Spring 2022, CS 586/IE 519: Combinatorial Optimization
Homework 3
Due: Thursday, March 10, 2022

Instructions and Policy: Each person should write up their own solutions independently. You need to indicate the names of the people you discussed a problem with. Solutions to most of these problems can be found from one source or the other. Try to solve on your own first, and cite your sources if you do use them.

Please write clearly and concisely. Refer to known facts. You should try to convince us that you know the solution, as quickly as possible.

Submit solutions to at least four problems.

Problem 1 Flow is typically defined as a function on the edges but path-based flow definition and formulations are useful and necessary in various applications. We illustrate some relevant issues via this problem. Let $D = (V, A)$ be a directed graph with non-negative capacities $c : A \to R^+$. Given nodes s, t let $P_{s,t}$ denote the set of all simple paths between s and t. See the notes for discussion on the first few parts which are mainly for background.

- **Not to submit** Write the maximum s-t flow problem as a linear programming problem with one variable for each path $p \in P_{s,t}$. Note that the primal can have an exponential (in $|V|$) number of variables. Write its dual.
- **Not to submit** What is the separation problem for the dual? Show that there is a polynomial time algorithm for the separation problem for the dual. Via the Ellipsoid method, this implies that you can solve the dual to optimality.
- **Not to submit** Suppose you have an optimum solution to the dual. Show via complementary slackness that you can restrict attention to only a polynomial number of paths in the primal and hence find an optimum solution to the primal by solving a linear program of size polynomial in the input.
- Via flow-decomposition observe that if the capacities are integral then the primal has an optimum integral solution. Does this imply that the polyhedron defined by the constraints is integral? If not, explain.
- Now consider the following problem. You want to find the maximum s-t flow when restricted to paths of length at most k where k is a given integer. Write this is a linear program with an exponential number of variables. Show that the separation oracle for the dual is polynomial time solvable.

Problem 2 This problem is regarding the notion of *element connectivity* which is useful in bridging edge and vertex connectivity. Let $G = (V, E)$ be a graph and let V be partitioned into two

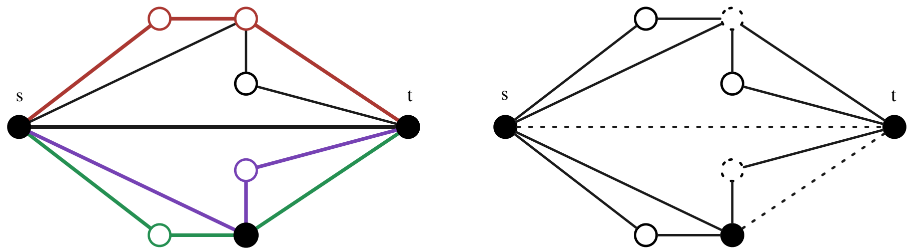

Figure 1: The black vertices are the terminals. The left image shows 4 element-disjoint st-paths. The right image shows removing 4 elements disconnects s and t. $\kappa'(s,t) = 4$.

sets B and W where B is the set of terminals and W is the set of non-terminals. The elements are $E \cup W$. For any two distinct terminals s, t the element-connectivity between s and t is the maximum number of elment-disjoint paths between s and t. Note that the paths need not be disjoint in terminals. We denote the element connectivity between s and t by $\kappa'_G(s,t)$. An alternative definition is via cuts: $\kappa'_G(s,t)$ is the minimum number of elements whose removal disconnects s from t. See figure. Note that κ' is defined only between the terminals.

- Given $G = (V, E)$ and $s, t \in B$ describe an efficient algorithm to compute the $\kappa'_G(s,t)$. Note that $\kappa'(s,t) = \kappa'(t,s)$.
- Given three terminals a, b, c prove that $\kappa'(a,b) \geq \min\{\kappa'(a,c), \kappa'(b,c)\}$.
- **Extra credit:** It suffices to prove the following. Let $f : 2^B \to \mathbb{Z}_+$ be a symmetric function where $f(S)$ is defined as the minimum element cut between S and $B - S$ in G. Prove that f is submodular. And conclude that there is a Gomory-Hu tree for κ' over the terminals.

Problem 3 Let $G = (V, E)$ be an undirected graph. Consider the following LP formulation for min-cost perfect matching.

$$\sum_{e \in E} w_e x_e$$
$$\sum_{e \in \delta(v)} x_e = 1 \quad v \in V$$
$$x_e \geq 0 \quad e \in E$$

The above LP is an integral polytope when G is bipartite but for non-bipartite graphs it is not necessarily so. Nevertheless it is useful to understand the properties of this LP with regards to matchings.

- Prove that the if y is an extreme point of the polytope then y is half-integral: in other words $y_e \in \{0, 1/2, 1\}$ for each edge e. In fact, prove that the support of y (those edges e with $y(e) > 0$) consists of a collection of vertex-disjoint edges and odd cycles.

- Using the preceding prove that if y^* is an optimum solution to the LP then there is a matching of size at least $|V|/3$ whose cost is at most $\sum_e w_e y_e^*$.

Problem 4 In a graph $G = (V, E)$ a set of edges $S \subseteq E$ is an edge cover if each vertex v is incident to at least one edge in S. The goal in the minimum edge cover problem is to find a an edge cover of smallest cardinality. If edges have non-negative weights we obtain the minimum weight edge cover problem. Note that in edge cover we seek to cover vertices by edges while in the NP-Hard vertex cover problem we seek to cover edges by vertices. The unweighted and weighted versions of edge cover can be solved via matching techniques. See the notes.

Prove that the following inequality system characterizes the edge cover polytope (the convex hull of the characteristic vectors of the edge covers of a given graph G).

$$\begin{array}{rcl} x\left(E[U] \cup \delta(U)\right) \geq & \frac{|U|+1}{2} & U \subseteq V; |U|\, odd \\ 0 \leq x(e) \leq 1; & & e \in E \end{array}$$

Problem 5 Petersen's theorem states that every bridgeless (no cut-edge) cubic graph has a perfect matching.

- Deduce this theorem from the Tutte-Berge formula.
- Prove this theorem by exhibiting a fractional solution that is feasible for the perfect matching polytope.
- **Extra Credit:** Show that every bridgeless cubic-graph has a T-join of size at most $|V|/2$ for any even $T \subseteq V$.

Problem 6 Let $G = (V, E)$ be an undirected graph with non-negative edge lengths $\ell : E \to R^+$. Given nodes s, t we wish to find a shortest s-t path with an odd number of edges. Show that this problem can be solved via matching techniques by following the hint below. Suppose s, t have degree 1. Consider the reduction of the maximum weight matching problem to the maximum weight perfect matching problem. Adapt this reduction and show how a perfect matching can be used to obtain a shortest odd length path from s to t. How can you get rid of the assumption that s, t have degree 1? Extend the ideas for finding a shortest even length s-t path.