semweb-2on1.pdf (application/pdf-Objekt) - Mozilla Firefox

Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe Farben Bilder JavaScript Flash Cookies Anpassen

http://www.dbis.informatik.uni-goettingen.de/Lectures/semweb-2on1.pdf

Meistbesuchte Seiten Erste Schritte Aktuelle Nachrichten

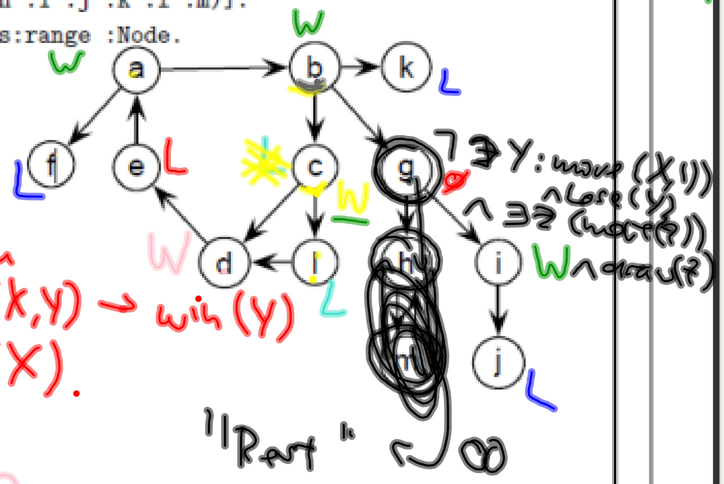
172 / 207 104% Suchen

1. Schritt: definitive Lose-Knoten

2. Schritt: definitiv gewonnen $\exists Y: move(X,Y) \wedge lose(Y)$

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix owl: <http://www.w3.org/2002/07/owl#>.
@prefix : <foo://bla/>.

:Node a owl:Class; owl:equivalentClass
  [ a owl:Class; owl:oneOf (:a :b :c :d :e :f :g :h :i :j :k :l :m)].
:edge a owl:ObjectProperty; rdfs:domain :Node; rdfs:range :Node.
:out a owl:DatatypeProperty.
:a a :Node; :out 2; :edge :b, :f.
:b a :Node; :out 3; :edge :c, :g, :k.
:c a :Node; :out 2; :edge :d, :l.
:d a :Node; :out 1; :edge :e.
:e a :Node; :out 1; :edge :a.
:f a :Node; :out 0.
:g a :Node; :out 2; :edge :i, :h.
:h a :Node; :out 1; :edge :m.
:i a :Node; :out 1; :edge :j.
:j a :Node; :out 0.
:k a :Node; :out 0.
:l a :Node; :out 1; :edge :d.
:m a :Node; :out 1; :edge :h.
```



3. Schritt: wenn $\forall y: move(X,Y) \rightarrow win(Y) \longrightarrow lose(X)$.

4. Schritt wie 2.

5. wie 3

[Filename: RDF/winmove-graph.n3]

6. wie 4.

7. wie 5.

$\Rightarrow$ well-founded Semantics

Polynomiell

① $\forall x\,(p(x) \rightarrow \exists y: q(x,y))$

$\forall x\,(conf(x) \rightarrow \exists y: capital(x,y))$

② $\forall x\,\exists y: (p(x) \rightarrow q(x,y))$

$\forall x\,\exists y: (conf(x) \rightarrow cap(x,y))$

D Berlin

Joe 3
Esel

falls conf(Joe) $\rightarrow$ ......

$\neg$ ... $\rightarrow$

Leichsinnig

etwas komplizierter wg. Joe

$\boxed{1 \models 2}$

$2 \models 1$

wird gezeigt.

$1 \rightarrow 2$

d.h. aus 1 folgt 2

Meta ebene

aus dem Erfülltsein von 1 folgt, daß auch 2 erfüllt ist.

$\models$ p, q

d.h. formal (Def):

für alle Belegungen $\mathcal{M} = (\mathcal{D}, \mathcal{I})$ gilt...

wenn $\mathcal{M} \models 1$, dann auch $\mathcal{M} \models 2$

$\Longleftrightarrow$

$\emptyset \models \boxed{1 \rightarrow 2}$ allg. Formel

$\Rightarrow$ Tableau

allg. Formel

Tableau für

$① \to ②$ $\Rightarrow$ Tableau mit

$\neg(① \to ②)$

$\Rightarrow$ wollen zeigen, daß das nicht sein kann, d.h. $\square$

$1 \to 2 \mathrel{\hat{=}} \neg 1 \vee 2$

$\neg(\neg 1 \vee 2)$

$1 \wedge \neg 2$

$\neg 2$

$\neg \forall x \exists y (p(x) \to q(x,y))$

$\exists x \neg \exists y (p(x) \to q(x,y))$

$\exists x \ \forall y (p(x) \wedge \neg q(x,y))$

$\exists x: p(x) \wedge \forall y: \neg q(x,y)$

$\neg A \to B$

$A \wedge \neg B$

① $\forall x\,(p(x) \to \exists y: q(x,y)) \wedge \neg 2$

$\neg 2$: $\exists \cancel{x}\,(p(\cancel{x}) \wedge \forall \overset{w}{y}: \neg q(\cancel{x}, \overset{w}{y}))$

v_0

aus ②: $p(v_0) \wedge \forall w: \neg q(v_0, w)$

$p(v_0)$

$\forall w: \neg q(v_0, w)$

$\neg q(v_0, W)$

W: Tableauvariable
→ darf alles dafür einsetzen

$p(X) \to \exists y: q(X, y)$

$\neg p(X)$ $\qquad$ $\exists y: q(\overset{v_0}{X}, y)$

$\square$ $X \to v_0$ $\qquad$ $\exists y: q(v_0, y)$

verwende $f(v_0)$

$y \mapsto cap(v_0)$

$q(v_0, cap(v_0))$

$W / cap(v_0)$

$\square$

Formel (3):

F_1: $\forall x \left(p(x) \rightarrow \exists y\ q(x,y)\right)$

person(x) → ∃y: kennt(x, y)

wählt x

F_3: $\exists y: \left(\forall x \left(p(x) \rightarrow q(x,y)\right)\right)$

y: (wenn p eine Person ist, dann kennt p y)

$F_3 \models F_1$

$F_1 \not\models F_3$ wo liegt der Unterschied?

Baue

Stelle $F_1 \wedge \neg F_3$

Fn ∧ ¬F3

(1)

$$\forall x\ p(x) \rightarrow \exists y : q(x,y)$$

$$\neg\exists y : \forall x : p(x) \rightarrow q(x,y)$$

$$\forall y : \exists x\ p(x) \wedge \neg q(x,y)$$

$\hat{=}$

∃'

für jedes y gibt es ein X das Y nicht kennt

$$\exists x\ p(x) \wedge \neg q(x, y)$$

bezeichne als $c_0(y)$ ✓

$$\cancel{p(c_0(y))} \wedge \neg q(c_0(y), y)$$

joe

mary

$$p(X) \rightarrow \exists y : q(X,y)$$

$$\neg p(X)$$

$$X \rightarrow c_0(y)$$

joe

$$\exists y\ q(X,y)$$

$$q(X, f_1(X))$$

$p(\text{mary}) = \text{alice}$

$$q(c_0(y), f_1(c_0(y)))$$

c_0 kennt jemanden anderes, als das alternativ y.

Forts. von Folie №6

⋮

italian(c)

¬lazy(c)

X latinlover(c)

6: $\forall x$: latinlover(x) -> gentleman(x)

latinlover(X_2) -> gentleman(X_2)

¬latinlover(X_2) gentleman(X_2)

X □ X_2 -> c gentleman(c)

7

X_3

□ X_3 -> c english(c)

2 $\forall x$ english(x) -> ¬italian(X)

english(X_4) -> ¬italian(X_4)

□ X_4 -> c ¬english(X_4) ¬italian(X_4) c

□