

## Planning and Optimization

M. Helmert, G. Röger  
C. Büchner, T. Keller, S. Sievers

University of Basel  
Fall Semester 2021

### Exercise Sheet E

Due: December 9, 2021

**Important:** for submission, consult the rules at the end of the exercise. Non-adherence to these rules might lead to a penalty in the form of a deduction of marks or, in the worst case, that your submission will not be corrected at all.

The files required for this exercise are in the directory `exercise-e` of the course repository (<https://github.com/aibasel-teaching/planopt-hs21>). All paths are relative to this directory. Update your clone of the repository with `git pull` to see the files. In the virtual machine, `/vagrant/plan-opt-hs21` is the repository.

#### Exercise E.1 (1+2+1+1+2 marks) (Lecture E2)

Consider the STRIPS task  $\Pi = \langle V, I, O, \gamma \rangle$  with variable set  $V = \{a, b, c, d, e, f, g\}$ , initial state  $I$  with  $I(a) = \top$  and  $I(v) = \perp$  for all  $v \in V \setminus \{a\}$ , operator set  $O = \{o_1, o_2, o_3, o_4, o_5, o_6\}$  and goal  $\gamma = e$ . All operators in  $O$  have cost 1 and are defined as follows:

$$\begin{aligned} o_1 &= \langle a, b \wedge \neg a \rangle \\ o_2 &= \langle a, c \rangle \\ o_3 &= \langle b, d \rangle \\ o_4 &= \langle c, d \rangle \\ o_5 &= \langle d \wedge g, e \wedge f \rangle \\ o_6 &= \langle \top, g \rangle \end{aligned}$$

- Provide the simplified relaxed task graph  $sRTG(\Pi^+)$  in graphical form.
- Compute the set of causal fact landmarks with the fixed-point algorithm introduced in Chapter E2. You can annotate the nodes of your graph from (a) as in the lecture, but denote in which order you update the values of the nodes, and if you change the same node several times provide all intermediate values.
- The following is a set of disjunctive action landmarks for  $\Pi$ :

$$\mathcal{L} = \{\{o_3, o_4\}, \{o_1, o_2, o_4\}, \{o_1, o_2, o_5\}, \{o_6\}, \{o_1, o_2, o_4, o_6\}, \{o_1, o_2, o_3\}\}$$

Provide a minimum hitting set for  $\mathcal{L}$ .

- Now consider  $\Pi'$  which is identical to  $\Pi$  except for the operator costs. Namely, we have  $cost'(o_5) = 0$ ,  $cost'(o_3) = cost'(o_4) = 1$ ,  $cost'(o_2) = cost'(o_6) = 2$ , and  $cost'(o_1) = 3$ . Find and provide a minimum hitting set for  $\mathcal{L}$  under cost function  $cost'$ . What is its cost?
- Provide a disjunctive action landmark  $L$  with  $|L| = 2$  such that  $L \notin \mathcal{L}$  and  $L' \not\subseteq L$  for all  $L' \in \mathcal{L}$  (this is because all supersets of e.g.  $\{o_6\}$  are trivial landmarks but we would like you to find something else). Also provide an operator set  $X \subset O$  with  $|X| = 2$  that is *not* a disjunctive action landmark for  $\Pi$ . Briefly justify why  $L$  is a disjunctive action landmark and why  $X$  is not (no formal proof required).

#### Exercise E.2 (5 marks) (Lecture E3)

Consider the delete-free STRIPS planning task  $\Pi^+ = \langle V, I, O, \delta \rangle$  with  $V = \{i, a, b, c, d, e, g\}$ ,  $I = \{i \mapsto \mathbf{T}\} \cup \{v \mapsto \mathbf{F} \mid v \in V \setminus \{i\}\}$ ,  $\delta = g$ , and  $O = \{o_1, \dots, o_7\}$ , where

$$\begin{aligned}
o_1 &= \langle \{i\}, \{a\}, \{\}, 1 \rangle \\
o_2 &= \langle \{i\}, \{b\}, \{\}, 1 \rangle \\
o_3 &= \langle \{a\}, \{b, c\}, \{\}, 4 \rangle \\
o_4 &= \langle \{b\}, \{a, c\}, \{\}, 5 \rangle \\
o_5 &= \langle \{a, b\}, \{d\}, \{\}, 3 \rangle \\
o_6 &= \langle \{c, d\}, \{e\}, \{\}, 2 \rangle \\
o_7 &= \langle \{d, e\}, \{g\}, \{\}, 0 \rangle.
\end{aligned}$$

Compute  $h^{\text{LM-cut}}(I)$  and provide all intermediate results in the same way they were given in the example of the lecture. Specifically, provide for each iteration (except the last):

- the justification graph with  $h^{\max}$  annotations and marked goal zone,
- the cut,
- the cost of the cut, and
- the updated costs.

If multiple preconditions of an operator have the same maximal  $h^{\max}$  value, the precondition choice function breaks ties in alphabetical order (i.e. for an operator  $o = \langle \{a, b\}, \{c\}, \{\}, 2 \rangle$  with  $h^{\max}(a) = 3$  and  $h^{\max}(b) = 3$ , we have  $pcf(o) = a$ ).

### Exercise E.3 (3+1+2)(Lecture E4)

Consider the following problem: you want to minimize the money you spend for breakfast but at the same time want to eat healthy, i.e., get at least the recommended dose of relevant nutrients. There are five types of food: cereals, eggs, bread, cheese, honey. The relevant nutrients for your breakfast are carbohydrates, proteins and fat. One unit of each food has the following cost:

- cereals: 3
- eggs: 4
- bread: 2
- cheese: 5
- honey: 3

One unit of each type of food contains the following amount of units of nutrients:

- cereals: 5 carbohydrates, 2 proteins, 2 fat
- eggs: 6 protein, 4 fat
- bread: 8 carbohydrates, 1 protein
- cheese: 10 fat, 3 protein
- honey: 10 carbohydrates

You want your breakfast to cover at least the following amount of units of nutrients:

- carbohydrates: 100
- proteins: 40

- fat: 20

- Formalize the problem as an integer problem (IP).
- What would you need to change to obtain a linear problem (LP)?
- Encode the LP as input to SoPlex and use SoPlex to solve it. Provide the solution found by SoPlex, i.e., the objective value and the values of all variables. Please also submit a single file encoding the LP with clear names for variables and constraints.

*Instructions on how to install and use SoPlex are in the file `soplex-readme.txt`.*

#### Exercise E.4 (5 marks)(Lecture E4)

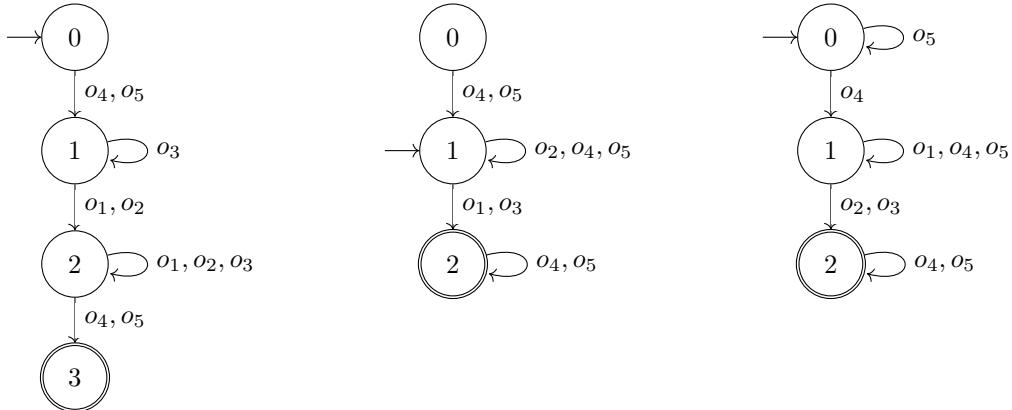
In this exercise, we want to draw a connection from integer programming to delete-relaxed planning tasks. In particular, the paper below describes an IP model for the delete relaxation. Provide the definition of the IP, however adapting the notation of the paper to the notation of planning tasks from the lecture. Explain the IP model in detail, i.e., explain the meaning of all involved variables and constraints. Use your own words and the notation from the lecture instead of the notation from the paper. Note that you do not have to read the paper beyond Section 2, i.e., reading (not even) the first two pages is enough.

Tatsuya Imai and Alex Fukunaga. A Practical, Integer-Linear Programming Model for the Delete-Relaxation in Cost-Optimal Planning. In *Proc. ECAI 2014*. pp. 459–464, 2014.

*A good answer can be written in less than 1 page.*

#### Exercise E.5 (1+1+1+2 marks)(Lecture E5)

Consider the planning task  $\Pi = \langle V, I, O, \gamma \rangle$  with  $V = \{a, b, c\}$ ,  $\text{dom}(a) = \{0, 1, 2, 3\}$ ,  $\text{dom}(b) = \text{dom}(c) = \{0, 1, 2\}$ ,  $I = \{a \mapsto 0, b \mapsto 1, c \mapsto 0\}$ ,  $O = \{o_1, o_2, o_3, o_4, o_5\}$ ,  $\text{cost}(o_i) = 1$  for all  $i \in \{1, 2, 3, 5\}$ ,  $\text{cost}(o_4) = 2$ , and  $\gamma = (a = 3) \wedge (b = 2) \wedge (c = 2)$ . The transition systems of the projections to variables are shown below:  $\mathcal{T}^{\pi_{\{a\}}}$  on the left,  $\mathcal{T}^{\pi_{\{b\}}}$  in the middle,  $\mathcal{T}^{\pi_{\{c\}}}$  on the right.



- For abstraction heuristics  $h^{\alpha_1}, \dots, h^{\alpha_n}$ , the uniform cost partitioning is defined for all  $o \in O$  and  $1 \leq i \leq n$  by

$$\text{cost}_i(o) := \begin{cases} \frac{\text{cost}(o)}{n_o} & \text{if } o \text{ affects } \mathcal{T}^{\alpha_i} \\ 0 & \text{otherwise,} \end{cases}$$

where  $n_o$  is the number of abstractions  $\alpha$  such that  $o$  affects  $\mathcal{T}^\alpha$ .

Compute the uniform cost partitioning for the three abstraction heuristics  $h^{\{a\}}, h^{\{b\}}, h^{\{c\}}$  and provide the corresponding heuristic value in the initial state  $I$ .

(b) For abstraction heuristics  $h^{\alpha_1}, \dots, h^{\alpha_n}$ , a zero-one cost partitioning is defined for all  $o \in O$  by  $cost_i(o) := cost(o)$  for exactly one  $i \in \{1, \dots, n\}$ , and  $cost_j(o) := 0$  for  $j \neq i$ . Compute a zero-one cost partitioning for the three abstraction heuristics  $h^{\{a\}}, h^{\{b\}}, h^{\{c\}}$  which yields the maximal heuristic value in the initial state  $I$  among all zero-one cost partitionings. Provide the corresponding heuristic value in  $I$  and justify why there is no zero-one cost partitioning with higher heuristic value for  $I$ .

(c) Compute the saturated cost partitionings for the three abstraction heuristics  $h^{\{a\}}, h^{\{b\}}, h^{\{c\}}$  for the two following orders:  $\langle h^{\{a\}}, h^{\{b\}}, h^{\{c\}} \rangle$  and  $\langle h^{\{b\}}, h^{\{c\}}, h^{\{a\}} \rangle$ . For both cost partitionings, provide both the cost and the minimal saturated cost functions for all three heuristics and provide the heuristic value of the resulting cost partitioning heuristic in  $I$ .

(d) Provide a cost partitioning that yields a higher heuristic value than the cost partitionings from (a), (b) and (c). What is the value of the corresponding heuristic in the initial state?

**Exercise E.6** (3+1 marks)(Lecture E6)

Consider the following task  $\Pi = \langle V, I, O, \gamma \rangle$  with

- Variables  $V = \{a, b\}$  with  $dom(a) = dom(b) = \{1, 2, 3\}$ ,
- Initial state  $I = \{a \mapsto 1, b \mapsto 1\}$ ,
- Operators  $O = \{o_1, \dots, o_4\}$  where
  - $o_1 = \langle (a = 1), (a := 2) \rangle, cost(o_1) = 2$
  - $o_2 = \langle (a = 2), (a := 3) \rangle, cost(o_2) = 2$
  - $o_3 = \langle (b = 1) \wedge (a = 2), (b := 2) \wedge (a := 1) \rangle, cost(o_3) = 1$
  - $o_4 = \langle (b = 2) \wedge (a = 3), (b := 3) \wedge (a := 1) \rangle, cost(o_4) = 1$
- Goal  $\gamma = (b = 3)$ .

(a) Use SoPlex to compute an optimal non-negative cost partitioning of the projections to  $a$  and  $b$ . Please submit the result (i) as a file encoding the LP with clear names for variables and constraints and (ii) as a table with one column per operator and one row per abstraction where each cell contains the cost of the operator in the abstraction.

*Instructions on how to install and use SoPlex are in the file `soplex-readme.txt`.*

(b) Repeat exercise (a) for a general cost partitioning.

**Exercise E.7** (3+3+2 marks)(Lectures E7, E7, E8)

Consider the following task  $\Pi = \langle V, I, O, \gamma \rangle$  with

- Variables  $V = \{a, b, c\}$  with  $dom(a) = dom(c) = \{1, 2, 3, 4, 5\}$ , and  $dom(b) = \{1, 2, 3, 4, 5, 6, 7\}$ ,
- Initial state  $I = \{a \mapsto 1, b \mapsto 1, c \mapsto 1\}$ ,
- Operators  $O = \{o_1, \dots, o_{12}\}$  where
  - $o_1 = \langle (a = 1), (a := 2) \rangle$
  - $o_2 = \langle (a = 2) \wedge (b = 1) \wedge (c = 1), (a := 1) \wedge (b := 3) \rangle$
  - $o_3 = \langle (a = 1) \wedge (b = 4) \wedge (c = 1), (b := 2) \wedge (c := 2) \rangle$
  - $o_4 = \langle (a = 1) \wedge (b = 3) \wedge (c = 1), (b := 4) \rangle$
  - $o_5 = \langle (a = 2) \wedge (b = 2) \wedge (c = 2), (b := 4) \rangle$
  - $o_6 = \langle (a = 2) \wedge (b = 4) \wedge (c = 2), (b := 3) \rangle$
  - $o_7 = \langle (a = 2) \wedge (b = 3) \wedge (c = 2), (b := 6) \wedge (c := 4) \rangle$

- $o_8 = \langle (a = 2) \wedge (b = 3) \wedge (c = 2), (a := 3) \wedge (b := 5) \rangle$
- $o_9 = \langle (a = 2) \wedge (b = 6) \wedge (c = 2), (a := 4) \wedge (c := 3) \rangle$
- $o_{10} = \langle (c = 4), (c := 2) \rangle$
- $o_{11} = \langle (a = 3) \wedge (b = 5) \wedge (c = 4), (a := 5) \wedge (b := 7) \wedge (c := 5) \rangle$
- $o_{12} = \langle (a = 4) \wedge (b = 6) \wedge (c = 3), (a := 5) \wedge (b := 7) \wedge (c := 5) \rangle$

and  $\text{cost}(o) = 1$  for all  $o \in O$ ,

- Goal  $\gamma = (a = 5) \wedge (b = 7) \wedge (c = 5)$ .

(a) Provide the LP solved by the flow heuristic for  $I$  as an input file for SoPlex. Use each atom as the constraint name for its flow constraint so it is easy to see which constraint belongs to which atom. Then solve the LP and provide the objective value.

*Instructions on how to install and use SoPlex are in the file `soplex-readme.txt`.*

*Python code encoding the task is available in the directory `network-flow` but no programming is required.*

(b) Draw the transition systems of the three projections to  $a$ ,  $b$ , and  $c$ . For operators that do not mention the variable, include just one representative self-loop at the goal state to keep the transition system concise. Annotate each edge with the flow of the operator from exercise (a) and highlight edges with a non-zero value. What do you notice in the abstractions? Discuss your observations.

*LaTeX source code for drawing the abstract transition systems is available in the directory `network-flow`.*

(c) The sets  $L_1 = \{o_4, o_6\}$  and  $L_2 = \{o_{10}, o_{12}\}$  are disjunctive action landmarks for  $\Pi$ . Provide the LP solved by the operator counting LP heuristic  $h_C^{LP}$  for  $I$ , where  $C$  contains the flow constraints from part (a) and the operator counting constraints induced by  $L_1$  and  $L_2$ . Then solve the LP and provide the objective value.

### Submission rules:

- Exercise sheets must be submitted in groups of three students. Please submit a single copy of the exercises per group (only one member of the group does the submission).
- Create a single PDF file (ending `.pdf`) for all non-programming exercises. Use a file name that does not contain any spaces or special characters other than the underscore `_`. If you want to submit handwritten solutions, include their scans in the single PDF. Make sure it is in a reasonable resolution so that it is readable, but ensure at the same time that the PDF size is not astronomically large. Put the names of all group members on top of the first page. Either use page numbers on all pages or put your names on each page. Make sure your PDF has size A4 (fits the page size if printed on A4).
- For programming exercises, only create those code textfiles required by the exercise. Put your names in a comment on top of each file. Make sure your code compiles and test it. Code that does not compile or which we cannot successfully execute will not be graded.
- For the submission: if the exercise sheet does not include programming exercises, simply upload the single PDF. If the exercise sheet includes programming exercises, upload a ZIP file (ending `.zip`, `.tar.gz` or `.tgz`; *not* `.rar` or anything else) containing the single PDF and the code textfile(s) and nothing else. Do not use directories within the ZIP, i.e., zip the files directly.
- Do not upload several versions to ADAM, i.e., if you need to resubmit, use the same file name again so that the previous submission is overwritten.