartite graph $G = (A \cup B, E)$ with costs $c : E \to \mathbb{R}$. Recall from the lecture that the dual linear program is

$$\begin{aligned} \text{Maximize} \quad & \sum_{a \in A} u_a + \sum_{b \in B} v_b \\ \text{Subject to} \quad & u_a + v_b \leq c(\{a, b\}) \qquad \text{for every edge } \{a, b\} \in E. \end{aligned}$$

Show that the dual linear program is unbounded if there is a set $S \subseteq A$ such that $|S| > |N(S)|$, where $N(S) = \{v \in B : \{u, v\} \in E \text{ for some } u \in S\}$ denotes the neighborhood of S. This proves (as expected) that the primal is infeasible in this case.

Solution: Let $v_b = 0$ for all $b \in B$ and $u_a = \min_{\{a,b\} \in E} c(\{a,b\})$ be a dual solution. By definition it is feasible. Now define the vector (u^*, v^*) by

$$u^*_a = \begin{cases} 1 & \text{if } a \in S \\ 0 & \text{otherwise} \end{cases} \qquad \text{and} \qquad v^*_b = \begin{cases} -1 & \text{if } b \in N(S) \\ 0 & \text{otherwise} \end{cases}$$

Note that $(u, v) + \alpha \cdot (u^*, v^*)$ is a feasible solution for any scalar $\alpha \geq 0$. Such a solution has dual value $\sum_{a \in A} u_a + \sum_{b \in B} v_b + \alpha \cdot \left(\sum_{a \in S} u^*_a - \sum_{b \in N(S)} v^*_b \right) = \sum_{a \in A} u_a + \sum_{b \in B} v_b + \alpha \cdot (|S| - |N(S)|)$, and as $|S| > |N(S)|$ this shows that the optimal solution to the dual is unbounded (letting $\alpha \to \infty$).

4 *(half a *)* Prove Hall's Theorem:

> "An n-by-n bipartite graph $G = (A \cup B, E)$ has a perfect matching if and only if $|S| \leq |N(S)|$ for all $S \subseteq A$."

(Hint: use the properties of the augmenting path algorithm for the hard direction.)

Solution: It is easy to see that if a bipartite graph has a perfect matching, then $|S| \leq |N(S)|$ for all $S \subseteq A$. This holds even if we only consider the edges inside the perfect matching. Now we focus on proving the other direction, i.e., if $|S| \leq |N(S)|$ for all $S \subseteq A$ then G has a perfect matching. We define a procedure that given a matching M with maximum size which does not cover $a_0 \in A$, it returns a set $S \subseteq A$ such that $|N(S)| < |S|$. This shows that the size of the matching should be n. To this end, let $A_0 = \{a_0\}$ and $B_0 = N(a_0)$. Note that all vertices of B_0 are covered by the matching M (if $b_0 \in B_0$ is not covered, the edge $a_0 b_0$ can be added to the matching which contradicts the fact that M is a maximum matching). If $B_0 = \emptyset$, $S = A_0$ is a set such that $|N(S)| < |S|$. Else, B_0 is matched with $|B_0|$ vertices of A distinct from a_0. We set $A_1 = N_M(B_0) \cup \{a_0\}$, where $N_M(B_0)$ is the set of vertices matched with vertices of B_0. We have $|A_1| = |B_0| + 1 \geq |A_0| + 1$. Let $B_1 = N(A_1)$. Again, no vertices in B_1 is exposed, otherwise there is an augmenting path. If $|B_1| < |A_1|$, the algorithm terminates with $|N(A_1)| < |A_1|$. If not, let $A_2 = N_M(B_1) \cup \{a_0\}$. Then $|A_2| \geq |B_1| + 1 \geq |A_1| + 1$. We continue this procedure till it terminates. This procedure eventually terminates since size of set A_i is strictly increasing. Hence it return a set $S \subseteq A$ such that $|N(A)| < |S|$. [1]

5 Consider the Maximum Disjoint Paths problem: given an undirected graph $G = (V, E)$ with designated source $s \in V$ and sink $t \in V \setminus \{s\}$ vertices, find the maximum number of edge-disjoint paths from s to t. To formulate it as a linear program, we have a variable x_p for each possible path p that starts at the source s and ends at the sink t. The intuitive meaning of x_p is that it should take value 1 if the path p is used and 0 otherwise[2]. Let P be the set of all such paths

[1] Some parts of this proof are taken from this link.

[2] I know that the number of variables may be exponential, but let us not worry about that.

from s to t. The linear programming relaxation of this problem now becomes

$$\begin{aligned} \text{Maximize} \quad & \sum_{p \in P} x_p \\ \text{subject to} \quad & \sum_{p \in P : e \in p} x_p \le 1, \qquad \forall e \in E, \\ & \quad x_p \ge 0, \qquad \forall p \in P. \end{aligned}$$

What is the dual of this linear program? What famous combinatorial problem do binary solutions to the dual solve?

Solution:

The dual is the following:

$$\begin{aligned} \text{minimize} \quad & \sum_{e \in E} y_e \\ \text{subject to} \quad & \sum_{e \in p} y_e \ge 1 \quad \forall p \in P, \\ & y_e \ge 0 \qquad \forall e \in E. \end{aligned}$$

Any binary solution $y \in \{0,1\}^{|E|}$ to the dual corresponds to a set of edges which, when removed from G, disconnect s and t (indeed, for every path p from s to t, at least one edge must be removed). This is called the minimum s,t-cut problem.