

Uitwerkingen van de oefenversie van de tussentoets

Brammert Ottens
bottens@science.uva.nl

March 23, 2005

Er was door sommige van jullie gevraagd of er uitwerkingen van de oefentamens op het net konden komen. Ik heb niet alle opdrachten uitgewerkt, maar het zo gedaan dat jullie een idee hebben van wat er verwacht wordt op het tentamen. Vraag 2 is uitgebreid behandeld op college en vraag 6 staat in het boek dus heb ik die weg gelaten.

1. (a) Give a backward chaining proof of $L(P(5, 1), P(P(1, 1), 8))$.

Het is bij deze opdracht niet nodig om het hele algoritme door te lopen. Het volstaat om bij iedere functie aanroep op te schrijven wat de goals zijn, welke regel wordt aangeroepen en wat de benodigde substitutie is. Bovendien mag je telkens de goede regel kiezen.

i. goals = $[L(P(5, 1), P(P(1, 1), 8))]$
 $\theta = []$
 $q' = L(P(5, 1), P(P(1, 1), 8))$
pick rule H: $L(x, y) \wedge L(y, z) \rightarrow L(x, z)$

$\theta' = [x/P(5, 1), z/P(P(1, 1), 8)]$

ii. goals = $[L(x, y), L(y, z)]$
 $\theta = [x/P(5, 1), y/P(P(1, 1), 8)]$
 $q' = L(P(5, 1), z)$
pick rule F: $L(P(x_1, y_1), P(y_1, x_1))$
 $\theta' = [x_1/5, y_1/1, y/P(1, 5)]$

iii. goals = $[L(y, z)]$
 $\theta = [x/P(5, 1), y/P(P(1, 1), 8), x_1/5, y_1/1, y/P(1, 5)]$
 $q' = L(P(1, 5), P(P(1, 1), 8))$
pick rule G : $L(x_2, y_2) \wedge L(w_2, z_2) \rightarrow L(P(x_2, w_2), P(y_2, z_2))$
 $\theta' = [x_2/1, w_2/5, y_2/P(1, 1), z_2/8]$

iv. goals = $[L(x_2, y_2), L(w_2, z_2)]$
 $\theta = [x/P(5, 1), y/P(P(1, 1), 8), x_1/5, y_1/1, y/P(1, 5), x_2/1, w_2/5, y_2/P(1, 1), z_2/8]$

- $q' = L(1, P(1, 1))$
 pick rule H: $L(x_3, y_3) \wedge L(y_3, z_3) \rightarrow L(x_3, z_3)$
 $\theta' = [x_3/1, z_3/P(1, 1)]$
- v. goals = $[L(x_3, y_3), L(y_3, z_3), L(w_2, z_2)]$
 $\theta = [x/P(5, 1), y/P(P(1, 1), 8), x_1/5, y_1/1, y/P(1, 5), x_2/1, w_2/5, y_2/P(1, 1), z_2/8, x_3/1, z_3/P(1, 1)]$
 $q' = L(1, y_3)$
 pick rule D: $L(x_4, P(x_4, 0))$
 $\theta' = [x_4/1, y_3/P(1, 0)]$
- vi. goals = $[L(y_3, z_3), L(w_2, z_2)]$
 $\theta = [x/P(5, 1), y/P(P(1, 1), 8), x_1/5, y_1/1, y/P(1, 5), x_2/1, w_2/5, y_2/P(1, 1), z_2/8, x_3/1, z_3/P(1, 1), x_4/1, y_3/P(1, 0)]$
 $q' = L(P(1, 0), P(1, 1))$
 pick rule G: $L(x_5, y_5) \wedge L(w_5, z_5) \rightarrow L(P(x_5, w_5), P(y_5, z_5))$
 $\theta' = [x_5/1, w_5/0, y_5/1, z_5/1]$
- vii. goals = $[L(x_5, y_5), L(w_5, z_5), L(w_2, z_2)]$
 $\theta = [x/P(5, 1), y/P(P(1, 1), 8), x_1/5, y_1/1, y/P(1, 5), x_2/1, w_2/5, y_2/P(1, 1), z_2/8, x_3/1, z_3/P(1, 1), x_4/1, y_3/P(1, 0), x_5/1, w_5/0, y_5/1, z_5/1]$
 $q' = L(1, 1)$
 pick rule C: $L(x_6, x_6)$
 $\theta' = [x_6/1]$
- viii. goals = $[L(w_5, z_5), L(w_2, z_2)]$
 $\theta = [x/P(5, 1), y/P(P(1, 1), 8), x_1/5, y_1/1, y/P(1, 5), x_2/1, w_2/5, y_2/P(1, 1), z_2/8, x_3/1, z_3/P(1, 1), x_4/1, y_3/P(1, 0), x_5/1, w_5/0, y_5/1, z_5/1, x_6/1]$
 $q' = L(0, 1)$
 pick rule A: $L(0, 1)$
 $\theta' = []$
- ix. goals = $[L(w_2, z_2)]$
 $\theta = [x/P(5, 1), y/P(P(1, 1), 8), x_1/5, y_1/1, y/P(1, 5), x_2/1, w_2/5, y_2/P(1, 1), z_2/8, x_3/1, z_3/P(1, 1), x_4/1, y_3/P(1, 0), x_5/1, w_5/0, y_5/1, z_5/1, x_6/1]$
 $q' = L(5, 8)$
 pick rule B: $L(5, 8)$
 $\theta' = []$

2.

3. (a) α is valid $\rightarrow \neg\alpha$ is unsatisfiable.

Dit kan bewezen worden met een zogenaamd bewijs uit het ongerijmde. De bedoeling hiervan is dat men aanneemt dat de implicatie onwaar is, en daaruit een tegenstelling afleid.

In dit geval nemen we dus aan dat $\forall M, M \models \alpha$, maar ook $\exists M', M' \models \neg\alpha$. We kunnen nu meteen al een contradictie afleiden. Het is

namelijk niet mogelijk hebben om een formule te hebben die satisfiable is in alle modellen en tegelijkertijd ook een model te kunnen vinden waarin de negatie van deze formule waar is. Onze aanname is dus ongeldig en de implicatie waar.

(b) α is satisfiable $\rightarrow \alpha$ is valid.

In dit geval is er een tegenvoorbeeld. Namelijk het volgende. Neem een model waarin α waar is. De linkerkant van de implicatie is nu waar gemaakt. Neem nu een model M' waarvoor geldt dat $M \models \neg\alpha$. De rechterkant van de implicatie is nu ongeldig, ofwel de implicatie is niet geldig.

$$\begin{aligned}
4. \quad a \quad & \neg((p \rightarrow q) \rightarrow ((q \rightarrow t) \rightarrow (p \rightarrow t))) \Leftrightarrow \\
& \neg(\neg(\neg p \vee q) \vee (\neg(\neg q \vee t) \vee (\neg p \vee t))) \Leftrightarrow \\
& \neg((p \wedge \neg q) \vee (q \wedge \neg t) \vee \neg p \vee t) \Leftrightarrow \\
& ((\neg p \vee q) \wedge (\neg q \vee t) \wedge p \wedge \neg t)
\end{aligned}$$

Hieruit kunnen we de volgende set van clauses afleiden

$\{\{\neg p, q\}, \{\neg q, t\}, \{p\}, \{\neg t\}\}$. Uit $\{p\}$ en $\{\neg p, q\}$ kunnen we $\{q\}$ afleiden en uit $\{\neg t\}$ en $\{\neg q, t\}$ kunnen we $\{\neg q\}$ afleiden. Uit $\{q\}$ en $\{\neg q\}$ kunnen we $\{\}$ afleiden en hebben we dus bewezen dat $\neg((p \rightarrow q) \rightarrow ((q \rightarrow t) \rightarrow (p \rightarrow t)))$ niet satisfiable is, oftewel $(p \rightarrow q) \rightarrow ((q \rightarrow t) \rightarrow (p \rightarrow t))$ is altijd waar en is dus een tautologie.

$$\begin{aligned}
c \quad & \neg((p \vee q) \Leftrightarrow (p \Rightarrow q)) \\
& \Leftrightarrow \\
& \neg(((p \vee q) \Rightarrow (p \Rightarrow q)) \wedge ((p \vee q) \Leftarrow (p \Rightarrow q))) \Leftrightarrow \\
& \neg((p \vee q) \Rightarrow (p \Rightarrow q)) \vee \neg((p \vee q) \Leftarrow (p \Rightarrow q)) \Leftrightarrow \\
& ((p \vee q) \wedge (p \wedge \neg q)) \vee ((\neg p \wedge \neg q) \wedge (\neg p \vee q)) \Leftrightarrow \\
& ((p \vee q) \wedge p \wedge \neg q) \vee (\neg p \wedge \neg q \wedge (\neg p \vee q)) \Leftrightarrow \\
& (((p \wedge p) \vee (q \wedge p)) \wedge \neg q) \vee (\neg p \wedge ((\neg q \wedge \neg p) \vee (\neg q \wedge q))) \Leftrightarrow \\
& (p \wedge \neg q) \vee (\neg p \wedge \neg q) \Leftrightarrow \\
& ((p \vee (\neg p \wedge \neg q)) \wedge (\neg q \vee (\neg p \wedge \neg q))) \Leftrightarrow \\
& (p \vee \neg p) \wedge (p \vee \neg q) \wedge (\neg p \vee \neg q) \wedge (\neg q \vee \neg q) \Leftrightarrow \\
& (p \vee \neg q) \wedge (\neg p \vee \neg q) \wedge \neg q
\end{aligned}$$

Zoals je kan zien kunnen er geen nieuwe clauses gemaakt worden m.b.v. resolutie. Dus de $RC(\neg((p \vee q) \Leftrightarrow (p \Rightarrow q))) = \{\{p, \neg q\}, \{\neg p, \neg q\}, \{\neg q\}\}$. Nu moet q wel false zijn, en blijven er 2 mogelijkheden over voor p . Kies p is true. We hebben nu het volgende model: $[p/1, q/0]$. Als we nu eens gaan kijken naar de formule zien we dat $p \vee q$ nu waar is, maar $p \Rightarrow q$ onwaar. De expressie $(p \vee q) \Leftrightarrow (p \Rightarrow q)$ is dus onwaar en de negatie daarvan weer waar. Het gevonden model is dus een model van onze zin.

5. In het resolutie algoritme op de handout kan de zin "if resolvents contains the empty clause then return true" vervangen worden door "if resolvents is just the empty clause the return true". Dit is heel makkelijk te zien. Als je naar de resolutie regel kijkt, zie je dat je per keer maar een symbool mag wegstrepen. Als resolvents de lege clause bevat betekent dit dus

dat beide originele clauses uit maar een element bestonden en er kan dus op geen enkele andere manier resolutie uitgevoerd worden. De lijst resolvents zal dus ook altijd uit alleen maar de lege lijst bestaan.