Final exam 14/01/25

Many of the questions build on the previous steps. You're allowed to assume a previous step even if you haven't been able to prove it.

Problem 1. The **distance** between two vertices (extreme points) u and v of a polytope P is defined as the shortest sequence of consecutive edges connecting u and v. The **diameter** of P is the maximum distance between two vertices of P. Prove that the diameter of the perfect matching polytope of a graph G is at most $|V(G)|/4$.

Hint: In the homework, we proved that the perfect matching polytope has an edge between two matchings M and N if and only if $M \Delta N$ is an even cycle.

Problem 2. A *cut-edge* in a graph $G = (V, E)$ is an edge $e \in E$ such that $G - e$ has one more connected component than G. Let G be a 3-regular graph that has no cut-edge. In this problem, you will use Tutte's theorem to prove that G has a perfect matching.

(a) Consider a subset $U \subseteq V$ and any odd-sized component H of $G - U$. Prove that there must be at least 3 edges between H and U in G.

(b) Use (a) to prove that the conditions of Tutte's theorem are satisfied. Conclude that G has a perfect matching.

Problem 3. Let $G = (V, E)$ be an undirected graph. Consider the polytope

$$Q_f(G) = \{x \in \mathbb{R}^E_{\geq 0} : \forall v \in V, \sum_{e \in \delta(v)} x_e \leq 1\}.$$

In the course we called this polytope $Q_{bi}(G)$ and proved that it is the matching polytope of G iff G is bipartite. For an arbitrary graph, $Q_f(G)$ is called the **fractional matching polytope**. In this problem, you will prove by induction on $|E|$ that $Q_f(G)$ is half-integral, i.e., that every extreme point $x \in Q_f(G)$ satisfies $x \in \{0, 1/2, 1\}^E$. The statement holds trivially if $|E| = 0$, so assume that $|E| > 0$ and let $x \in Q_f(G)$ be an extreme point.

(a) Suppose that $x_e = 0$ for some $e \in E$. Show how to apply the induction hypothesis.

(b) Suppose that $x_e = 1$ for some $e \in E$. Show how to apply the induction hypothesis.

From here on, assume that $0 < x_e < 1$ for every $e \in E$.

(c) Show that every vertex in G must have degree 0 or 2, and that $\sum_{e \in \delta(v)} x_e = 1$ for every degree-2 vertex v. (*Hint: use the right definition of an extreme point to show that* $|E| \leq |v \in V : \deg(v) \geq 2|$.)

(d) Deduce from (c) that $x_e = 1/2$ for every $e \in E$.

Problem 4. In the **set cover problem**, we are given as input a ground set of elements $E = \{e_1, \ldots, e_n\}$, some subsets $S_1, S_2, \ldots, S_m \subseteq E$, and a nonnegative weight $w_j \geq 0$ for each S_j. The goal is to find $I \subseteq \{1, \ldots, m\}$ that minimises $\sum_{j \in I} w_j$ subject to $\bigcup_{j \in I} S_j = E$.

Consider the following integer programme formulation of the set cover problem.

$$\begin{aligned} \min \quad & \sum_{j=1}^{m} w_j x_j & & \\ \text{subject to} \quad & \sum_{j : e_i \in S_j} x_j \geq 1 & & i = 1, \ldots, n \\ & x_j \in \{0, 1\} & & j = 1, \ldots, m \end{aligned} \tag{1}$$

(a) Relax the constraints $x_j \in \{0,1\}$ to $x_j \geq 0$ in (1) and write down the dual LP with dual variables y_i for $i = 1, \ldots, n$.

(b) Write down the corresponding complementary slackness conditions.

(c) We will analyse a primal-dual algorithm in which we start with an empty set I (infeasible primal) and all dual variables equal to 0 (feasible dual). In each iteration, we pick an unsatisfied primal constraint and increase its corresponding dual variable until some dual constraint becomes tight. Let j be the index corresponding to the tight dual constraint and add j to I. **Write out this algorithm in detail.**

(d) Show that after every iteration of the algorithm, we have

$$\sum_{j \in I} w_j = \sum_{i=1}^{n} y_i \cdot |\{j \in I : e_i \in S_j\}|. \tag{2}$$

(e) Let $f = \max_i |\{j \in I : e_i \in S_j\}|$. Use (2) to show that the algorithm finds an f-approximation of the optimum of (1).

Remark: Since f grows as n in the worst case, this is not always a good approximation. However, it can be useful in special cases where f is bounded.

(f) Show that the analysis of this algorithm is tight.