Freie Universität Berlin
FB Mathematik und Informatik
WS 2017/18

Prof. Dr. Ralf Borndörfer
Isabel Beckenbach

Optimization 2

Exercise Sheet 13

Submission: None

Exam Preparation
This is an example how the final exam can look like. You do not have to hand in your solutions. Instead we discuss them on February 6th.

Exercise 13.1 — 10 Points

Decide for each of the following statements whether it is true or false, give a proof or counterexample each time.

(a) Let $G = (V, E)$ be a connected graph with costs $c \in \mathbb{R}^E$ on the edges. The set of all edges contained in some minimum cost spanning tree can be found in polynomial time.

(b) Use the KKT-conditions to solve

$$\begin{aligned} \min \frac{3}{2}x_1^2 + x_1x_2 + \frac{3}{2}x_2^2 \\ x_1 + x_2 = 4 \\ x_1 \geq 3 \end{aligned}$$

This quadratic program has a unique solution.

(c) If a quadratic program has a finite solution, then it has a unique solution.

(d) Let $G = (V, E)$ be a connected graph. The independence system $\mathcal{M} = (E, \{F \subseteq E : \text{ there exists a hamiltonian cycle containing } F\})$ is a matroid.

(e) The active set method always stops after a finite number of iterations.

Exercise 13.2 — 10 Points

Let $G = (V, E)$ be a connected graph, $c \in \mathbb{R}^E$, and $v^* \in V$ a fixed vertex of G. Is it possible to find a minimum cost spanning tree in polynomial time such that

(a) v^* is a leaf,

(b) v^* is not a leaf in the spanning tree.

In both cases, either show that the problem is $\mathcal{NP}$-hard or give a polynomial time algorithm for this problem.

Exercise 13.3 **10 Points**

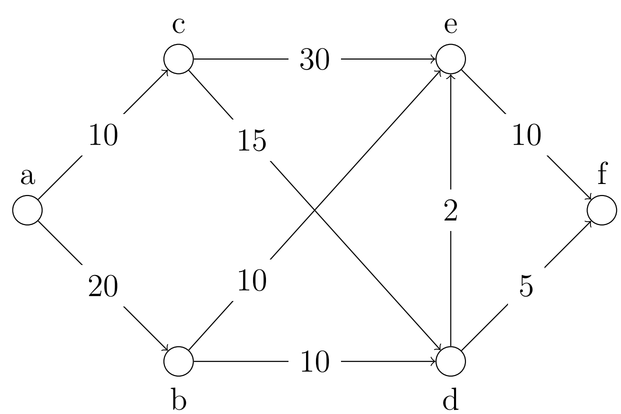
Suppose we run the Floyd-Warshall Algorithm on a digraph $D = (V, A)$ with general weights $c \in \mathbb{R}^A$. Adapt the algorithm such that it either outputs "D contains a cycle C of negative weight" or the all pair shortest path length matrix $(dist_{i,j})_{1 \leq i,j \leq n}$.

Exercise 13.4 **10 Points**

Let $D = (V, A)$ be a directed graph, with capacities $u \in \mathbb{R}^A_{\geq 0}$, costs $c \in \mathbb{R}^A_{\geq 0}$, and $I = \{1, \ldots, k\}$ a set of k commodities. For each commodity a source s_i, a sink t_i, and a demand d_i is given ($i = 1, \ldots, k$). The task is to find an integral (s_i, t_i) flow of value d_i for each commodity i such that the sum of flows on an arc a is at most its capacity $u(a)$ and the sum of the arc costs of all flows is minimal.

Exercise 13.5 **10 Points**

Run Dijkstra's Algorithm on the following graph starting at vertex a.



Exercise 13.6 **10 Points**

Given a set of jobs $\{1, \ldots, n\}$ where each jobs can be processed in one time unit. All jobs are known at time 0, and each job i has a deadline $d_i \geq 0$ and a profit $c_i \geq 0$. The profit for job i is earned if and only if job i is completely finished before its deadline d_i. Jobs cannot be processed in parallel, and once a job is started it has to be finished before a new job can start. Show that there exists a greedy algorithm solving this problem.

Hint: Show that if a set of jobs can be processed before their deadlines, then one can also process the jobs in the order of their deadlines.