Technische Universität München, Zentrum Mathematik
Lehrstuhl für Operations Research

Advanced Graph Algorithms (MA5227)

Dr. Jannik Matuschke | M.Sc. Marinus Gottschau

Problem Set 5

Homework Exercise 5.1 (Matchings & Stable Sets & Vertex-Covers & Edge-Covers)
Given a graph $G = (V, E)$, let $\alpha(G)$ denote the size of a maximum stable set, $\tau(G)$ the minimum cardinality of a vertex cover, $\nu(G)$ the maximum cardinality of a matching, and $\xi(G)$ the minimum cardinality of an edge cover in G. Prove that:

a) $\alpha(G) + \tau(G) = |V|$

b) $\nu(G) + \xi(G) = |V|$ if G does not contain isolated vertices.

c) $\xi(G) = \alpha(G)$ if G is a bipartite graph without isolated vertices.

Homework Exercise 5.2 (Carefully Shrink Blossoms)
Let G be a graph, M a matching in G and C be a blossom in G (with respect to M).
Let G' and M' result from G and M by shrinking $V(C)$ to a single vertex. Construct an example, where M' is a maximum matching in G' but M is not a maximum matching in G.

Homework Exercise 5.3 (MinimumWeightMatchingProblem and Shortest Even/Odd s-t-Paths)
Given a graph $G = (V, E)$ with weights $w : E \to \mathbb{R}_+$ and two vertices s and t.
Finding a perfect matching M of minimum weight, i.e. a perfect matching that minimizes $\sum_{e \in M} w(e)$, or deciding that there is no perfect matching in the graph, is called the MinimumWeightPerfectMatchingProblem.

a) In the lecture you saw an algorithm that solves the MinimumCostFlowProblem. Use this algorithm to solve the MinimumWeightPerfectMatchingProblem on a bipartite graph $G' = (A \dot{\cup} B, E)$ and $w' : E \to \mathbb{R}_+$.

b) We now look for a shortest s-t-path in G. Use an algorithm that solves the MinimumWeightPerfectMatchingProblem on bipartite graphs to find a shortest s-t-path.
Hint: Consider two copies of V.

c) Suppose you have an algorithm that solves the general MinimumWeightPerfectMatchingProblem efficiently. How do you find the shortest s-t-path with an even (odd) number of edges using that algorithm?
Hint: Consider two copies of G.

Tutorial Exercise 5.4 (Unmatchable Edges)
Let $G = (V, E)$ be a graph. We call an edge $e = \{u, v\} \in E$ unmatchable, if no perfect matching in G contains e. Give an $\mathcal{O}(|V|^3)$ time algorithm that finds all unmatchable edges.

This problem set will be discussed in the tutorials on December 18th & 19th, 2017.