Note: Your TA probably will not cover all the problems. This is totally fine. The discussion worksheets 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.

1 Provably Optimal

Consider the following linear program:

$$\begin{aligned} \max \ & x_1 - 2x_3 \\ & x_1 - x_2 \leq 1 \\ & 2x_2 - x_3 \leq 1 \\ & x_1, x_2, x_3 \geq 0 \end{aligned}$$

For the linear program above,

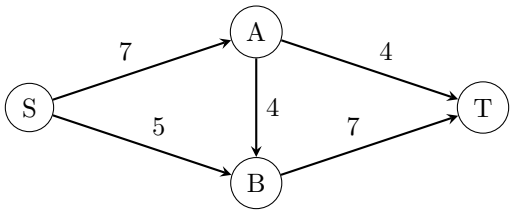
(a) First compute the dual of the above linear program

(b) show that the solution $(x_1, x_2, x_3) = (3/2, 1/2, 0)$ is optimal **using its dual**. You do not have to solve for the optimum of the dual. (*Hint:* Recall that any feasible solution of the dual is an upper bound on any feasible solution of the primal)

2 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:

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

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?

3 Advertising

Pen & Perry's has a business competitor, Cold Pebble. Both of these dessert shops sell ice-pops, ice-creams, and ice-cakes. When both shops advertise the same thing, Cold Pebble gains an extra two customers. Otherwise, when Pen & Perry's advertises ice-pops, they gain one extra customer, and when they advertise ice-cakes, they gain three extra customers. In all other scenarios, neither shop gains nor loses customers.

(a) Representing Pen & Perry's as the row and Cold Pebble as the column, what is the payoff matrix (for row's gains)?

(b) Suppose at the start of the day, Pen & Perry announces that there's a 20% chance they will advertise ice-pops, 30% chance for ice-creams, and a 50% chance for ice-cakes. What is Cold Pebble's optimal advertising strategy?

(c) Write the corresponding LP for the optimal strategy for Pen & Perry's.

(d) Write the corresponding LP for the optimal strategy for Cold Pebble.

4 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.

5 NP Basics

Assume A reduces to B in polynomial time. In each part you will be given a fact about one of the problems. What information can you derive of the other problem given each fact? Examples of information we can gather: "B is NP," "A is not in P," or "no additional information." Each part should be considered independent; i.e., you should not use the fact given in part (a) as part of your analysis of part (b).

For each part, briefly explain your reasoning.

(a) A is in P.

(b) B is in P.

(c) A is NP-hard.

(d) B is NP-hard.

6 Bad Reductions

In each part we make a wrong claim about some reduction. Explain for each one why the claim is wrong.

(a) The shortest simple path problem with non-negative edge weights can be reduced to the longest simple path problem by just negating the weights of all edges. There is an efficient algorithm for the shortest simple path problem with non-negative edge weights, so there is also an efficient algorithm for the longest path problem.

(b) We have a reduction from problem B to problem A that takes an instance of B of size n, and creates a corresponding instance of A of size n^2. There is an algorithm that solves A in quadratic time. So our reduction also gives an algorithm that solves B in quadratic time.

(c) We have a reduction from problem B to problem A that takes an instance of B of size n, and creates a corresponding instance of A of size n in $O(n^2)$ time. There is an algorithm that solves A in linear time. So our reduction also gives an algorithm that solves B in linear time.

(d) Minimum vertex cover can be reduced to shortest path in the following way: Given a graph G, if the minimum vertex cover in G has size k, we can create a new graph G' where the shortest path from s to t in G' has length k. The shortest path length in G' and size of the minimum vertex cover in G are the same, so if we have an efficient algorithm for shortest path, we also have one for vertex cover.

7 Cycle Cover

In the cycle cover problem, we have a directed graph G, and our goal is to find a set of directed cycles $C_1, C_2, \ldots C_k$ in G such that every vertex appears in exactly one cycle (a cycle cannot revisit vertices, e.g. $a \to b \to a \to c \to a$ is not a valid cycle, but $a \to b \to c \to a$ is), or declare none exists.

In the bipartite perfect matching problem, we have a undirected bipartite graph (a graph where the vertices can be split into L, R, and there are no edges between two vertices in L or two vertices in R), and our goal is to find a set of edges in this graph such that every vertex is adjacent to exactly one edge in the set, or declare none exists.

Give a reduction from cycle cover to bipartite perfect matching.

(Hint: In a cycle cover, every vertex has one incoming and one outgoing edge.)

8 NP or not NP, that is the question

For the following questions, circle the (unique) condition that would make the statement true.

Note: for now, you may not be able to solve all of these questions formally. Try to solve these intuitively! Revisit these problems after Thursday's lecture and try to solve them more formally then.

(a) Minimum Spanning Tree is in NP.

Always True True iff P = NP True iff P ≠ NP Always False

(b) 2-SAT is NP-complete. (Informally, 2-SAT is one of the hardest problems in NP)

Always True True iff P = NP True iff P ≠ NP Always False

(c) If B is in NP, then for any problem $A \in$ P, there exists a polynomial-time reduction from A to B. (Informally, B is at least as hard as A)

Always True True iff P = NP True iff P ≠ NP Always False

(d) If B is NP-complete, then for any problem $A \in$ NP, there exists a polynomial-time reduction from A to B. (Informally, B is at least as hard as A)

Always True True iff P = NP True iff P ≠ NP Always False

