CMPUT 675 - Final Assignment

Winter 2021, University of Alberta

Pages: 6

There are 2 categories of problems: quick and maybe not quick. You must solve:

- 4 quick problems
- 3 maybe not quick problems

If you solve more from a category than is required, clearly indicate which ones you want to be considered for marks. I can still provide feedback on the others, but it would be just for your own learning.

As always, for problems with multiple parts you may assume one part is true (even if you fail to prove it) and use it in a later part.

If you would like clarification on any question, please ask me as early as you can!

Do This! While the deadline is one week after the release date, you are encouraged to look over all problems as soon as this exam is released so you can start thinking about them.

1 Quick Problems: 2 marks each (8 marks total)

These are problems that have fairly short solutions. You must complete **four** of them.

1.1 Cycle Covers

Let $G = (V, E)$ be a directed graph. A **cycle partition** of G is a collection of directed cycles $\mathcal{C} = \{C_1, \ldots, C_\ell\}$ such that each $v \in V$ lies on precisely one cycle in $\mathcal{C}$.

Describe how to compute a cycle partition in polynomial time or determine that none exists.

1.2 Minimum Degree Graphs

Let $G = (V, E)$ be an undirected graph and $k \geq 1$ be an integer such that **at most** one $v \in V$ has $|\delta(v)| < k$ (i.e. at least $|V| - 1$ vertices have degree $\geq k$).

Prove there exists distinct nodes $u, v \in V$ such that there are k edge-disjoint paths from u to v.

Hint: Gomory-Hu tree

1.3 Edge-Disjoint Spanning Trees

Let $G = (V, E)$ be an undirected graph and $k \geq 1$ an integer such that $|\delta(S)| \geq 2 \cdot k$ for every $\emptyset \subsetneq S \subsetneq V$.

Prove there are k edge-disjoint spanning trees. Namely, there are spanning trees $T_1, \ldots, T_k \subseteq E$ such that $T_i \cap T_j = \emptyset$ for any pair $1 \leq i < j \leq k$.

Hint: Partition matroid rank function

Comment: This result is tight for every integer $k \geq 1$ in the sense that sometimes we can only find k edge-disjoint spanning trees (not more). For example, consider the clique K_n where $n \geq 3$ is odd. The cuts all have at least $n - 1$ edges. On the other hand, by counting edges we see the maximum number of edge-disjoint spanning trees is at most $(n - 1)/2$.

1.4 Bounded-Degree Bipartite Graphs

Let $G = (L \cup R, E)$ be bipartite graph whose maximum degree is d. That is, $|\delta(u)| \leq d$ for each $u \in V$ and there is at least one vertex $v \in V$ with $|\delta(v)| = v$.

You may assume G is simple (no parallel edges), but this result does hold even if it is not simple.

- Show there is a matching $M \subseteq E$ such that $\delta(v) \cap M \neq \emptyset$ for each $v \in V$ with $|\delta(v)| = d$. That is, M matches all vertices with the maximum degree.
- Conclude the line graph of a bipartite graph is perfect. That is, show the graph $L(G) := (E, F)$ whose vertices are actually edges of G and edges are $F = \{ee' : e \text{ and } e' \text{ share an endpoint in } G\}$ is a perfect graph.

Hint: It is possible to do this with combinatorial arguments involving alternating paths, but a substantially easier way is to use properties of the bipartite matching polytope.

1.5 Matroid Mappings

Let $\mathcal{M} = (X, \mathcal{I})$ be a matroid and suppose it has an independence oracle that runs in $poly(|X|)$ time (i.e. we can decide if $A \in \mathcal{I}$ in $poly(|X|)$ time for any $A \subseteq X$).

Let Y be another finite set and let $f : X \to Y$ be an arbitrary function that can be evaluated in $poly(|X|)$ time.

- Show $\mathcal{M}_f = (Y, \mathcal{I}')$ is a matroid where $\mathcal{I}' = \{f(A) : A \in \mathcal{I}\}$.
  Here, $f(A) = \{y \in Y : \exists\, x \in A \text{ such that } f(x) = y\}$.
- Show how to determine if some $A \subseteq Y$ lies in $\mathcal{I}'$ in $poly(|X|, |Y|)$ time.
  Hint: Matroid intersection.

1.6 Sizes of Laminar Families

Let X be a nonempty finite set and $\mathcal{L}$ a laminar family of nonempty subsets of X. That is, $\mathcal{L} \subseteq 2^X$, $\emptyset \notin \mathcal{L}$, and for any two $A, B \in \mathcal{L}$ we have $A \subseteq B, B \subseteq A$ or $A \cap B = \emptyset$.

- Prove $|\mathcal{L}| \leq 2 \cdot |X| - 1$.
- For each $n \geq 1$ describe an instance with $|X| = n$ where $|\mathcal{L}| = 2n - 1$.
- Now suppose $|A| \geq 2$ for each $A \in \mathcal{L}$. Prove $|\mathcal{L}| \leq |X| - 1$.

Comment
The last item is interesting because our proof of integrality of the spanning tree polytope showed there is a laminar family of tight constraints spanning $\mathbb{R}^E$ (after deleting the 0-edges) where each set S in this family has $|S| \geq 2$. The last part of this exercise would show there are then at most $|V| - 1$ edges, so all must be integral due to the $x(E[V]) = |V| - 1$ constraint. That is, one could complete the proof of integrality without using total unimodularity.

2 Maybe Not Quick Problems: 4 mark each (12 marks total)

There are solutions to these problems that do not take much time to write up, but will likely take a bit more effort to solve. You must complete **three** of these problems.

2.1 Doubly-Stochastic Matrices

Let $M \in \mathbb{R}_{\geq 0}^{n \times n}$ be a square matrix of nonnegative values such that each row and each column sums to 1. That is:

- $\sum_{j=1}^{n} M_{i,j} = 1$ for each $1 \leq i \leq n$, and
- $\sum_{i=1}^{n} M_{i,j} = 1$ for each $1 \leq j \leq n$.

Finally, say $P \in \mathbb{R}^{n \times n}$ is a **permutation matrix** if each entry of P is either 0 or 1 and each row and each column contain a single 1 (i.e. P can be obtained by permuting rows of the identity matrix).

- Prove there is a permutation matrix P such that $P_{i,j} = 0$ if $M_{i,j} = 0$ (that is, the 1s in P can only be in nonzero positions of M).
- For this P, show we can write $M = \lambda \cdot P + (1 - \lambda) \cdot M'$ where $0 \leq \lambda \leq 1$ and M' is another doubly-stochastic matrix with at least one more zero entry than M.
- Conclude we can write M as a convex combination of at most $n^2 - n + 1$ permutation matrices. For partial credit, you may simply show some polynomial number of permutation matrices suffice.

2.2 Exploring Preflows

Let $G = (V, E)$ be a directed graph with edge capacities $\mu : E \to \mathbb{R}_{\geq 0}$. Let $s, t \in V$ be distinct vertices.

An s-preflow is called **t-maximum** if $ex_f(t)$ is as large as possible among all s-preflows (you may want to consult the lectures for definitions of preflows and excess).

- Show for any t-maximum s-preflow f, there exists a maximum $s - t$ **flow** f' with $f'(e) \leq f(e)$ for each $e \in E$.

  Hint: Argue one can return some excess inflow to a node $v \neq t$ back to the source s without changing $ex_f(t)$.
- Show how to compute a maximum $s - t$ flow in $O(|V| \cdot |E|)$ time when given a t-maximum s-preflow.
- Let f be an s-preflow such that G_f has no $s - t$ path. Prove $-ex_f(s) \geq F$ where F is the value of a maximum $s - t$ flow (equivalently, the capacity of a minimum cut).

2.3 Compact LPs

Let $G = (V, E)$ be a directed graph with a designated root node $r \in V$ and edge costs $c : E \to \mathbb{R}_{\geq 0}$. Let $k \geq 1$ be an integer. Consider the problem of finding the cheapest $F \subseteq E$ such that there are at least k edge-disjoint paths from r to any vertex v using only edges in F. Equivalently, in the graph (V, F), even if we delete $k - 1$ edges from F the root r can still reach every other node.

You proved the following LP relaxation was integral in an assignment.

$$
\begin{array}{rrcll}
\textbf{minimize}: & \sum_{e \in E} c(e) \cdot x_e & & & \\
\textbf{subject to}: & x(\delta^{in}(S)) & \geq & k & \forall\, \emptyset \subsetneq S \subseteq V - \{r\} \\
& x_e & \leq & 1 & \forall\, e \in E \\
& x & \geq & 0 &
\end{array}
$$

Describe how to reformulate the LP so **the number of variables and constraints is polynomial in $|V|$ and $|E|$.**

Consider adding new variables y (indexed by appropriate items) and writing an LP over variables (x, y) with polynomially-many constraints such that:

- Given a feasible solution x to the exponentially-large LP described above, there is some y such that (x, y) is feasible for your new formulation.
- Given a point (x, y) that is feasible for your new formulation, x is then feasible for the exponentially-large LP.

In this way, you can solve the LP by solving the polynomially-size variant you described (and not rely on separation oracles).

Hint: The original constraints assert the minimum cut separating r from any $v \in V - \{r\}$ has x-value at least k. Instead of trying to enforce this cut condition for every subset, use flows to the vertices and let y help model these flows.

2.4 Perfect Graphs from Trees

Let $T = (V, E)$ be a tree and let $\mathcal{F} = \{T_1, \ldots, T_k\}$ be a collection of connected trees, each of which is a subtree of T.

Construct a graph $G = (\mathcal{F}, E')$ where we say two distinct subtrees $T_i, T_j \in \mathcal{F}$ are adjacent in G if $V(T_i) \cap V(T_j) \neq \emptyset$ (i.e. they subtrees share a common vertex).

- Show the following, if $\mathcal{C} \subseteq \mathcal{F}$ is a clique in G (i.e. each pair of trees in $\mathcal{C}$ intersects at some node of T), then in fact there is some $v \in V$ such that $v \in \bigcap_{T_i \in \mathcal{C}} V(T_i)$. That is, there is a vertex in the underlying tree T that is shared by all subtrees in $\mathcal{C}$.

  Comment: Thus, we can find a maximum clique in G by scanning all vertices of T and seeing which one lies in the most subtrees in $\mathcal{F}$.

- Now show $\omega(G) = \chi(G)$ by showing how we can assign one of $\omega(G)$ colours to each $T_i \in \mathcal{G}$ such that no two that intersect receive the same colour.
- Conclude G is a perfect graph.

2.5 Orienting Graphs

The description of this one is a bit longer, but the actual work is not as bad as the description length makes it seem.

Background
Let $G = (V, E)$ be a connected, undirected graph. A classic result says it is possible to orient the edges of E to get a strongly connected, directed graph if and only if G has no bridges (edges whose deletion would disconnect the graph). To see the harder direction, consider a depth-first search tree: orient all tree edges away from the root of the search and orient all other edges "upward" (there are no cross edges in a depth-first search tree). You do not have to actually work out these details, this is just background information.

This question asks you to generalize this fact to higher connectivity.

The Problem
Say an undirected graph $G = (V, E)$ is k'-edge connected if it is a connected graph and deleting any set of $k' - 1$ edges leaves a connected graph. Equivalently, $|\delta(S)| \geq k'$ for every $\emptyset \subsetneq S \subsetneq V$. Say a directed graph $H = (V', A)$ is k-arc connected if $|\delta^{in}(S)| \geq k$ for every $\emptyset \subsetneq S \subsetneq V$. For example, 1-arc connected is the same as saying strongly connected.

You will show for any $k \geq 1$ that an undirected graph G has a k-arc connected orientation (i.e. some orientation of edges yields a k-arc connected digraph) if and only if G is $2k$-edge connected. One direction is clear, if such an orientation exists then there must be at least $2k$ edges crossing any cut of G (since there would be at least k in either direction in the orientation). Your task is to show the other half of this statement.

Start by picking an arbitrary orientation for each edge e, call the resulting set of arcs A. For each $a \in A$, let x_a be a variable indicating that we will reverse the orientation of a. That is, $x_a = 0$ means we do not change the orientation of a and $x_a = 1$ means we will flip the orientation of a.

For any $\emptyset \subsetneq U \subsetneq V$, we let $\delta^{in}(U)$ be the set of all arcs in A that enter U. Similarly, $\delta^{out}(U)$ is the set of all arcs in A exiting U (i.e. the directed cuts are with respect to the initial arc set A).

Your Task
Consider the following polytope:

$$\begin{array}{rcll} |\delta^{in}(U)| - x(\delta^{in}(U)) + x(\delta^{out}(U)) & \geq & k & \forall\, \emptyset \subsetneq U \subsetneq V \\ 0 \leq \quad x_a & \leq & 1 & \forall\, a \in A \end{array} \qquad (1)$$

- Argue that an integral point in the polytope corresponds to an orientation of G that yields a k-arc connected graph. Also show that setting $x_a = 1/2$ for each $a \in A$ yields a feasible solution.
- For the remaining questions, let x^* be an extreme point of the polytope. You will argue x^* is integral.

  For this part, argue that if U, W are sets corresponding to two tight constraints where both $U \cap W \neq \emptyset$ and $U \cup W \neq \emptyset$, then both $U \cap W$ and $U \cup W$ correspond to tight constraints.
- **Comment (nothing to submit for this part)**: Ultimately, using uncrossing arguments this implies there is a collection $\mathcal{F}$ of subsets of V such that $\{\chi_{\delta^{out}(U)} - \chi_{\delta^{in}(U)} : U \in \mathcal{F}\}$ constitutes a basis for the space spanned by all tight constraints of form (1). Further, while we cannot assume $\mathcal{F}$ is laminar (we could not necessarily uncross sets U, W if $U \cup W = V$),

we can assume that for any U, W with $U \cup W \neq V$ we have $U \subseteq W, W \subseteq U$ or $U \cap W = \emptyset$. The arguments are similar to what we saw in the lectures and modifying them to get such a set $\mathcal{F}$ is quite straightforward, so you do not have to do it.

- Demonstrate the matrix consisting of rows $\chi_{\delta^{out}(U)} - \chi_{\delta^{in}(U)}$ for the sets $U \in \mathcal{F}$ is a network matrix (or the transpose of a network matrix).

  Hint: Pick any $r \in V$ and consider the vectors $\{U \subseteq V - \{r\} : U \in \mathcal{F}\} \cup \{U \subseteq V - \{r\} : V - U \in \mathcal{F}\}$. Argue this is a laminar family and complete the construction of the network matrix.

- Conclude that x^* must be integral, thus there is an orientation of edges of G to get a k-arc connected graph.