University of Konstanz
Department of Computer & Information Science
Sabine Cornelsen / Jan Christoph Athenstädt

Combinatorial Optimization
Summer 2014

Assignment 5

Post Date: 23 May 2014 **Due Date:** 30 May 2014 **Tutorial:** 4 June 2014
You are permitted and encouraged to work in groups of two.

Problem 1: Dating Problem — 5 Points

An online dating-service has received personal information from a set V of men and a set W of women. For each man $v_i \in V$ and each woman $w_j \in W$, a compatibility score $s_{i,j} > 0$ is calculated. The task is to arrange dates such that the total compatibility of the partners is maximized.

Translate the task into a min-cost flow problem.

Problem 2: Duality of Min-Cost Flows — 7 Points

Let B be the incidence matrix of a directed graph $D = (V, E)$. The linear program of the uncapacitated min-cost flow program is the following:

$$\begin{array}{lll} P: \text{ minimize} & c^T f & \\ \text{subject to} & Bf & = \quad b \\ & f & \geq \quad 0 \end{array}$$

(a) What does the dual problem look like?

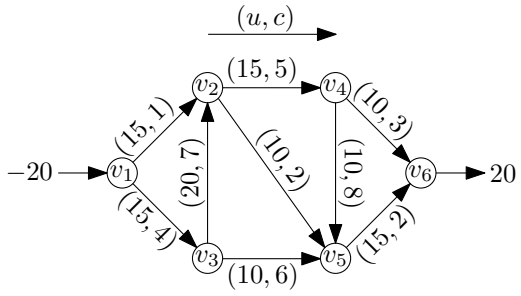
(b) Relate the dual to the problem of maximizing $\sum_{v \in V} b(v)\pi(v)$ over all vertex potentials $\pi : V \longrightarrow \mathbb{R}$ for which $c_\pi(e) \geq 0$, $e \in E$.

(c) What does the complementary slackness condition look like?

Please turn over

Problem 3: Network Simplex **8 Points**

Apply the network simplex algorithm to find a min-cost flow in the network shown below.



- Use the following initial spanning tree structure:

$$\mathbf{T} = \{(v_1, v_2), (v_3, v_2), (v_2, v_5), (v_4, v_5), (v_4, v_6)\}$$
$$\mathbf{L} = \{(v_3, v_5)\}$$
$$\mathbf{U} = \{(v_1, v_3), (v_2, v_4), (v_5, v_6)\}$$

- Use the first eligible pivot-rule in a wraparound fashion to find the entering arc, using the following order of arcs:

$$(v_1, v_2), (v_1, v_3), (v_2, v_4), (v_2, v_5), (v_3, v_2), (v_3, v_5), (v_4, v_5), (v_4, v_6), (v_5, v_6)$$

  The rule works as follows: Scan the arc list from the beginning and select the first arc that is eligible to enter the spanning tree. In the next step, keep scanning the list, starting at the arc chosen in the first step. When the end of the list is reached, start over at the beginning.