ChE 597 Computational Optimization

Homework 4 Solutions

1. Consider the following linear programming problem:

$$\text{minimize} \quad x_1 - x_2$$

Subject to the constraints:

$$\begin{aligned}
&2x_1 + 3x_2 - x_3 + x_4 \leq 0\\
&3x_1 + x_2 + 4x_3 - 2x_4 \geq 3\\
&-x_1 - x_2 + 2x_3 + x_4 = 6\\
&x_1 \leq 0\\
&x_2,\ x_3 \geq 0
\end{aligned}$$

Write down the corresponding dual problem.

Solution: The original problem is as follows:

$$\begin{array}{lll}
\text{minimize} & x_1 - x_2 & \\
\text{subject to} & 2x_1 + 3x_2 - x_3 + x_4 \leq 0 & (p_1)\\
& 3x_1 + x_2 + 4x_3 - 2x_4 \geq 3 & (p_2)\\
& -x_1 - x_2 + 2x_3 + x_4 = 6 & (p_3)\\
& x_1 \leq 0 & \\
& x_2 \geq 0 & \\
& x_3 \geq 0 & \\
& x_4 \text{ free} &
\end{array}$$

The dimension of the original optimization problem is $n = 4$ (variables) and $m = 3$ (equations); thus the number of constraints and the dimension of the dual problem will thus be $m' = 4$ (equations) and $n' = 3$ (variables) respectively. The dual variables corresponding to the primal constraints are listed alongside the primal constraints in the original problem.

Recall the way to take LP dual in slides:

$$
\begin{array}{llll}
\min & c^T x & & \\
\text{s.t.} & a_i^T x \geq b_i, & i \in M_1, \\
& a_i^T x \leq b_i, & i \in M_2, \\
& a_i^T x = b_i, & i \in M_3, \\
& x_j \geq 0, & j \in N_1, \\
& x_j \leq 0, & j \in N_2, \\
& x_j \text{ free}, & j \in N_3,
\end{array}
\qquad
\begin{array}{llll}
\max & p^T b & & \\
\text{s.t.} & p_i \geq 0, & i \in M_1, \\
& p_i \leq 0, & i \in M_2, \\
& p_i \text{ free}, & i \in M_3, \\
& p^T A_j \leq c_j, & j \in N_1, \\
& p^T A_j \geq c_j, & j \in N_2, \\
& p^T A_j = c_j, & j \in N_3.
\end{array}
$$

PRIMAL	minimize	maximize	DUAL
constraints	$\geq b_i$ $\leq b_i$ $= b_i$	≥ 0 ≤ 0 free	variables
variables	≥ 0 ≤ 0 free	$\leq c_j$ $\geq c_j$ $= c_j$	constraints

We can express the dual of the initial problem in the following manner:

$$
\begin{array}{ll}
\text{maximize} & 0p_1 + 3p_2 + 6p_3 \\
\text{subject to} & p_1 \leq 0 \\
& p_2 \geq 0 \\
& p_3 \text{ free} \\
& 2p_1 + 3p_2 - p_3 \geq 1 \\
& 3p_1 + p_2 - p_3 \leq -1 \\
& -p_1 + 4p_2 + 2p_3 \leq 0 \\
& p_1 - 2p_2 + p_3 = 0
\end{array}
$$

Rewriting in the conventional form for the sake of completeness, we have

$$
\begin{array}{ll}
\text{maximize} & 3p_2 + 6p_3 \\
\text{subject to} & 2p_1 + 3p_2 - p_3 \geq 1 \\
& 3p_1 + p_2 - p_3 \leq -1 \\
& -p_1 + 4p_2 + 2p_3 \leq 0 \\
& p_1 - 2p_2 + p_3 = 0 \\
& p_1 \leq 0 \\
& p_2 \geq 0
\end{array}
$$

2. In this question, we will show the ***max-flow*** problem and the ***min-cut*** problem are dual to each other. First, let's formulate both problems to facilitate solving a numerical problem in Pyomo.

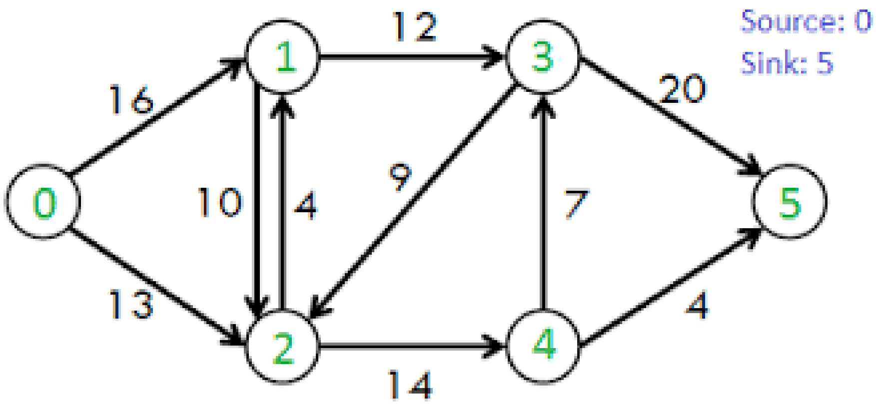

Figure 1: Directed graph (Network flow)

(a) **Formulating the *max-flow problem* as a LP problem:** Consider a directed graph $G = (V,E)$, where V denotes a set of vertices and $E \subseteq V \times V$ denotes a set of edges. Let $e = (u,v) \in E$ be an edge for vertex u to vertex v, we define $e \in \delta^+(v)$ to be the set of all outgoing edges to vertex v and $e \in \delta^-(v)$ to be the set of all incoming nodes from vertex v. Let $c(e)$, $e \in E$ be the capacity of the edge, i.e., the maximum amount of commodity that one can push through the edge. Let $f(e)$ be the flow across an edge e. $f(e)$ is non-negative as the flow in the edge cannot flow in the reverse direction i.e., towards the source. Also, the flow across each node should be conserved. In other words, the flow coming into a node has to be equal to the flow leaving the node. This conservation property does not apply to the source node (s) and the sink node (t). Formulating a linear programming problem to maximize the overall flow departing from the source node leads us to the formulation provided below:

$$
\begin{array}{llll}
\max & \sum_{e\in\delta^+(s)} f(e) - \sum_{e\in\delta^-(s)} f(e) & & \\
\text{subject to} & \sum_{e\in\delta^-(v)} f(e) = \sum_{e\in\delta^+(v)} f(e) & \forall v \in V \setminus \{s,t\} & \text{(flow conservation)} \\
& f(e) \le c(e) & \forall e \in E & \text{(capacity constraints)} \\
& f(e) \ge 0 & \forall e \in E &
\end{array}
$$

(b) **Formulating the *min-cut problem* as an LP problem:** The minimum cut problem requires identifying the minimal set of edges that, when removed, completely separates the source and sink vertices, creating two non-overlapping partitions in the graph. In other words, given any subset S of nodes with source node $s \in S$, let T be the set of the remaining nodes. The cut (S, T) is the set of edges $e = (u,v)$ with $u \in S$ and $v \in T$. It is so-called because removing all those edges in the cut would cut the flow from s to t. Let c(e), $e \in E$ be the capacity of the edge, i.e., the maximum amount of commodity that one can push through the edge. Let x_k, $\forall\ k \in V$ be a variable corresponding to the vertices and let y_e, $\forall\ e = (u,v) \in E$ be a variable corresponding to the edges. The min-cut problem can be formulated as the following:

$$
\begin{array}{lll}
\min & \sum_{e\in E} c(e) y_e & \\
\text{s.t.} & x_u - x_v + y_e \geq 0 & \forall e = (u,v) \in E \\
& -x_s + x_t \geq 1 & \\
& x_v \in \mathbb{R} & \forall v \in V \\
& y_e \geq 0 & \forall e \in E
\end{array}
$$

(a) For the directed graph in Figure 1, solve the max-flow problem using Pyomo.

(b) For the directed graph in Figure 1, solve the min-cut problem using Pyomo. Check if the problem have the same objective as (a).

(c) Show that the max-flow problem and the min-cut problem are duals of each other.

Hint: We can write an equivalent formulation for the maximum flow problem to facilitate the ease of obtaining its dual by introducing a new variable f^* and following the steps below:

(i). Replacing f^* with "$\sum_{e\in\delta^+(s)} f(e) - \sum_{e\in\delta^-(s)} f(e)$" in the objective function.

(ii). Adding constraints to compensate for the objective coefficient replacement.

$$\sum_{e\in\delta^+(s)} f(e) - \sum_{e\in\delta^-(s)} f(e) - f^* = 0$$

$$\sum_{e\in\delta^+(t)} f(e) - \sum_{e\in\delta^-(t)} f(e) + f^* = 0$$

The initial constraint involves setting f^* equal to the objective function in our initial max-flow LP formulation. The subsequent constraint pertains to the sink node. Given that the flow originating from the source node must traverse the sink node, we establish an equivalence between f^* and the flow entering the sink node.

(iii). Finally, the equivalent formulation for the max-flow problem is as follows:

$$
\begin{array}{lll}
\max & f^* & \\
\text{s.t.} & \sum_{e\in\delta^+(s)} f(e) - \sum_{e\in\delta^-(s)} f(e) - f^* = 0 & \\
& \sum_{e\in\delta^+(v)} f(e) - \sum_{e\in\delta^-(v)} f(e) = 0 & \forall v \in V \backslash \{s,t\} \\
& \sum_{e\in\delta^+(t)} f(e) - \sum_{e\in\delta^-(t)} f(e) + f^* = 0 & \\
& f(e) \leq c(e) & \forall e \in E \\
& f(e) \geq 0 & \forall e \in E \\
& f^* \geq 0 &
\end{array}
$$

Now, derive the dual for the above max-flow problem formulation and show that it is equivalent to the provided minimum cut problem formulation in the question.

Solution: (a) For the directed graph in Figure 1, solve the max-flow problem using Pyomo.

The Pyomo code for this problem can be found here.

(b) For the directed graph in Figure 1, solve the min-cut problem using Pyomo. Check if the problem has the same objective as (a).

The Pyomo code for this problem can be found here.

Observing that both problems yield the same objective value, namely 23, confirms the necessary condition for them to be duals of each other, demonstrating strong duality.

(c) Show that the max-flow problem and the min-cut problem are duals of each other.

The equivalent form of the max-flow problem given at the end of the question is as follows:

$$
\begin{array}{llll}
\max & f^* & & \\
\text{s.t.} & \sum_{e\in\delta^+(s)} f(e) - \sum_{e\in\delta^-(s)} f(e) - f^* = 0 & & (x_s) \\
& \sum_{e\in\delta^+(v)} f(e) - \sum_{e\in\delta^-(v)} f(e) = 0 & \forall v \in V\backslash\{s,t\} & (x_v) \\
& \sum_{e\in\delta^+(t)} f(e) - \sum_{e\in\delta^-(t)} f(e) + f^* = 0 & & (x_t) \\
& f(e) \leq c(e) & \forall e \in E & (y_e) \\
& f(e) \geq 0 & \forall e \in E & \\
& f^* \geq 0 & &
\end{array}
$$

Note: The dual variables corresponding to the primal constraints are listed alongside the primal constraints in the above problem.

Taking the dual of the above formulation involves the following steps:

We know that there is one primal variable for every dual constraint. Considering the first three constraint equations in the above dual LP formulation, we see that there is one constraint for each node in the graph. Hence, let us denote the variable to be x_v; $\forall\ v \in V$. Since all these constraints are equality constraints, x_v would be a free variable.

Next, we have edge capacity constraints, one for each edge. Let us consider the variable to be y_e; $\forall\ e \in E$. This variable would be nonnegative since we have a $\leq$ sign in the constraint.

Now, the objective function for the primal LP would be $\sum_{e\in E} c(e) y_e$. Remember that the RHS of dual constraints becomes the cost function for the primal problem. For the x_v variable, we have all the corresponding RHS to be zero, and for the y_e variable, we have the corresponding RHS to be $c(e)$. Hence, the objective function becomes $\sum_{e\in E} c(e)y_e$.

Next, the primal constraints can be deduced from the dual variables. We know that each dual variable corresponds to one primal constraint. Hence, we will have one constraint corresponding to every edge and one constraint corresponding to f^*. Since both these dual variable types are nonnegative, we will have a $\leq$ sign for the primal constraints.

Now, the RHS of the primal constraints correspond to the costs multiplied by the dual variable in the dual LP objective function, hence for primal constraints corresponding to variable y_e the RHS would be zero, and for primal constraint corresponding to f^*, the RHS would be 1.

Finally, to construct the LHS of the primal constraints, we need to check where the dual variables show up in the dual constraints. Considering the f^* variable, it shows up in two constraints, one corresponding source node and one corresponding sink node. Hence the LHS for the primal constraint corresponding to f^* dual variable would be $-x_s + x_t$ (make sure you check the signs properly). Finally, the variable $f(e)$ appears in all the nodes as well as edge constraints. Hence the corresponding RHS would be $x_u - x_v + y_e$, considering edge $e = (u, v) \in E$ exits node u and enters node v.

Expressing the dual problem mathematically, we have:

$$
\begin{array}{lll}
\min & \sum_{e\in E} c(e)y_e & \\
\text{s.t.} & x_u - x_v + y_e \geq 0 & \forall e = (u,v) \in E \\
 & -x_s + x_t \geq 1 & \\
 & x_v \in \mathbb{R} & \forall v \in V \\
 & y_e \geq 0 & \forall e \in E
\end{array}
$$

3. Interdiction game

Let us consider an interesting scenario. Suppose that terrorists may attack any of three sites, and the police can only patrol one site each day. To keep the terrorists guessing, the police will randomly choose a site to patrol each day. Similarly, to keep the police guessing, the terrorists will randomly choose a site to attack each day. Now, the question is with what probability should the police and terrorists choose each site to maximize their expected utility?

Let the gain to police entails an equal loss to the terrorists (zero-sum game). The table given below indicates the gain to the police for each possible outcome.

		police		
		site 1	site 2	site 3
terrorists	site 1	4	−10	−10
	site 2	−8	5	−8
	site 3	−12	−12	9

For example, if police patrol site 2 but terrorists hit site 1, the police lose 10 and the terrorists gain 10.

Let the probability of police choosing site j be x_j and terrorists choosing site i is u_i. Let the table above be the payoff (utility) matrix A. The police assume that whatever probabilities they choose, the terrorists will choose an optimal strategy based on those probabilities. Therefore, they wish to find x^* by solving the following problem:

$$\max_{\substack{x\geq 0\\ e^T x=1}} \left\{ \min_{\substack{u\geq 0\\ e^T u=1}} \left\{ u^T A x \right\} \right\}$$

The constraint $e^T x = 1$, where e is a vector of ones, ensures that the probabilities sum to 1 .

The min problem is trivial to solve, since Ax is a fixed vector. If component i of Ax is the smallest component, let $u_i = 1$ and all other u_k 's vanish. So it can be written as

$$\max_{\substack{x\geq 0\\ e^T x=1}} \left\{ \min_{i} \left\{ e_i^T A x \right\} \right\}$$

where e_i is the i th unit vector (1 in position $i, 0$ in all other positions). This problem can be written

$$\begin{array}{ll} \max & z \\ \text{s.t.} & z \leq e_i^T A x, \text{ all } i \\ & e^T x = 1 \\ & x \geq 0 \end{array}$$

where z is a scalar.

or equivalently

$$\begin{array}{ll} \max_{z,x} & z \\ \text{s.t.} & ze - Ax \leq 0 \\ & e^T x = 1 \\ & x \geq 0 \end{array}$$

Where e is a vector of ones to ensure that the probabilities sum to one.

Similarly, the terrorists would try to find u^* by solving the following problem:

$$\min_{\substack{u\geq 0\\ e^T u=1}} \left\{ \max_{\substack{x\geq 0\\ e^T x=1}} \left\{u^T Ax\right\} \right\}$$

which can be rewritten as

$$\begin{array}{ll} \min_{w,u} & w \\ \text{s.t.} & we^T - u^T A \geq 0 \\ & u^T e = 1 \\ & u \geq 0 \end{array}$$

where w is a scalar.

(a) Calculate the optimal solution for the police problem using Pyomo.

(b) Calculate the optimal solution for the terrorist problem using Pyomo. Check if the problem have the same objective as (a).

(c) Show that the police problem and the terrorist problem are duals of each other.
Hint: it suffices to show that the dual of the terrorist problem is the police problem. Alternatively, you can also show that the dual of the police problem is the terrorist problem.

Solution: (a) Calculate the optimal solution for the police problem using Pyomo.

The Pyomo code for this problem can be found here.

(b) Calculate the optimal solution for the terrorist problem using Pyomo. Check if the problem have the same objective as (a).

The Pyomo code for this problem can be found here.

Observing that both problems yield the same objective value, namely -4.598, confirms the necessary condition for them to be duals of each other, demonstrating strong duality.

(c) Show that the police problem and the terrorist problem are duals of each other.

The formulation for the Police problem is as follows:

$$\begin{array}{lll} \max_{z,x} & z & \\ \text{s.t.} & ze - Ax \leq 0 & (u \in \mathbb{R}^n) \\ & e^T x = 1 & (w \in \mathbb{R}) \\ & x \geq 0 & \end{array}$$

Considering $A \in \mathbb{R}^{n\times n}$, $x \in \mathbb{R}^n$, $z \in \mathbb{R}$ and $e \in \mathbb{R}^n$ (vector of ones), the problem formulation has n inequality constraints and 1 equality constraint. With n variables for x and one for w, the total count of variables is $n+1$.

It's essential to note that this remains an optimization problem rather than merely a linear system to solve, as it involves n inequality constraints. Upon transforming the problem into standard form, the addition of slack variables results in more variables than constraints.

Therefore, in the dual of the police problem, there will be $n+1$ variables and constraints respectively. Let's designate w and u as the dual variables corresponding to the equality constraint and the inequality constraints in the primal problem.The dual variables corresponding to the primal constraints are listed alongside the primal constraints in the original problem. //

Utilizing the rules outlined in Figure 3, we can express the dual problem as follows:

$$
\begin{array}{ll}
\min_{w,u} & \vec{0}^T u + 1 * w \\
\text{s.t.} & u \geq 0 \\
& w(free) \\
& -u^T A + we^T \geq 0 \\
& u^T e = 1
\end{array}
$$

Here, $\vec{0}$ is a zero vector. Rewriting in the conventional form for the sake of completeness, we have

$$
\begin{array}{ll}
\min_{w,u} & w \\
\text{s.t.} & we^T - u^T A \geq 0 \\
& u^T e = 1 \\
& u \geq 0
\end{array}
$$

The above formulation is what we had for the terrorist problem. Hence, it can be concluded that the police problem and the terrorist problem are duals of each other.

$$
\begin{array}{llllll}
\text{minimize} & \mathbf{c}'\mathbf{x} & & \text{maximize} & \mathbf{p}'\mathbf{b} & \\
\text{subject to} & \mathbf{a}_i'\mathbf{x} \geq b_i, & i \in M_1, & \text{subject to} & p_i \geq 0, & i \in M_1, \\
& \mathbf{a}_i'\mathbf{x} \leq b_i, & i \in M_2, & & p_i \leq 0, & i \in M_2, \\
& \mathbf{a}_i'\mathbf{x} = b_i, & i \in M_3, & & p_i \text{ free}, & i \in M_3, \\
& x_j \geq 0, & j \in N_1, & & \mathbf{p}'\mathbf{A}_j \leq c_j, & j \in N_1, \\
& x_j \leq 0, & j \in N_2, & & \mathbf{p}'\mathbf{A}_j \geq c_j, & j \in N_2, \\
& x_j \text{ free}, & j \in N_3, & & \mathbf{p}'\mathbf{A}_j = c_j, & j \in N_3.
\end{array}
$$

Figure 2: Primal -Dual of an LP

4. Let A be a symmetric square matrix. Consider the linear programming problem

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t. } & Ax \geq c \\ & x \geq 0. \end{aligned}$$

Prove that if x^* satisfies $Ax^* = c$ and $x^* \geq 0$, then x^* is an optimal solution.

Solution: The dual problem for the problem given in the question can be written as follows:

$$\begin{array}{llll} \text{minimize} & \mathbf{c}^T\mathbf{x} & \text{maximize} & \mathbf{p}^T\mathbf{c} \\ \text{subject to} & \mathbf{A}\mathbf{x} \geq \mathbf{c} & \text{subject to} & \mathbf{p}^T\mathbf{A} \leq \mathbf{c}^T \\ & \mathbf{x} \geq \mathbf{0}, & & \mathbf{p} \geq \mathbf{0}. \end{array}$$

Let us assume that the primal problem has a feasible solution $\mathbf{x}^*$, i.e., $\mathbf{A}\mathbf{x}^* = \mathbf{c}$ and $\mathbf{x}^* \geq \mathbf{0}$.

Since $\mathbf{A}$ is square, both primal and dual problems will have the same dimension, i.e., the same number of constraints and variables. Furthermore, notice that $\mathbf{p}^T\mathbf{A} \leq \mathbf{c}^T$ is equivalent to $\mathbf{A}^T\mathbf{p} \leq \mathbf{c}$, since $\mathbf{A}$ is symmetric, i.e., $\mathbf{A}^T = \mathbf{A}$.

Hence, we conclude that the condition $\mathbf{p}^T\mathbf{A} \leq \mathbf{c}^T$ in the dual problem is equivalent to $\mathbf{A}\mathbf{p} \leq \mathbf{c}$. Thus, if we set $\mathbf{p}^* := \mathbf{x}^*$, we see that $\mathbf{p}^*$ is feasible for the dual problem since $\mathbf{A}\mathbf{p}^* = \mathbf{c}$ satisfies the constraint $\mathbf{A}\mathbf{p} \leq \mathbf{c}$, and also a feasible solution to the primal problem since $\mathbf{p}^* := \mathbf{x}^*$ and $\mathbf{x}^*$ is a feasible solution to the primal problem.

Moreover, the objective values of both the primal, $\mathbf{c}^T\mathbf{x}$, and the dual, $\mathbf{p}^T\mathbf{c}$, coincide, thus fulfilling the strong duality theorem for linear programs (LPs). Consequently, $\mathbf{x}^* = \mathbf{p}^*$ must be the optimal solution to both the primal and dual problems.

This ends the proof.

5. Solve the following SDP problem using the cutting plane algorithm in python.

$$\max_{X \in \mathcal{S}^4} \sum_{i,j,i \neq j} Q_{ij} \cdot (1 - X_{ij})$$
$$\text{diag}(X) = e$$
$$X \succeq 0$$

where $Q = \begin{bmatrix} 0 & 1 & 3 & 1 \\ 1 & 0 & 0 & 2 \\ 3 & 0 & 0 & 4 \\ 1 & 2 & 4 & 0 \end{bmatrix}$

$\text{diag}(X) = e$ means the diagonal entries of X equal to 1.

Hint: the linear program in the cutting plane algorithm can be solved using Gurobi. The eigenvalue decomposition can be done using packages like numpy. The algorithm can terminate when the smallest eigenvalue of X is greater than -10^{-4}