Communications and Transport Systems
Department of Science and Technology **Spring 2017**
Linköping University

Valentin Polishchuk
Christiane Schmidt

Design and Analysis of Algorithms Part 1 - Mathematical tools and Network problems homework 4, 07.06.2017

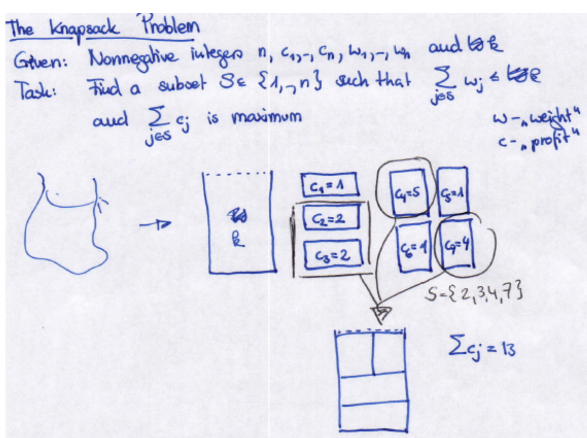
Problem 1 (Distributed algorithms):
Assume all vertices have an ID, but they can only apply the operations = and ≠to these IDs. Is it possible to solve leader election in this model? (Hint: give a reason for your answer, not just no/yes.)

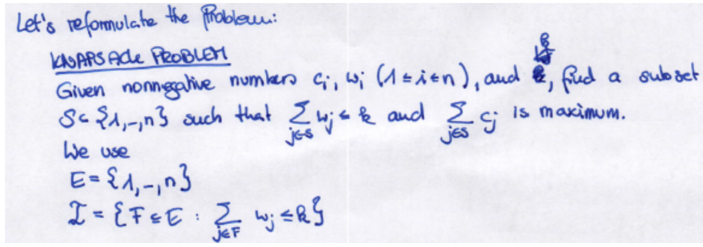
Problem 2 (Independence Systems): Let $E = \{1, \dots, 10\}$ and
$\mathcal{I}_1 = \{\{1,2,3,4\},\{2,3,4,5\},\{3,4,5,6\},\{4,5,6,7\},\{5,6,7,8\},\{6,7,8,9\},$
$\{7,8,9,10\},\{1,2,3\},\{2,3,4\},\{3,4,5\},\{4,5,6\},\{5,6,7\},\{6,7,8,\},$
$\{7,8,9\},\{8,9,10\},\{1,2\},\{2,3\},\{3,4\},\{4,5\},\{5,6\},\{6,7\},\{7,8\},$
$\{8,9\},\{9,10\},\{1\},\{2\},\{3\},\{4\},\{5\},\{6\},\{7\},\{8\},\{9\},\{10\},\varnothing\}$
and
$\mathcal{I}_2 = \{\{1,2,3\},\{6,7,9\},\{1,2\},\{1,3\},\{2,3\},\{6,7\},\{6,9\},\{7,9\},\{1\},\{2\},\{3\},\{6\},\{7\},\{9\}\}$

a) Is $(E, \mathcal{I}_1)$ an independence system?

b) Is $(E, \mathcal{I}_2)$ an independence system?

Problem 3 (Independence Systems II):




Let's reformulate the Problem:

KNAPSACK PROBLEM

Given nonnegative numbers c_i, w_i $(1 \le i \le n)$, and k, find a subset $S \subseteq \{1,\dots,n\}$ such that $\sum_{j \in S} w_j \le k$ and $\sum_{j \in S} c_j$ is maximum.

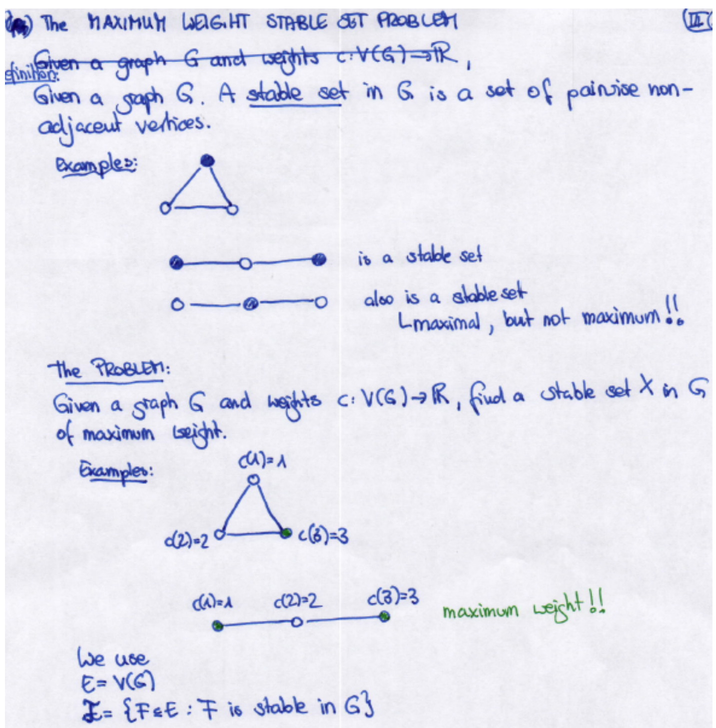
We use

$E = \{1,\dots,n\}$

$\mathcal{I} = \{F \subseteq E : \sum_{j \in F} w_j \le k\}$

Is $(E, \mathcal{I})$ an independence system?

Problem 4 (Independence Systems II):


The MAXIMUM WEIGHT STABLE SET PROBLEM

~~Given a graph G and weights $c: V(G) \to \mathbb{R}$~~,

Definition: Given a graph G. A stable set in G is a set of pairwise non-adjacent vertices.

Examples:

is a stable set

also is a stable set

Lmaximal, but not maximum!!

The PROBLEM:

Given a graph G and weights $c: V(G) \to \mathbb{R}$, find a stable set X in G of maximum weight.

Examples:

$c(1)=1$

$c(2)=2$

$c(3)=3$

$c(1)=1$ $c(2)=2$ $c(3)=3$

maximum weight!!

We use

$E = V(G)$

$\mathcal{I} = \{F \subseteq E : F \text{ is stable in } G\}$

Is $(E, \mathcal{I})$ an independence system?

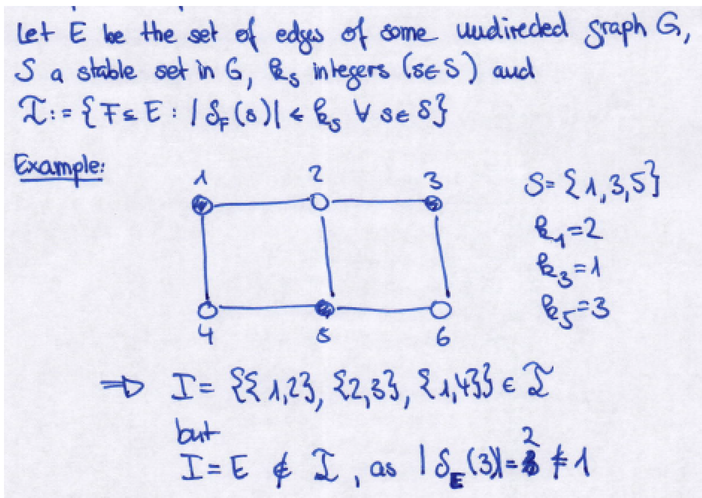
Problem 5 (Matroids): Let $E_2 = \{1, \dots, 7\}$ and
$\mathcal{I}_3 = \{\{1,2,3\},\{1,2,5\},\{1,2,6\},\{1,2,7\},\{1,3,4\},\{1,3,6\},\{1,3,7\},\{1,4,5\},\{1,4,6\},$
$\{1,4,7\},\{1,5,6\},\{1,5,7\},\{2,3,4\},\{2,3,5\},\{2,3,7\},\{2,4,5\},\{2,4,6\},\{2,4,7\},\{2,5,6\},$

$\{3,4,5\}, \{3,4,6\}, \{3,5,6\}, \{3,5,7\}, \{4,5,7\}, \{5,6,7\}, \{1,2\}, \{1,3\}, \{1,4\}, \{1,5\}, \{1,6\}, \{1,7\}, \{2,3\}, \{2,4\}, \{2,5\}, \{2,6\}, \{2,7\}, \{3,4\}, \{3,5\}, \{3,6\}, \{3,7\}, \{4,5\}, \{4,6\}, \{4,7\}, \{5,6\}, \{5,7\}, \{6,7\}, \{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}, \{7\}, \varnothing\}$

a) Is $(E_2, \mathcal{I}_3)$ an independence system?

b) Is $(E_2, \mathcal{I}_3)$ a matroid?

Problem 6 (Matroids II):



Let E be the set of edges of some undirected graph G,
S a stable set in G, k_s integers $(s \in S)$ and
$\mathcal{I} := \{F \subseteq E : |\delta_F(s)| \leq k_s \ \forall s \in S\}$

Example:

$S = \{1,3,5\}$

$k_1 = 2$

$k_3 = 1$

$k_5 = 3$

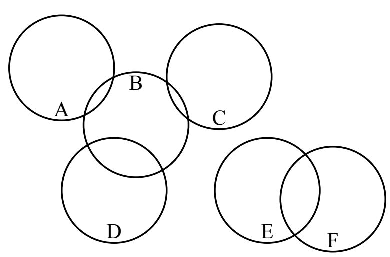
$\Rightarrow I = \{\{1,2\}, \{2,3\}, \{1,4\}\} \in \mathcal{I}$

but

$I = E \notin \mathcal{I}$, as $|\delta_E(3)| = 2 \neq 1$

Is $(E, \mathcal{I})$ a matroid?

Problem 7 (IS and matroids): Consider the following system: We are given a ground set, consisting of circles with uniform radius in the plane. For an example:



We say that a selection of some of these circles is *independent*, iff no two of them intersect. For example, $\{C, D\}$ is independent, but $\{E, F\}$ is dependent.

a) Prove that this system is an independence system for any given ground set of circles.

b) Find a nonempty example of circles for which the system is a matroid (and prove it).

c) Find an example of circles for which the system fulfills these criteria:

- It is **not** a matroid (prove it).
- All bases have the same size k, with $k \geq 3$.

Problem 8 (Algorithm 8.7, Ford-Fulkerson):

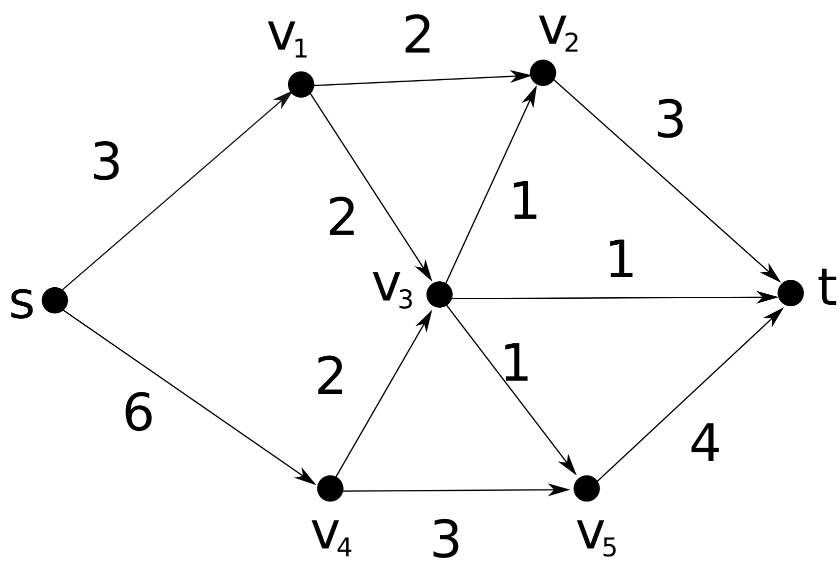


Figure 1: The network (G, u, s, t). The numbers at the edges give the capacities

Use the algorithm by Ford and Fulkerson to determine a maximum $s-t$-flow in the network (G, u, s, t). Give the residual graph in each step.
In addition: give a minimum cut.

Problem 9 (Menger's Theorem (Menger 1927)):
Two paths P and Q are called edge-disjoint if they have no common edge.
Let G be a graph (directed or undirected), let s and t be two vertices and $k \in \mathbb{N}$. Then there are k edge-disjoint s-t-paths if and only if after deleting $k-1$ edges t is still reachable from s.

Problem 10 (Ford-Fulkerson algorithm and irrational capacities):
Show that the algorithm by Ford and Fulkerson might not terminate when it is applied to a network with irrational capacities.

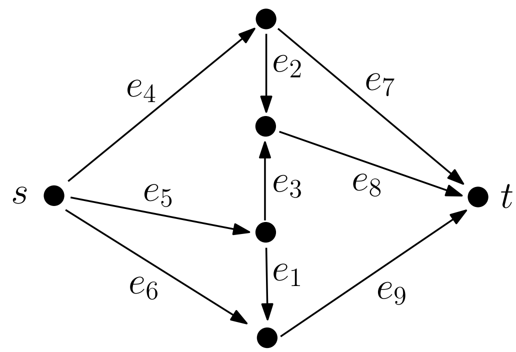


Figure 2: A network with irrational capacities

Consider the network in Figure 2 with capacities $u(e_1) = 1$, $u(e_2) = \sigma$, $u(e_3) = 1$ und $u(e_4) = u(e_5) = \ldots = u(e_9) = 4$, with $\sigma = \frac{\sqrt{5}-1}{2}$. First show $\sigma^n = \sigma^{n+1} + \sigma^{n+2}$.
(Hint: Consider the paths $P_1 = \{e_4, e_2, \overleftarrow{e_3}, e_1, e_9\}$, $P_2 = \{e_5, e_3, \overleftarrow{e_2}, e_7\}$, $P_3 = \{e_6, \overleftarrow{e_1}, e_3, e_8\}$ und $P_4 = \{e_5, e_3, e_8\}$. Show by induction that we can change the *residual capacities* of e_1, e_2 and e_3 from σ^n, σ^{n+1} and 0 to σ^{n+2}, σ^{n+3} and 0, respectively. Induction base: augment along P_4. Induction step: augment, consecutively, along P_1, P_2, P_1 and P_3.)

Problem 11 (Integer Flow): Show Corrolary 8.12 from the seminar: Let $N = (G, u, s, t)$ be a network. If the capacities $u(e)$ are all integers, then there exists a maximum flow in N, such that all $f(e)$ are integeres (in particular, the optimum flow is integer).

Problem 12 (PUSH-RELABEL algorithm): For each proof you can use all theorems, lemmata etc. with a smaller number.

(a) Show Proposition 8.20: During the execution of the Push-Relabel algorithm f is always an s-t-preflow and ψ is always a distance labeling with respect to f. (Hint: Show that the procedures PUSH and PRELABEL preserve these properties.)

(b) Show Lemma 8.21: If f is an s-t-preflow and ψ is a distance labeling with respect to f, then

  (1) s is reachable from any active vertex v in G_f.

  (2) t is not reachable from s in G_f.

  (Hint: For (1) consider the set of vertices that are reachable from an active vertex v. For (2) use contradiciton.)

(c) Show Theorem 8.22: When the algorithm 8.19 terminates, f is a maximum s-t-flow.

(d) Show Lemma 8.24: The number of saturating pushes is at most mn.

Problem 13 (MIN CUT problem): The MIN CUT problem is defined as follows:
INPUT: Network (G, u, s, t).
OUTPUT: An s-t-cut of minimum capacity.

Show how you can compute a MIN CUT in time $O(n^3)$.

Problem 14 (Maximum matching in bipartite graphs):

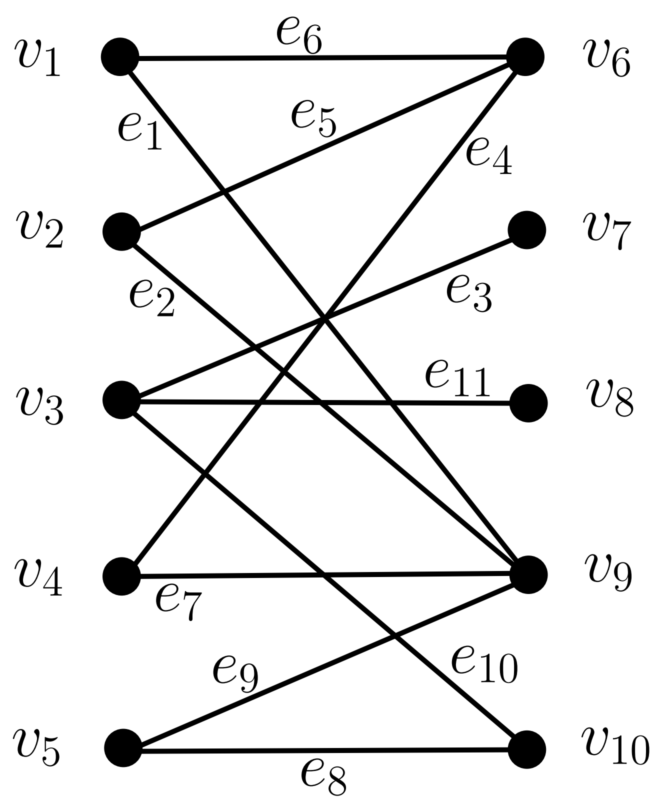


Figure 3: A graph.

Use the flow formulation from the lecture to determine a maximal matching in the graph G from Figure 3. Use your preferred flow algorithm.

Problem 15 (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 16 ((Inclusion-wise) maximal matchings):

A matching M_0 in a graph G is called *(nclusion-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 17 (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.