Künstliche Intelligenz – Übung 8

Marius Frinken

Friedrich-Alexander-Universität Erlangen–Nürnberg

19.12.2018

Organizational

Homework 7

Homework 8

Misc: Questions, Anecdotes & etc

Organizational

Personal information

my email address: marius.frinken@fau.de

PGP encrypted mails are preferred!

my PGP fingerprint:

`F4BD 7ED4 96A5 9BA6 9FD6 901C 1EEC 9B1B 8CD5 3DA1`

Hand-In Procedure

Please hand in ***.pdf**, ***.txt** or ***.pl** files!

Please include your name in your files, my suggestions:

- **LaTeX:** use the `author` field together with a `\maketitle` command
- **.txt :** simply write your name at the top
- **.pl Code :** use comments for a header:

```
% author:  Alan Turing
```

Solutions

Solutions for old homeworks are available at
https://kwarc.info/teaching/AI/assignments.pdf

Homework 7

Problem 7.1

(have a look at the official solution)

Problem 7.2

One way to solve this:

- new variable x_4 that is a tuple of x_1 and x_2: $x_4 = (x_1, x_2)$
- $D_{x_4} = \{D_{x_1} \times D_{x_2}\}$ and $(u, v) \in D_{x_4}$ iff $\exists x_3.(u, v, x_3) \in C_{tern}$ where C_{tern} is the ternary Constraint
- now we need a binary Constraint $C_{x_3 x_4} \subseteq \{D_{x_4} \times D_{x_3}\}$ where $(y, z) \in C_{x_3 x_4}$ iff $y = (u, v)$ and $(u, v, z) \in C_{tern}$

(gray parts are not needed but make things easier/faster)

Problem 7.3

The 50 Queens:

- $V = \{Q_1, ..., Q_{50}\}$, where Q_i is the queen in row i
- $D_i = \{1, ..., 50\}$ representing the queen's column
- $C = \{C_{uv}\}$ for all combinations of two queens u, v where $u \neq v$
- $C_{uv} = \{(Q_u, Q_v)\}$ where $Q_u \neq Q_v$ **and** $|Q_u - Q_v| \neq |u - v|$ (no queen in the same column and no queen on a diagonal)

Homework 8

Recap: Arc-Consistency I

Arc Consistency: Definition

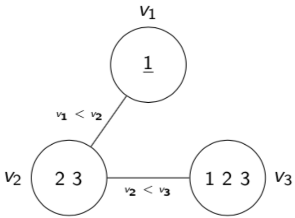
- **Definition 4.1 (Arc Consistency).** Let $\gamma = \langle V, D, C \rangle$ be a constraint network.
  - (i) A variable $u \in V$ is arc consistent relative to another variable $v \in V$ if either $C_{uv} \notin C$, or for every value $d \in D_u$ there exists a value $d' \in D_v$ such that $(d, d') \in C_{uv}$.
  - (ii) The network γ is arc consistent if every variable $u \in V$ is arc consistent relative to every other variable $v \in V$.
- Arc consistency = for every domain value and constraint, at least one value on the other side of the constraint "works".
- Note the asymmetry between u and v: arc consistency is "directed".
- **Examples:** (previous slide)
  - On top, middle, is v_3 arc consistent relative to v_2?

Enforcing Arc Consistency for *One* Pair of Variables

- **Algorithm enforcing consistency of u relative to v:**

  function Revise(γ,u,v) **returns** modified γ
    for each $d \in D_u$ **do**
      if there is no $d' \in D_v$ with $(d,d') \in C_{uv}$ **then** $D_u := D_u \backslash \{d\}$
    return γ

- Runtime, if k is maximal domain size: $\mathcal{O}(k^2)$, based on implementation where the test "$(d,d') \in C_{uv}$?" is constant time.
- **Example 4.2.** Revise(γ, v_3, v_2)



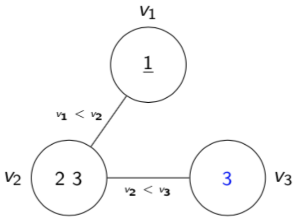
Recap: Arc-Consistency III

Enforcing Arc Consistency for *One* Pair of Variables

- **Algorithm enforcing consistency of u relative to v:**

  function Revise(γ,u,v) **returns** modified γ
    for each $d \in D_u$ **do**
      if there is no $d' \in D_v$ with $(d, d') \in C_{uv}$ **then** $D_u := D_u \backslash \{d\}$
    return γ

- Runtime, if k is maximal domain size: $\mathcal{O}(k^2)$, based on implementation where the test "$(d, d') \in C_{uv}$?" is constant time.
- **Example 4.2.** Revise(γ, v_3, v_2)



Example

$V = \{X, Y, Z\}$
$D_X = D_Y = D_Z = \{0, 1\}$
implicitly/dirtily defined constraints:
$C = \{X \neq Y, Y \neq Z, Z \neq X\}$

Example

$V = \{X, Y, Z\}$
$D_X = D_Y = D_Z = \{0, 1\}$
implicitly/dirtily defined constraints:
$C = \{X \neq Y, Y \neq Z, Z \neq X\}$

Now does enforcing Arc-Consistency do anything? Is this CSP solvable?

Example

$V = \{X, Y, Z\}$
$D_X = D_Y = D_Z = \{0, 1\}$
implicitly/dirtily defined constraints:
$C = \{X \neq Y, Y \neq Z, Z \neq X\}$

Now does enforcing Arc-Consistency do anything? Is this CSP solvable?
NO and NO!
Now consider the same CSP but without the last Constraint $Z \neq X$.

Example

$V = \{X, Y, Z\}$
$D_X = D_Y = D_Z = \{0, 1\}$
implicitly/dirtily defined constraints:
$C = \{X \neq Y, Y \neq Z, Z \neq X\}$

Now does enforcing Arc-Consistency do anything? Is this CSP solvable?
NO and NO!
Now consider the same CSP but without the last Constraint $Z \neq X$. Now, the Arc-Consistency directly says that this is solvable!

Conclusion: Arc-Consistency is really nice when a Constraint-Graph is acyclic. How nice? See Problem 8.1

Recap: AcyclicCG

Acyclic Constraint Graphs: How To

- **Algorithm:** AcyclicCG(γ)
  1. Obtain a directed tree from γ's constraint graph, picking an arbitrary variable v as the root, and directing arcs outwards.[1]
  2. Order the variables topologically, i.e., such that each vertex is ordered before its children; denote that order by $v_1, \ldots, v_n$.
  3. **for** $i := n, n-1, \ldots, 2$ **do**:
     3.1 Revise($\gamma, v_{parent(i)}, v_i$).
     3.2 **if** $D_{v_{parent(i)}} = \emptyset$ **then return** "inconsistent"

     Now, every variable is arc consistent relative to its children.
  4. Run BacktrackingWithInference with forward checking, using the variable order $v_1, \ldots, v_n$.
- This algorithm will find a solution without ever having to backtrack!
- Proof: Exercises

[1] We assume here that γ's constraint graph is connected. If it is not, do this and the following for each connected component separately.

Problem 8.1

(have a look at the task)

Problem 8.2

- What are implicitly defined constraints: see the example from above on these slides
- How to draw a constraint graphs: see lecture or black-board

Recap: PL – Assignments & Interpretations

(go through slides 284 – 288)

Problem 8.3

(have a look at the task)

Example

We want to show that $A \Rightarrow A$ is valid using the definitions of **Interpretation**, **Value Function** and **Assignment**("quick" and dirty):

$$\mathcal{M} \models^{\varphi} A \Rightarrow A \text{ for all Assignments } \varphi \tag{1}$$

Example

We want to show that $A \Rightarrow A$ is valid using the definitions of **Interpretation**, **Value Function** and **Assignment**("quick" and dirty):

$$\mathcal{M} \models^{\varphi} A \Rightarrow A \text{ for all Assignments } \varphi \tag{1}$$

$$\overset{valid}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(A \Rightarrow A) = T \tag{2}$$

Example

We want to show that $A \Rightarrow A$ is valid using the definitions of **Interpretation**, **Value Function** and **Assignment**("quick" and dirty):

$$\mathcal{M} \models^{\varphi} A \Rightarrow A \text{ for all Assignments } \varphi \tag{1}$$

$$\overset{valid}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(A \Rightarrow A) = T \tag{2}$$

$$\overset{\Rightarrow}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg A \vee A) = T \tag{3}$$

Example

We want to show that $A \Rightarrow A$ is valid using the definitions of **Interpretation**, **Value Function** and **Assignment**("quick" and dirty):

$$\mathcal{M} \models^{\varphi} A \Rightarrow A \text{ for all Assignments } \varphi \tag{1}$$

$$\overset{valid}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(A \Rightarrow A) = T \tag{2}$$

$$\overset{\Rightarrow}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg A \vee A) = T \tag{3}$$

$$\overset{\vee}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg(\neg\neg A \wedge \neg A)) = T \tag{4}$$

Example

We want to show that $A \Rightarrow A$ is valid using the definitions of **Interpretation**, **Value Function** and **Assignment**("quick" and dirty):

$$\mathcal{M} \models^{\varphi} A \Rightarrow A \text{ for all Assignments } \varphi \tag{1}$$

$$\overset{valid}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(A \Rightarrow A) = T \tag{2}$$

$$\overset{\Rightarrow}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg A \vee A) = T \tag{3}$$

$$\overset{\vee}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg(\neg\neg A \wedge \neg A)) = T \tag{4}$$

$$\overset{\mathcal{I}(\neg)}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg\neg A \wedge \neg A) = F \tag{5}$$

Example

We want to show that $A \Rightarrow A$ is valid using the definitions of **Interpretation**, **Value Function** and **Assignment**("quick" and dirty):

$$\mathcal{M} \models^{\varphi} A \Rightarrow A \text{ for all Assignments } \varphi \tag{1}$$

$$\overset{valid}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(A \Rightarrow A) = T \tag{2}$$

$$\overset{\Rightarrow}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg A \vee A) = T \tag{3}$$

$$\overset{\vee}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg(\neg\neg A \wedge \neg A)) = T \tag{4}$$

$$\overset{\mathcal{I}(\neg)}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg\neg A \wedge \neg A) = F \tag{5}$$

$$\overset{\mathcal{I}(\wedge)}{\Leftrightarrow} \text{ for any } \varphi : \text{ either } \mathcal{I}_{\varphi}(\neg\neg A) = F \quad \text{or: } \mathcal{I}_{\varphi}(\neg A) = F \tag{6}$$

Example

We want to show that $A \Rightarrow A$ is valid using the definitions of **Interpretation**, **Value Function** and **Assignment**("quick" and dirty):

$$\mathcal{M} \models^{\varphi} A \Rightarrow A \text{ for all Assignments } \varphi \tag{1}$$

$$\overset{valid}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(A \Rightarrow A) = T \tag{2}$$

$$\overset{\Rightarrow}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg A \vee A) = T \tag{3}$$

$$\overset{\vee}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg(\neg\neg A \wedge \neg A)) = T \tag{4}$$

$$\overset{\mathcal{I}(\neg)}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg\neg A \wedge \neg A) = F \tag{5}$$

$$\overset{\mathcal{I}(\wedge)}{\Leftrightarrow} \text{ for any } \varphi : \text{ either } \mathcal{I}_{\varphi}(\neg\neg A) = F \text{ or: } \mathcal{I}_{\varphi}(\neg A) = F \tag{6}$$

$$\overset{\mathcal{I}(\neg) thrice}{\Leftrightarrow} \text{ for any } \varphi : \text{ either } \mathcal{I}_{\varphi}(A) = F \text{ or: } \mathcal{I}_{\varphi}(A) = T \tag{7}$$

Example

We want to show that $A \Rightarrow A$ is valid using the definitions of **Interpretation**, **Value Function** and **Assignment**("quick" and dirty):

$$\mathcal{M} \models^{\varphi} A \Rightarrow A \text{ for all Assignments } \varphi \tag{1}$$

$$\overset{valid}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(A \Rightarrow A) = T \tag{2}$$

$$\overset{\Rightarrow}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg A \vee A) = T \tag{3}$$

$$\overset{\vee}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg(\neg\neg A \wedge \neg A)) = T \tag{4}$$

$$\overset{\mathcal{I}(\neg)}{\Leftrightarrow} \text{ for any } \varphi : \mathcal{I}_{\varphi}(\neg\neg A \wedge \neg A) = F \tag{5}$$

$$\overset{\mathcal{I}(\wedge)}{\Leftrightarrow} \text{ for any } \varphi : \text{ either } \mathcal{I}_{\varphi}(\neg\neg A) = F \text{ or: } \mathcal{I}_{\varphi}(\neg A) = F \tag{6}$$

$$\overset{\mathcal{I}(\neg)thrice}{\Leftrightarrow} \text{ for any } \varphi : \text{ either } \mathcal{I}_{\varphi}(A) = F \text{ or: } \mathcal{I}_{\varphi}(A) = T \tag{7}$$

$$\overset{def\,\varphi}{\Leftrightarrow} \text{this holds always} \quad \square \tag{8}$$

Template

The previous slides concerning the example $A \Rightarrow A$ were **quick and dirty**.
For a complete proof and a nice LaTeXtemplate see https://gl.kwarc.info/teaching/AI/tree/master/Marius/uebung08

Example II

We want to show that $A \wedge \neg A$ is **NOT** valid using an assignment as a counterexample:

Example II

We want to show that $A \wedge \neg A$ is **NOT** valid using an assignment as a counterexample:

$$\varphi(A) = T$$

Here it is obvious that if we apply the Interpretation $\mathcal{I}$ with this assignment on our formula ($A \wedge \neg A$), it will output **F**.

IMPORTANT

USE THE NOTATIONS WE INTRODUCE IN THE LECTURE!

(see slide 285)

only exceptions: $\top, \bot$ for T,F

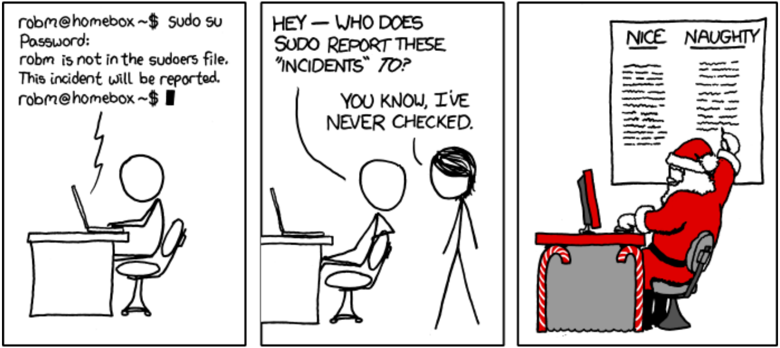
Tips for logic-newbies

- **Wikipedia!** (English) (but still obey the last slide!)
- **De Morgan's laws**
- LaTeXis how **WE** write the lecture/the assignments, so you might consider using it as well
- most of this stuff is a matter of how experienced you are $\Rightarrow$ exercise (=training) helps a lot
- Problems and no tutor available? $\Rightarrow$ Ask any computer science student that has passed the 3rd semester

Misc: Questions, Anecdotes & etc

Questions?

Happy holidays!



Source: https://xkcd.com/838/