Note: Your TA probably will not cover all the problems. This is totally fine, the discussion worksheets are not designed to be finished in an hour. They are deliberately made long so they can serve as a resource you can use to practice, reinforce, and build upon concepts discussed in lecture, readings, and the homework.

Zero Sum Games: In this game, there are two players: a maximizer and a minimizer. We generally write the payoff matrix M in the perspective of the maximizer, so every row corresponds to an action that the maximizer can take, every column corresponds to an action that the minimizer can take, and a positive entry corresponds to the maximizer winning. M is a n by m matrix, where n is the number of choices the maximizer has, and m is the number of choices the minimizer has.

$$M = \begin{bmatrix} M_{1,1} & M_{1,2} & \cdots & M_{1,m} \\ M_{2,1} & M_{2,2} & \cdots & M_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ M_{n,1} & M_{n,2} & \cdots & M_{n,m} \end{bmatrix}$$

A linear program that represents fixing the maximizer's choices to a probabilistic distribution where the maximizer has n choices, and the probability that the maximizer chooses choice i is p_i is the following:

$$\begin{gathered} \max \ z \\ M_{1,1} \cdot p_1 + \cdots + M_{n,1} \cdot p_n \geq z \\ M_{1,2} \cdot p_1 + \cdots + M_{n,2} \cdot p_n \geq z \\ \vdots \\ M_{1,m} \cdot p_1 + \cdots + M_{n,m} \cdot p_n \geq z \\ p_1 + p_2 + \cdots + p_n = 1 \\ p_1, p_2, \ldots, p_n \geq 0 \end{gathered}$$

or in other words,

$$\max z \text{ s.t. } M^\top p \geq z\mathbf{1}, \ \mathbf{1}^\top p = 1, \ p \geq 0$$

where $p = (p_1, \ldots, p_n)$.

The dual represents fixing the minimizer's choices to a probabilistic distribution. If we let the probability that the minimizer chooses choice j be q_j, then the dual is the following:

$$\begin{gathered} \min \ w \\ M_{1,1} \cdot q_1 + \cdots + M_{1,m} \cdot q_m \leq w \\ M_{2,1} \cdot q_1 + \cdots + M_{2,m} \cdot q_m \leq w \\ \vdots \\ M_{n,1} \cdot q_1 + \cdots + M_{n,m} \cdot q_m \leq w \\ q_1 + q_2 + \cdots + q_m = 1 \\ q_1, q_2, \ldots, q_m \geq 0 \end{gathered}$$

or in other words,

$$\min w \text{ s.t. } Mq \leq w\mathbf{1},\ \mathbf{1}^\top q = 1,\ q \geq 0$$

where $q = (q_1, \ldots, q_m)$.

By strong duality, the optimal value of the game is the same regardless of whether you fix the minimizer's distribution first or the maximizer's distribution first; i.e. $z^* = w^*$, or $z^* + (-w^*) = 0$.

1 Zero-Sum Games Short Answer

(a) Suppose a zero-sum game has the following property: The payoff matrix M satisfies $M = -M^\top$. What is the expected payoff of the row player?

Hint: try rewriting the minimizer's (i.e. column player's) LP as a maximization problem. Then, use the definition of a ZSG (i.e. when both players try to maximize their payoff, their optimal payoffs sum to 0).

(b) True or False: If every entry in the payoff matrix is either 1 or -1 and the maximum number of 1s in any row is k, then for any row with less than k 1s, the row player's optimal strategy chooses this row with probability 0. Justify your answer.

(c) True or False: Let M_i denote the ith row of the payoff matrix. If $M_1 = \frac{M_2+M_3}{2}$, then there is an optimal strategy for the row player that chooses row 1 with probability 0. Justify your answer.

2 Domination

In this problem, we explore a concept called *dominated strategies*. Consider a zero-sum game with the following payoff matrix for the row player:

Column:

		A	B	C
	D	1	2	-3
Row:	E	3	2	-2
	F	-1	-2	2

Note: All parts of this problem can be solved without using an LP solver or solving a system of linear equations.

(a) If the row player plays optimally, can you find the probability that they pick D without directly solving for the optimal strategy? Justify your answer.

Hint: How do the payoffs for the row player picking D compare to their payoffs for picking E?

(b) Given the answer to part a, if the both players play optimally, what is the probability that the column player picks A? Justify your answer.

(c) Given the answers to part a and b, what are both players' optimal strategies?

This content is protected and may not be shared, uploaded, or distributed.

Flow. The *capacity* indicates how much flow can be allowed on an edge. Given a directed graph $G = (V, E)$ with edge capacites $c(u, v)$ (for all $(u, v) \in E$) and vertices $s, t \in V$, a flow is a mapping $f : E \to \mathbb{R}^+$ that satisfies

- Capacity constraint: $f(u, v) \leq c(u, v)$, the flow on an edge cannot exceed its capacity.
- Conservation of flows: $f^{\text{in}}(v) = f^{\text{out}}(v)$, flow in equals flow out for any $v \notin \{s, t\}$

Here, we define $f^{\text{in}}(v) = \sum_{u:(u,v)\in E} f(u, v)$ and $f^{\text{out}}(v) = \sum_{u:(v,u)\in E} f(u, v)$. We also define $f(v, u) = -f(u, v)$, and this is called *skew-symmetry.* Note that the total flow in the graph is $\sum_{v:(s,v)\in E} f(s, v) = \sum_{u:(u,t)\in E} f(u, t)$, where s is the source node of the graph and t is the target node.

Residual Graph. Given a flow network (G, s, t, c) and a flow f, the *residual capacity* (w.r.t. flow f) is denoted by $c_f(u, v) = c_{uv} - f_{uv}$. And the *residual network* $G_f = (V, E_f)$ where $E_f = \{(u, v) : c_f(u, v) > 0\}$.

Max Flow. Given a directed graph $G = (V, E)$ with edge capacities $c(u, v)$ for all $(u, v) \in E$ and vertices $s, t \in V$, the goal is to compute the maximum flow that can go from s to t. This can be solved with either Ford-Fulkerson (or its optimized variant Edmonds-Karp).

Ford-Fulkerson. Keep pushing along $s-t$ paths in the residual graph and update the residual graph accordingly. The runtime of this algorithm is $O(mF)$, where $m = |E|$ and F is the value of the max flow.

Edmonds-Karp. We can implement Ford-Fulkerson using BFS to find the $s - t$ paths. This yields a faster runtime of $O(nm^2)$, where $n = |V|, m = |E|$.

Min-Cut. Given a directed graph $G = (V, E)$ with edge weights $w(u, v)$ for all $(u, v) \in E$ and vertices $s, t \in V$, the goal is to compute the lightest $s - t$ cut (i.e. the cut separating t from s with the smallest sum of edge weights crossing it). Note that the Min Cut problem is the dual of the Max Flow problem.

3 Max-Flow Min Cut Basics

For each of the following, state whether the statement is True or False. If true provide a short proof, if false give a counterexample.

(a) If all edge capacities are distinct, the max flow is unique.

(b) If all edge capacities are distinct, the min cut is unique.

(c) If all edge capacities are increased by an additive constant, the min cut remains unchanged.

(d) If all edge capacities are multiplied by a positive integer, the min cut remains unchanged.

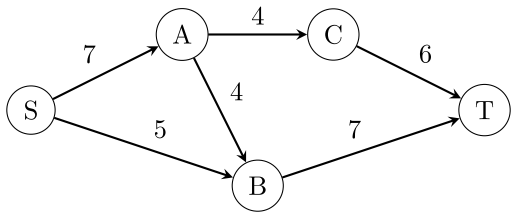
(e) In any max flow, there is no directed cycle on which every edge carries positive flow.

(f) There exists a max flow such that there is no directed cycle on which every edge carries positive flow.

This content is protected and may not be shared, uploaded, or distributed.

4 Residual in graphs

Consider the following graph with edge capacities as shown:



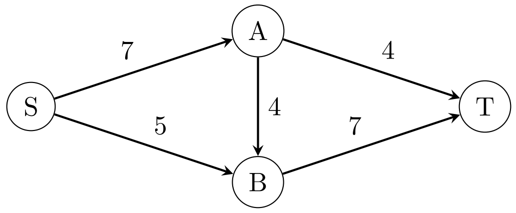
(a) Consider pushing 4 units of flow through $S \to A \to C \to T$. Draw the residual graph after this push.

(b) Compute a maximum flow of the above graph. Find a minimum cut. Draw the residual graph of the maximum flow.

5 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 graph 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:

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

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?

6 Flow vs LP

You play a middleman in a market of m suppliers and n purchasers. The i-th supplier can supply up to $s[i]$ products, and the j-th purchaser would like to buy up to $b[j]$ products.

However, due to legislation, supplier i can only sell to a purchaser j if they are situated at most 1000 miles apart. Assume that you're given a list L of all the pairs (i, j) such that supplier i is within 1000 miles of purchaser j. Given m, n, $s[1..m]$, $b[1..n]$, and L as input, your job is to compute the maximum number of products that can be sold. The runtime of your algorithm must be polynomial in m and n.

Show how to solve this problem using a network flow algorithm as a subroutine. Describe the graph and explain why the output from running the network flow algorithm on your graph gives a valid solution to this problem.