Communications and Transport Systems
Department of Science and Technology **Fall 2019**
Linköping University

Christiane Schmidt

Design and Analysis of Algorithms Part 1 - Mathematical tools and Network problems homework 6, 10.02.2020

Problem 1 (Maximum matching in bipartite graphs):

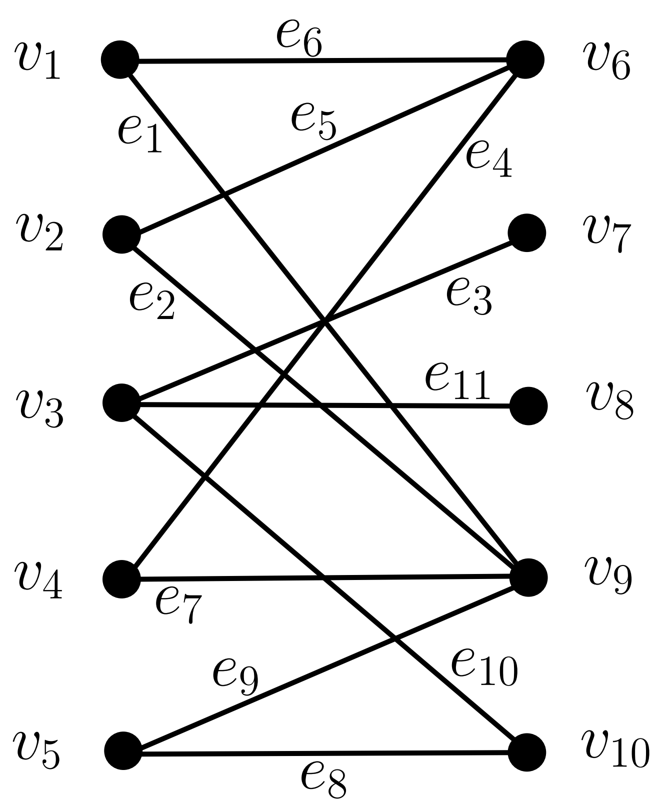

Figure 1: A graph.

Use the flow formulation from the lecture to determine a maximal matching in the graph G from Figure 1. Use your preferred flow algorithm.

Problem 2 (Matching and Vertex Cover):
In bipartite graphs we have $\nu(G) = \tau(G)$ (see seminar notes). In general: $\nu(G) \leq \tau(G)$.

(a) Give a graph with $\nu(G) < \tau(G)$, more precisely $\tau(G) = 2 \cdot \nu(G)$.

(b) Give a graph class with $\nu(G) < \tau(G)$, more precisely $\tau(G) = 2 \cdot \nu(G)$.

Problem 3 ((Inclusion-wise) maximal matchings):

A matching M_0 in a graph G is called *(inclusion-wise) maximal*, if there is no matching M in G with $M_0 \subset M$. Let G be a graph and M_1, M_2 two (inclusion-wise) maximal matchings in G. Show that $|M_1| \leq 2|M_2|$ gilt.
(Hint: Why do the vertices of the matching edges from M_1 and M_2 each constitute a vertex cover? Moreover, we showed that every matching is smaller every vertex cover.)

Problem 4 (Perfect matching in bipartite graphs):
A perfect matching $M \subseteq E$ is a set of pairwise nonadjacent edges, where there is *exactly one* edge incident to each vertex. Show that in a bipartite graph $G = (V, E)$ with $V = V_1 + V_2$ in which each vertex has exactly degree $k \geq 1$, there is a perfect matching. Use the theorem by Hall.

Problem 5 (Blossom Algorithm I.):

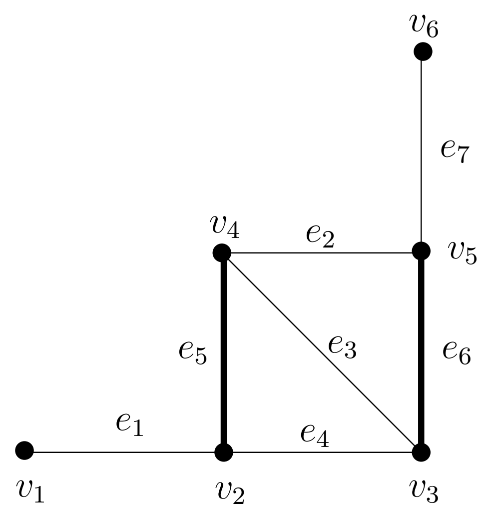

Figure 2: Graph G.

(a) Is the graph G from Figure 2 bipartite? Justify your claim.

(b) Given the graph G from Figure 2 and the matching $M = \{e_5, e_6\}$.

With the help of the blossom algorithm from the lecture decide whether G has a perfect matching or not. Startwith the matching M. After each

- *Augmentation* give the new matching

· *Tree-extension operation* give the new tree

· *Shrinking* give the new tree and the graph G'

Always choose the unmatched vertex with smallest index as starting vertex for your tree. If there is more than one edge to choose from in step 3 of the blossom algorithm, choose the edge with smallest edge index.

Problem 6 (Blossom Algorithm II.):

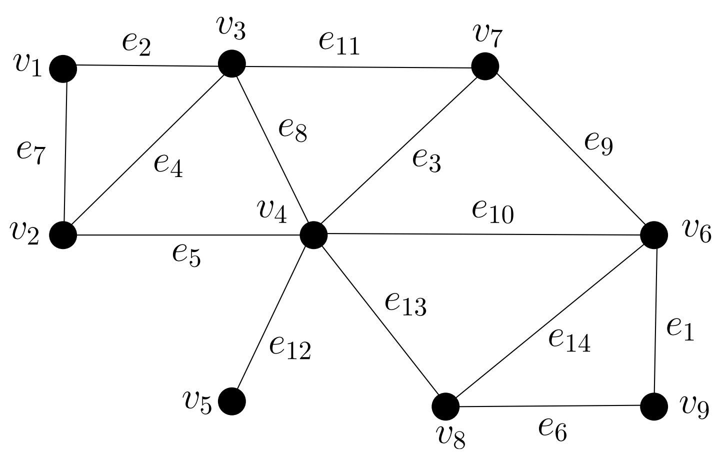

Figure 3: Graph H.

Use the blossom algorithm from the lecture to decide whether the graph H from Figure 3 has a perfect matching or not.
Always choose the unmatched vertex with smallest index as starting vertex for your tree. If there is more than one edge to choose from in step 3 of the blossom algorithm, choose the edge with smallest edge index. Consider the constructed tree when the algorithm stops. Delete the black vertices from G and justify that a perfect matching exists.

Problem 7 (Perfect Matching):

Use the theorem from Tutte to show whether the graph H' from Figure 4 has a perfect matching.

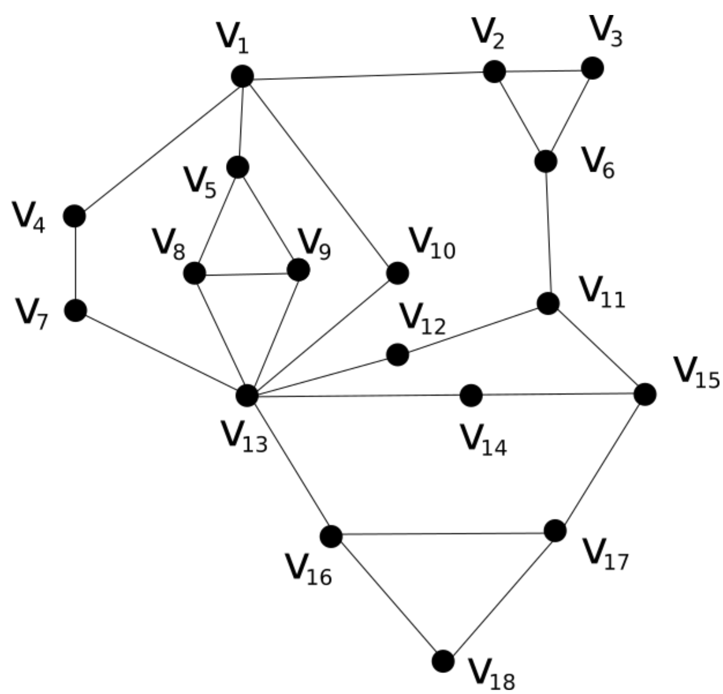

Figure 4: Graph H'.