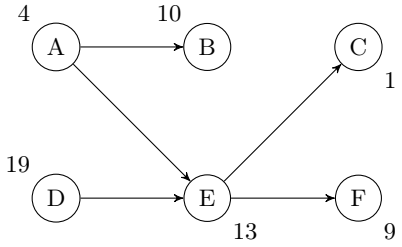
Combinatorial Optimization at Work – Exam

Zuse Institute Berlin

Australia Day

Exercise 1:

Consider the following transitively reduced precedence graph $G = (\mathcal{N}, \mathcal{P})$ with $\mathcal{N} = \{A, B, C, D, E, F\}$ and $\mathcal{P} = \{AB, AE, DE, EC, EF\}$:



An arc ik is in $\mathcal{P}$ if item k is a predecessor of i (e.g. B is a predecessor of A), meaning that if i is selected then k has to be selected as well. Each item $i \in \mathcal{N}$ has weight w_i as is written next to its node (e.g. item A has weight $w_A = 4$). The profit of each item equals its weight: $c_i = w_i$ for all $i \in \mathcal{N}$.

(a) Write down an IP formulation for the corresponding *precedence constrained knapsack problem* with arbitrary knapsack capacity W to maximize the profit. Write all constraints and values explicitly (do not use $\mathcal{N}$, $\mathcal{P}$, w_i, or c_i).

(b) Determine the smallest possible value for W such that for each item in $\mathcal{N}$ there exists a feasible solution containing this item. Give a short reason.

(c) Now suppose we have knapsack capacity $W = 27$: List all items which can not be included in any feasible solution. Give a short reason.

(d) For $W = 27$, state a valid *induced* 2-cover inequality.

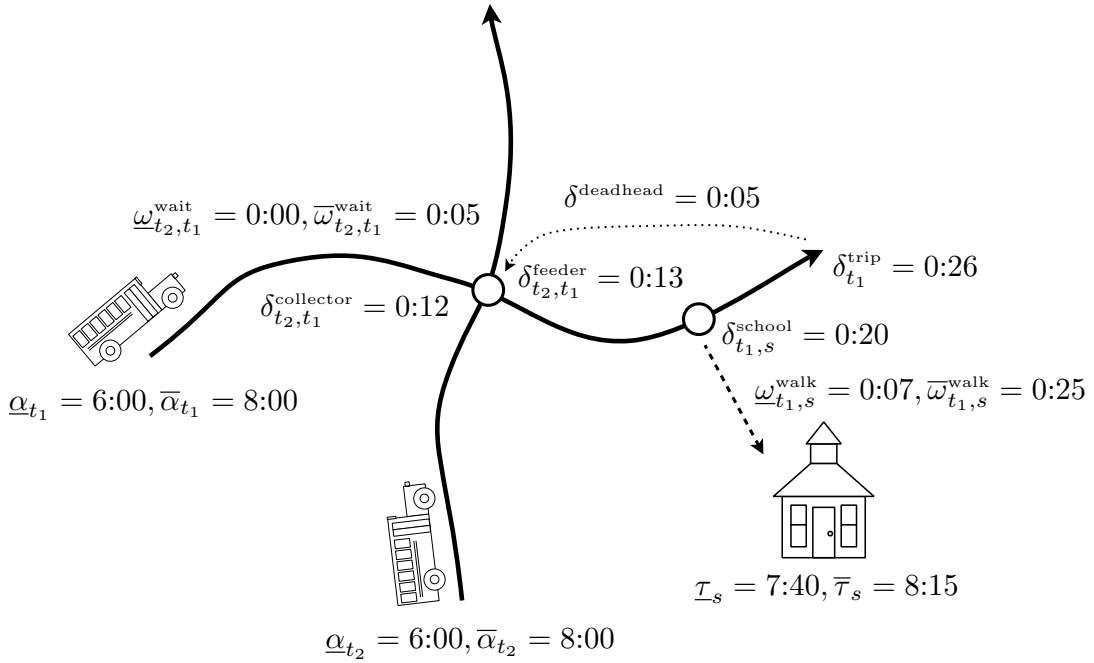
Logistics

Exercise 2: We consider an example of two bus trips t_1, t_2. Trip t_1 has a west-east direction, whereas trip t_2 goes from south to north. The starting time of trips t_1 and t_2 has to be in the interval $\underline{\alpha}_{t_1} = \underline{\alpha}_{t_2} = 6{:}00$ and $\overline{\alpha}_{t_1} = \overline{\alpha}_{t_2} = 8{:}00$. The overall running time of trip t_1 is $\delta^{\text{trip}}_{t_1} = 26$ minutes, and $\delta^{\text{trip}}_{t_2} = 29$ for t_2.

Both trips meet at an intermediate transfer station. Here only transfers from t_2 to t_1 occur. Ideally the waiting time is $\underline{\omega}^{\text{wait}}_{t_2,t_1} = 0$ minutes, that is, both buses arrive at the same time. However, waiting of at most $\overline{\omega}^{\text{wait}}_{t_2,t_1} = 5$ minutes is allowed. The feeder trip t_2 needs $\delta^{\text{feeder}}_{t_2,t_1} = 13$ minutes from the start of the trip to get to the transfer station. The collector trip t_1 needs $\delta^{\text{collector}}_{t_2,t_1} = 12$ minutes from start to this station.

Trip t_1 transports pupils to school s. The ride time from the first bus stop of t_1 to school s is $\delta^{\text{school}}_{t_1,s} = 20$ minutes. When pupils leave the bus they have $\underline{\omega}^{\text{walk}}_{t_1,s} = 7$ minutes time to get to their class room. At most $\overline{\omega}^{\text{walk}}_{t_1,s} = 25$ minutes after their arrival school has to start.

The starting time of school s can vary between $\underline{\tau}_s = 7{:}40$ and $\overline{\tau}_s = 8{:}15$, but must be aligned to 5 minutes (i.e., 7:40, 7:45, 7:50, etc. are allowed, but 7:42 is not allowed, for instance).



1. Compute the best (i.e., smallest) starting time windows for t_1, t_2 and s, that is, $\underline{\alpha}_{t_1}, \overline{\alpha}_{t_1}, \underline{\alpha}_{t_2}, \overline{\alpha}_{t_2}$ and $\underline{\tau}_s, \overline{\tau}_s$.

2. After picking up pupils from trip t_2 at the transfer station, the bus serving trip t_1 is typically overcrowded. The bus company wants to know whether it's possible to let the bus turn at the end of trip t_1 and drive back to the transfer station for a second round, starting at the transfer station and following the route of trip t_1 after that. We assume that this deadhead trip takes $\delta^{\text{deadhead}} = 5$ minutes (the bus takes a more direct route and does not pick up passengers). We further assume that those pupils that couldn't catch the bus on the first route may wait safely at the transfer station for more than $\overline{\omega}^{\text{wait}}_{t_2,t_1} = 5$ minutes. Under this assumptions, is it possible for the bus to drive a second round?

Polyhedral Combinatorics

Exercise 3: Let $G = (V, E)$ be a graph. A subset $S \subseteq V$ is called a *stable set* of G (or a *clique* of G) if no two nodes in S are adjacent (if each pair of nodes in S is adjacent). A *coloring* of G is an assignment of colors to nodes such that any two adjacent nodes have a different color. Let $Q(G)$ be the convex hull of the incidence vectors of the cliques of G, i.e.,

$$Q(G) = \text{conv}\{\chi^K \in \mathbb{R}^V \mid K \subseteq V \text{ is clique}\}.$$

(a) Prove that $Q(G)$ has dimension $|V|$, i.e., show that $Q(G)$ contains $|V| + 1$ points (incidence vectors of cliques) that are affinely independent.
(Hint: The vectors $x_1, \ldots, x_k \in \mathbb{R}^n$ are affinely independent if the vectors $x_2 - x_1, x_3 - x_1, \ldots, x_k - x_1$ are linearly independent.)

(b) Prove that

$$x(S) = \sum_{v \in S} x_v \leq 1$$

is valid for $Q(G)$ for every stable set $S \subseteq V$.

(c) Let $S \subseteq V$ be a stable set. Prove that $x(S) \leq 1$ defines a facet of $Q(G)$ if and only if S is a maximal stable set. (Maximal means that there exists no stable set in G strictly containing S.)

(d) Find an integer linear program whose optimal solutions are the colorings of G with the smallest number of colors, i.e., find an IP formulation of the coloring problem.

Public Transport

Exercise 4: Let $G = (V, E)$ be a connected graph and $\mathfrak{L}$ be a set of predefined simple paths (potential bus lines), called lines from now on, in G with cost $c \in \mathbb{R}_+^{\mathfrak{L}}$. We assume that each edge in G is covered by at least one line in $\mathfrak{L}$. The task is to find a set $\mathfrak{L}' \subseteq \mathfrak{L}$ with minimal cost that connects all nodes in G, i.e., for each two nodes there exists a path in G such that each edge of this path is covered by at least one line of $\mathfrak{L}'$.

Consider the following greedy algorithm:

Input : A connected graph $G = (V, E)$, a set of lines $\mathfrak{L}$ with costs $c \in \mathbb{R}_+^{\mathfrak{L}}$ that covers all edges E.
Output: A set of lines $\mathfrak{L}' \subseteq \mathfrak{L}$ with minimal costs that connects all nodes V.
All nodes are unmarked;
Define $f(\ell) = \frac{|\{v \in V \,|\, v \in \ell \text{ and } v \text{ unmarked }\}|}{c_\ell}$ for $\ell \in \mathfrak{L}$;
$\mathfrak{L}' = \emptyset$;
while $\exists$ *unmarked node* **do**
  Find $\ell \in \mathfrak{L}$ with $f(\ell) = \max\{f(\ell') \,|\, \ell' \in \mathfrak{L}\}$
  $\mathfrak{L}' = \mathfrak{L}' \cup \{\ell\}$;
  for *all* $v \in \ell$ *with* v *unmarked* **do**
    mark v;
    for *all* $\ell' \in \mathfrak{L}$ *with* $v \in \ell'$ **do**
      update $f(\ell')$, i.e., $f(\ell') = f(\ell') - \frac{1}{c_{\ell'}}$;
    end
  end
end
return $\mathfrak{L}'$

Construct an example for which the algorithm does not compute the optimal solution.

Exercise 5: Consider the following linear program:

$$
\begin{array}{lllllll}
\text{(P)} & \min & x_1 & + & x_2 & & \\
\text{(i)} & & x_1 & + & x_2 & = & -1 \\
\text{(ii)} & & x_1, & & x_2 & \leq & 1 \\
\text{(iii)} & & x_1, & & x_2 & \geq & 0
\end{array}
$$

Consider the Lagrange relaxation (L) of (P) with respect to equation (i). Determine the Lagrange function of (L) and show that the optimal values of (L) and (P) are not equal.

Telecommunication

Exercise 6: You have to dimension an Internet backbone network at minimum cost. The following mixed integer programming model takes a directed network $G = (V, A)$ as input together with traffic demands d_{st} for node-pairs $(s, t) \in V \times V$. These demands are realized in the network with a multi-commodity flow (equations (1)). The variable $f_{st}(a) \geq 0$ denotes the flow for demand d_{st} routed on arc $a \in A$. The flow on every arc a is bounded by the number of capacity modules $y(a) \in \mathbb{Z}_+$ times the module capacity $u > 0$ (constraints (2)). The total cost of installed capacity modules is minimized. (A single module costs c.)

$$\min \sum_{a \in A} c y(a)$$

$$\sum_{a \in \delta^+(v)} f_{st}(a) - \sum_{a \in \delta^-(v)} f_{st}(a) = \begin{cases} d_{st}, & v = s \\ 0, & \text{else} \end{cases} \qquad \forall\ (s,t) \in V \times V, v \neq t \tag{1}$$

$$\sum_{(s,t) \in V \times V} f_{st}(a) \leq u y(a) \qquad \forall\ a \in A \tag{2}$$

$$f_{st}(a) \geq 0 \qquad \forall\ a \in A, (s,t) \in V \times V$$

$$y(a) \in \mathbb{Z}_+ \qquad \forall\ a \in A$$

This is a nice model but it is not very useful in practice since it ignores (i) the switching capacity at the nodes and (ii) the survivability of the network. Your task is to extend the above model (constraints and objective function) to meet these additional requirements. It suffices to state additional constraints or necessary changes. Do not write down the whole model.

(i) *Node capacities:* You are given a set M of IP router types. An IP router of type $m \in M$ has capacity $k_m > 0$ and cost $c_m > 0$. At most one IP router can be installed at every node in the network. The total flow (i.e., the sum of flows) leaving and entering a node may not exceed the installed router capacity.

(ii) *Survivability:* If an excavator destroys a cable (arc failure) all demands must still be satisfied. Use the following approach: Firstly double all demands and secondly ensure that at most 50% of the flow for every demand is routed through a particular arc.

Integer Programming

Exercise 7: Let $a \in \mathbb{Z}^n$ and $b \in \mathbb{Z}$. We assume, without loss of generality, $0 < a_j \leq b$ for all $j \in N = \{1, \ldots, n\}$. The *0-1 knapsack polytope* is defined as

$$P := \operatorname{conv}\{x \in \{0,1\}^n : \sum_{j \in N} a_j x_j \leq b\}.$$

1. Give the definition of a *minimal cover* and the corresponding *minimal cover inequality*, which is a valid inequality for P.
2. Consider the 0-1 knapsack polytope

   $$P := \operatorname{conv}\{x \in \{0,1\}^7 : 2\,x_1 + 3\,x_2 + 6\,x_3 + 7\,x_4 + 3\,x_5 + 9\,x_6 + 2\,x_7 \leq 9\}.$$

   Find three valid *minimal cover inequalities* for P such that each of these three inequalities contains a different number of *nonzero* coefficients.

SCIP

Exercise 8:

1. Which of these inclusions does *not* hold?

   O MIP $\subset$ CIP

   O MIP $\subset$ FD

   O FD $\subset$ CIP

   O CIP $\subset$ CP

   Abbreviations: Constraint Integer Programming (CIP), Constraint Programming (CP), Finite Domain (FD), Mixed Integer Programming (MIP)

2. Which of these statements is *not* true?

   O SCIP can be used as a branch-and-price framework.

   O SCIP has a modular structure, the functionality comes in via plugins.

   O SCIP considers the LP solver to be a black-box, the communication happens via interfaces.

   O SCIP is purely object-based, there is an inheritance hierarchy between all components.

3. What are pseudocosts?

   O an estimate of the gain in the objective function per unit change in a single variable

   O the first derivative of the objective function at an extreme point of a polyhedron

   O the objective function of the dual linear program

   O change of the objective value obtained by relaxing a constraint by one unit

4. In general, the currently best node selection strategy for pure feasibility problems is

   O depth first search

   O best first search

   O best estimate search

   O breadth first search

5. Which of the following statements about (SCIP) presolving is true?

   O Presolving is not able to cut off parts of the LP relaxation.

   O Presolving may cut off parts of the LP relaxation, but is not able to cut off feasible solutions.

   O Presolving may cut feasible solutions, but is not able to cut off optimal solutions.

   O Presolving may cut off optimal solutions.

6. Which callback is fundamental for the implementation of a SCIP constraint handler?

   O CONSACTIVE – to locally activate the constraint handler

   O CONSSEPALP – the separation of LP solutions.

   O CONSENFOLP – the enforcement of LP solutions.

   O CONSDELETE – freeing memory