Freie Universität Berlin
FB Mathematik und Informatik
SS 2018

Prof. Dr. Ralf Borndörfer
Pedro Maristany

Optimization III

Exercise Sheet 1

Submission: Wednesday, 25.04.2018, 12:00

Exercises:
Rigorous mathematical proofs/arguments are expected if not stated otherwise. You are allowed to work in groups of two. Exercises labeled with '*' are very difficult but their solution can be found after some literature research. They can be used for interesting discussions during the exercise sessions.

Homepage of the Lecture: `http://www.zib.de/ws17_Optimierung_II`
Questions?: `maristany@zib.de`

Exercise 1.1 — 10 Points

We introduce the *Shortest Path Problem with Piecewise Constant Crossing Costs*. The first part of the input is familiar: a digraph $D := (V, A)$, two nodes $s, t \in V$, and a cost function $c : A \to \mathbb{R}_+$. In addition, we are given a distance function $d : A \to \mathbb{R}_+$, a set $R \subseteq A$ of arcs, and a resource consumption function

$$r : A \to \mathbb{R}_+$$
$$a \mapsto \begin{cases} d_a, & a \in R \\ 0, & \text{else.} \end{cases}$$

For $k \in \mathbb{N}$ given resource consumption limits R_i with $R_i < R_j$ for $i < j$ and prices F_i ($F_i < F_j$ for $i < j$), $i, j = 0, 1, .., k$ (assume $R_k = \infty$), the crossing costs function f is defined as follows:

$$f : \mathbb{R}_+ \to \mathbb{R}_+$$
$$x \mapsto \begin{cases} F_0, & x \in [0, R_0] \\ F_1, & x \in (R_0, R_1] \\ \dots & \dots \\ F_k, & x \in (R_{k-1}, R_k) \end{cases}$$

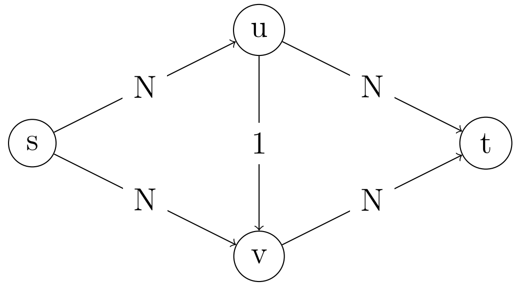

Figure 1: Example network with capacities on arcs. $N \in \mathbb{N}$.

The Shortest Path Problem with Piecewise Constant Crossing Costs is to find an (s,t)-path P in D that minimizes

$$\sum_{a \in P} c(a) + f(\sum_{a \in P} r(a)).$$

Model the Shortest Path Problem with Piecewise Constant Crossing Costs as an Integer Program.

Exercise 1.2 **6 Points**

Consider the network G shown in Figure 1. Assume that only integer flow units can be sent through the network.

a) What is the value of a maximum flow from s to t in G?

b) If the algorithm of Ford and Fulkerson is used to compute a maximum flow from s to t in G, how many iterations would the algorithm need to terminate in a worst-case scenario? Please describe this scenario.

c) How many iterations would Edmond-Karps algorithm need to find a maximum flow from s to t in G?

Exercise 1.3 **10 Points**

Consider the following optimization problem:

INSTANCE: Directed graph $D := (V, A)$, vertices $s, t \in V$, a multiplier $h(v) \in \mathbb{Z}_+$ for each $v \in V \backslash \{s,t\}$, capacities $c(a) \in \mathbb{Z}_+$ for each $a \in A$, and a flow requirement $R \in \mathbb{Z}_+$.

QUESTION: Is there a flow function $f : A \to \mathbb{Z}_+$ such that

a) $f(a) \leq c(a)$ for all $a \in A$,

b) for each $v \in V \backslash \{s,t\}$ there holds

$$\sum_{a \in \delta^-(v)} h(v) \cdot f(a) = \sum_{a \in \delta^+(v)} f(a)$$

c) the net flow into t is at least R.

a) Construct a YES-instance of this problem.

b) *Prove that this problem is NP-complete (hint: you can use the PARTITION problem for the transformation.)

Exercise 1.4 **10 Points**

Prove the following statement: Given a digraph $D := (V, A)$, and lower and upper capacities $l, u : A \to \mathbb{R}_+$ with $l(a) \leq u(a)$ for all $a \in A$, there is a circulation f with $f(a) \in [l(a), u(a)]$ for all $a \in A$ if and only if

$$\sum_{a \in \delta^-(X)} l(a) \leq \sum_{a \in \delta^+(X)} u(a), \quad \forall X \subseteq V.$$