Foundations of Artificial Intelligence

M. Helmert
S. Eriksson
Spring Term 2021

University of Basel
Computer Science

Exercise Sheet 13

Due: June 2, 2021

All exercises on this sheet are bonus exercises.

The goal of this exercise is to compare different problem solving methods covered in this course on the problem of *graph coloring*: Given an undirected graph $\mathbf{G} = \langle \mathbf{V}, \mathbf{E} \rangle$ (with $\mathbf{V} = \{\mathbf{v}_1, \dots, \mathbf{v}_{|\mathbf{V}|}\}$ and $\mathbf{E} \subseteq \{\{\mathbf{v}_i, \mathbf{v}_j\} \mid \mathbf{v}_i \in \mathbf{V}, \mathbf{v}_j \in \mathbf{V}, \mathbf{v}_i \neq \mathbf{v}_j\}$) and a finite set of colors $\mathbf{C}$, assign each vertex $\mathbf{v}_i \in \mathbf{V}$ a color $\mathbf{c} \in \mathbf{C}$ such that for every edge $\{\mathbf{v}_i, \mathbf{v}_j\} \in \mathbf{E}$ the colors assigned to $\mathbf{v}_i$ and $\mathbf{v}_j$ differ. We can represent this problem as *combinatorial optimization problem*, *constraint network*, *CNF formula* and *STRIPS planning task* in the following way:

Combinatorial optimization We define $P = \langle C, S, \min, v \rangle$ with

- $C = \{\langle c_1, \dots, c_{|\mathbf{V}|}\rangle \mid c_i \in \mathbf{C} \textit{ for all } 1 \leq i \leq |\mathbf{V}|\}$ where c_i denotes the color assigned to $\mathbf{v}_i$,
- $S = \{\langle c_1, \dots, c_{|\mathbf{V}|}\rangle \mid \langle c_1, \dots, c_{|\mathbf{V}|}\rangle \in C \textit{ and } c_i \neq c_j \textit{ for all } \{\mathbf{v}_i, \mathbf{v}_j\} \in \mathbf{E}\}$,
- $v(c) = 0$ for all $c \in S$,

For hill climbing we define the neighbors of $c \in C$ to be all c' which differ from c in the assignment of exactly one c_i. Furthermore we use the heuristic $h(c) = |\{\{\mathbf{v}_i, \mathbf{v}_j\} \mid \{\mathbf{v}_i, \mathbf{v}_j\} \in \mathbf{E} \textit{ and } c_i = c_j\}|$.

Constraint Network We define $\mathcal{C} = \langle V_{\mathcal{C}}, \mathrm{dom}, (R_{xy})\rangle$ with

- $V_{\mathcal{C}} = \{x_i \mid \mathbf{v}_i \in \mathbf{V}\}$,
- $\mathrm{dom}(x_i) = \mathbf{C}$ for all $x_i \in V_{\mathcal{C}}$, and
- $R_{x_i x_j} = \{\langle \mathbf{c}, \mathbf{c}'\rangle \mid \mathbf{c}, \mathbf{c}' \in \mathbf{C} \text{ and } \mathbf{c} \neq \mathbf{c}'\}$ for all $\{\mathbf{v}_i, \mathbf{v}_j\} \in \mathbf{E}$.

CNF formula We define a set of propositional variables $V_{\mathrm{CNF}} = \{v^c \mid \mathbf{v} \in \mathbf{V}, \mathbf{c} \in \mathbf{C}\}$, where v^c denotes that vertex $\mathbf{v}$ is assigned color $\mathbf{c}$, and encode the problem with:

$$\Phi = \bigwedge_{\mathbf{v} \in \mathbf{V}} \Big(\bigvee_{\mathbf{c} \in \mathbf{C}} v^c\Big) \wedge \bigwedge_{\mathbf{v} \in \mathbf{V}} \bigwedge_{\{\mathbf{c}, \mathbf{c}'\} \subseteq \mathbf{C}, \mathbf{c} \neq \mathbf{c}'} (\neg v^c \vee \neg v^{c'}) \wedge \bigwedge_{\{\mathbf{v}_i, \mathbf{v}_j\} \in \mathbf{E}} \bigwedge_{\mathbf{c} \in \mathbf{C}} (\neg v_i^c \vee \neg v_j^c)$$

Planning We define a STRIPS planning task $\Pi = \langle V_\Pi, I, G, A\rangle$:

- $V_\Pi = \{\texttt{v-not-c} \mid \mathbf{v} \in \mathbf{V}, \mathbf{c} \in \mathbf{C}\} \cup \{\texttt{v-assigned} \mid \mathbf{v} \in \mathbf{V}\} \cup \{\texttt{v-not-assigned} \mid \mathbf{v} \in \mathbf{V}\}$, where `v-not-c` denotes that vertex $\mathbf{v}$ is *not* colored with $\mathbf{c}$, `v-assigned` denotes that $\mathbf{v}$ has been assigned a color and `v-not-assigned` denotes that $\mathbf{v}$ has not been assigned any color.
- $I = \{\texttt{v-not-c} \mid \mathbf{v} \in \mathbf{V}, \mathbf{c} \in \mathbf{C}\} \cup \{\texttt{v-not-assigned} \mid \mathbf{v} \in \mathbf{V}\}$, i.e., all vertices have not been assigned any color yet.
- $G = \{\texttt{v-assigned} \mid \mathbf{v} \in \mathbf{V}\}$, i.e., all vertices must be assigned to some color.
- $A = \{\texttt{assign-v-c} \mid \mathbf{v} \in \mathbf{V}, \mathbf{c} \in \mathbf{C}\}$, where for each action $a = \texttt{assign-v-c}$ we have

  $$\mathit{cost}(a) = 1 \qquad \mathit{pre}(a) = \{\texttt{v-not-assigned}\} \cup \{\texttt{v}'\texttt{-not-c} \mid \{\mathbf{v}, \mathbf{v}'\} \in \mathbf{E}\}$$
  $$\mathit{add}(a) = \{\texttt{v-assigned}\} \qquad \mathit{del}(a) = \{\texttt{v-not-c}, \texttt{v-not-assigned}\}.$$

Exercise 13.1 (4 bonus marks)

Compare the number of candidates of P, the number of (full) assigments of $\mathcal{C}$, the number of interpretations of Φ and the number of states of Π. Which encoding(s) is/are the most and least compact?

Exercise 13.2 (1 bonus mark)

Assume we want to apply backtracking with inference using minimum remaining values as variable order on a graph-coloring constraint network $\mathcal{C}$ as defined above.
Why is it not beneficial to use arc consistency in addition to forward checking as an inference method?

Exercise 13.3 (1 bonus mark)

Assume we apply DPLL to Φ. When DPLL sets a variable v^c to $\mathbf{T}$ (i.e., assigns $\mathbf{c}$ to vertex $\mathbf{v}$), which variable assignments can be inferred with unit propagation? Justify your answer.

Exercise 13.4 (2 bonus marks)

Consider a STRIPS task Π encoding a graph coloring problem as described above. Are the following statements correct? Justify your answer.

(a) All solutions of Π have cost $|\mathbf{V}|$.

(b) Every state reachable from I corresponds to a consistent partial assignment of $\mathcal{C}$.

Exercise 13.5 (2 bonus marks)

Compare how well the following algorithms are suited for solving graph coloring:

- Hill climbing on the combinatorial optimization problem P,
- Backtracking with forward checking on the constraint network $\mathcal{C}$,
- DPLL on Φ, and
- A^* with an admissible heuristic on Π.

Which one would you choose? List one advantage of each algorithm, and one disadvantage of each one you would not choose.
Hint: the solution to the other parts of this sheet can and should be used here!

Submission rules:
Upload a single PDF file (ending .pdf). If you want to submit handwritten parts, include their scans in the single PDF. Put the names of all group members on top of the first page. Use page numbers or put your names on each page. Make sure your PDF has size A4 (fits the page size if printed on A4).