Fakultät für Informatik
Übungen zu Kognitive Systeme
Sommersemester 2017

Yoo-Jin Azad (yoo-jin.jeong@kit.edu)
Daniel Reichard (reichard@kit.edu)

Musterlösung zu Übungsblatt 6

Wissensrepräsentation, Bahnplanung und Robotik

Aufgabe 6.1 Resolution und DPLL

6.1.a

Jeder Satz läßt sich, soweit er erfüllbar ist, durch eine Wahrheitstabelle darstellen. Daraus gewinnt man eine KF wie folgt:
Alle Zeilen deren Gesamtwahrheitswert FALSE ist, werden konjunktiv verknüpft. Die Konjunktionen selbst bestehen dabei aus den disjunktiv verknüpften und negiert oder nicht negierten Variablen aus der Tabelle.
Für die Sätze α, β und γ ergibt sich folgende Wertetabelle:

α	β	γ	A	B	C
1	0	1	0	0	0
1	1	0	0	0	1
1	0	1	0	1	0
1	1	0	0	1	1
0	0	1	1	0	0
0	0	0	1	0	1
0	1	1	1	1	0
1	1	1	1	1	1

Daraus ergibt sich die KF für α, β und γ:

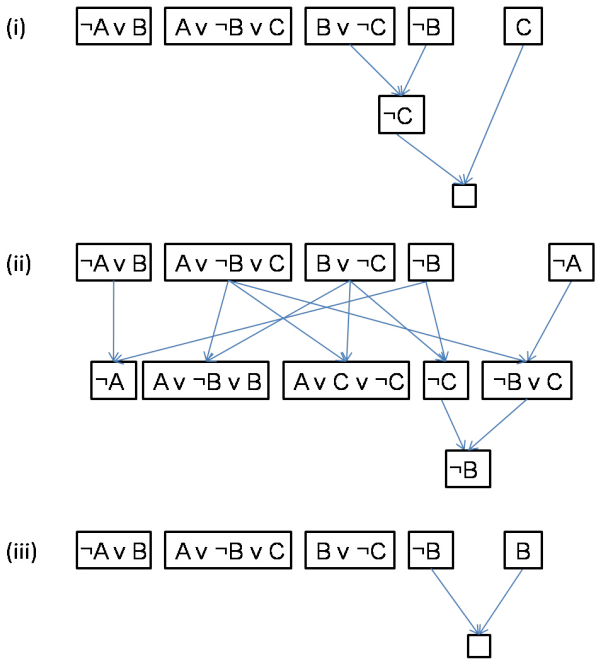
$$\alpha_{KF} = (\neg A \vee B \vee C) \wedge (\neg A \vee B \vee \neg C) \wedge (\neg A \vee \neg B \vee C)$$

$$\beta_{KF} = (A \vee B \vee C) \wedge (A \vee \neg B \vee C) \wedge (\neg A \vee B \vee C) \wedge (\neg A \vee B \vee \neg C)$$

$$\gamma_{KF} = (A \vee B \vee \neg C) \wedge (A \vee \neg B \vee \neg C) \wedge (\neg A \vee B \vee \neg C)$$

6.1.b

Onlinefrage Nr. 1:
A kann nicht abgeleitet werden.



6.1.c

Symbol A kommt in einer Einheitsklausel vor. Setze A=TRUE. Dies ergibt

$$\begin{array}{lllll} B & \vee & \neg C & & \\ \neg C & \vee & \neg B & \vee & E \\ \neg D & \vee & \neg B & \vee & C \\ \neg E & \vee & \neg D & \vee & B \\ B & \vee & E & \vee & C \end{array}$$

Keine Einheitsklauseln. Das Symbol D kommt nur negiert vor, ist also ein reines Symbol. Wähle deshalb D=FALSE. Dies ergibt:

$$\begin{array}{lllll} B & \vee & \neg C & & \\ \neg C & \vee & \neg B & \vee & E \\ B & \vee & E & \vee & C \end{array}$$

Keine Einheitsklauseln. Das Symbol E kommt nur nicht negiert vor, ist also ein reines Symbol. Wähle deshalb E=TRUE. Dies ergibt:

$$B \quad \vee \quad \neg C$$

Das Symbol B (oder C) kommt nur (nicht) negiert vor, ist also ein reines Symbol. Wähle deshalb B=TRUE. Dadurch sind alle Klauseln erfüllt.
Ein mögliches Modell lautet somit

$$M_1 = \{\{A = TRUE\}, \{B = TRUE\}, \{C = TRUE \vee FALSE\}\{D = FALSE\}, \{E = TRUE\}\}.$$

Alternativ kann C=FALSE gewählt werden. Auch hierdurch sind alle Klauseln erfüllt. Ein weiteres mögliches Modell lautet somit

$M_2 = \{\{A = TRUE\}, \{B = TRUE \vee FALSE\}, \{C = FALSE\}\{D = FALSE\}, \{E = TRUE\}\}.$

Da ein Modell für die Klauselmenge gefunden wurde, ist sie also erfüllbar.
Bemerkung: Ein weiteres Modell, welches die Klauselmenge erfüllt und dabei keine Aussagen über E macht, lautet

$M_3 = \{\{A = TRUE\}, \{B = TRUE\}, \{C = FALSE\}\{D = FALSE\}, \{E = TRUE \vee FALSE\}\}.$

Dieses Modell wird jedoch durch den DPLL-Algorithmus nicht gefunden. Beachte: Der DPLL-Algorithmus prüft nur die (Un-)Erfüllbarkeit einer Klauselmenge. Er findet aber nicht alle Modelle, die diese Klauselmenge erfüllen.
Onlinefrage Nr. 2:
Es existiert sowohl für $C = TRUE$ ***als auch*** $C = FALSE$ ***ein Modell, das*** S ***erfüllt, und das vom DPLL-Algorithmus gefunden wird.***

Aufgabe 6.2 - STRIPS & ADL

6.2.a

$Action\big(Load(c,p,a),$
$Vorbed : At(c,a) \wedge At(p,a) \wedge Fracht(c) \wedge Flugzeug(p) \wedge Flughafen(a)$
$Effekt : \neg At(c,a) \wedge In(c,p)\big)$
$Action\big(Unload(c,p,a),$
$Vorbed : In(c,p) \wedge At(p,a) \wedge Fracht(c) \wedge Flugzeug(p) \wedge Flughafen(a)$
$Effekt : At(c,a) \wedge \neg In(c,p)\big)$
$Action\big(Fly(p,from,to),$
$Vorbed : At(p,from) \wedge Flugzeug(p) \wedge Flughafen(from) \wedge Flughafen(to)$
$Effekt : \neg At(p,from) \wedge At(p,to)\big)$

6.2.b

$Init\big(At(C_1,FRA) \wedge At(C_2,JFK) \wedge At(P_1,FRA) \wedge Fracht(C_1) \wedge Fracht(C_2) \wedge Flugzeug(P_1) \wedge Flughafen(FRA) \wedge Flughafen(JFK)\big)$

$Ziel\big(At(C_1,JFK) \wedge At(C_2,FRA)\big)$
Diese Aktionsfolge ist eine Lösung des Planungsproblems:
$[Load(C_1,P_1,FRA), Fly(P_1,FRA,JFK), Unload(C_1,P_1,JFK),$
$\quad Load(C_2,P_1,JFK), Fly(P_1,JFK,FRA), Unload(C_2,P_1,FRA)]$

6.2.c

$Init\big(At(C_1,FRA) \wedge At(C_2,JFK) \wedge At(P_1,FRA) \wedge Fracht(C_1) \wedge Fracht(C_2) \wedge Flugzeug(P_1) \wedge Flughafen(FRA) \wedge Flughafen(JFK)\big)$

$Ziel\big(\exists x At(C_1, x) \wedge At(C_2, x)\big)$
Diese Aktionsfolge ist eine Lösung des Planungsproblems:
$[Load(C_1, P_1, FRA), Fly(P_1, FRA, JFK), Unload(C_1, P_1, JFK)]$

Aufgabe 6.3 - Planungsgraph

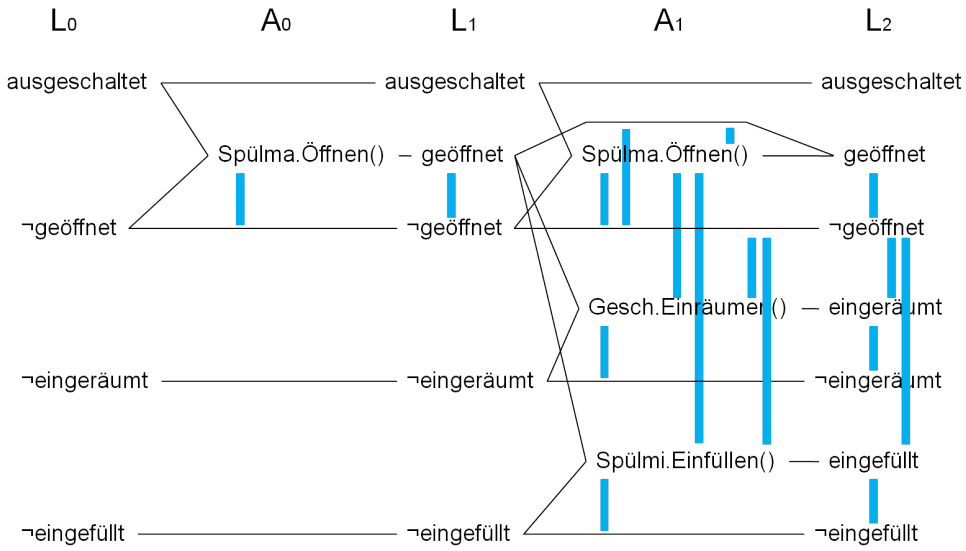
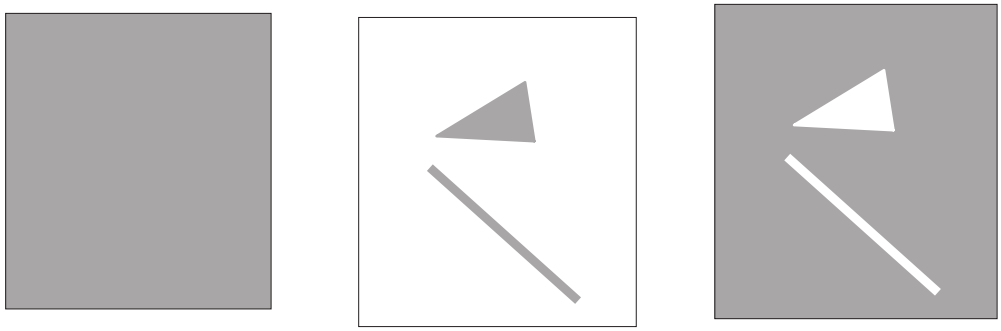


Abbildung 1: Planungsgraph zu Aufgabe 6.3

Aufgabe 6.4 - Bahnplanung

6.4.a

Die entsprechenden Räume sind in Abb. 2 jeweils in grau zu sehen. Der Konfigurationsraum besteht aus allen Punkten die für den Roboter in der Welt erreichbar sind. In diesem Falle also aus allen Punkten innerhalb der gegebenen quadratischen Welt (Abb. 2 a). Der Hindernissraum besteht aus den Konfigurationen, bei denen eine Berührung oder eine Durchdringung des Roboters mit dem Hindernis vorläge (Abb. 2 b). Der Freiraum besteht aus dem Konfigurationsraum ohne den Hindernisraum (Abb. 2 c). Der angegebene Raum ist jeweils in grau dargestellt. Zur Darstellung des Roboters als beweglichen Punkt können alle Hindernisse um den Roboterradius $\frac{d}{2}$ erweitert werden und auf diesem Modell Frei- und Hindernisraum berechnet werden.



(a) Konfigurationsraum (b) Hindernisraum (c) Freiraum

Abbildung 2: Räume zur Bahnplanung

6.4.b

Im Sichtgraph sind die Verbindungen aller sichtbaren Eckpunkte der Hindernisse (und des Start- und Zielpunktes) als mögliche Pfade dargestellt (Abb. 3). Auf den hierdurch

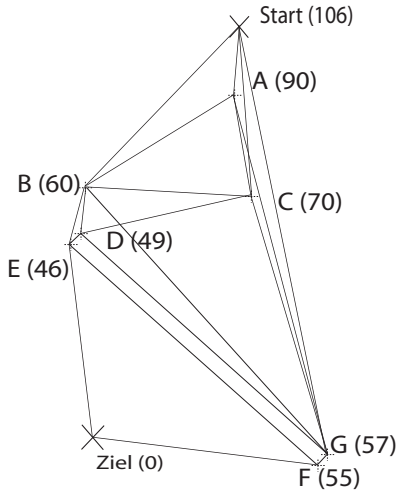


Abbildung 3: Sichtgraph

ermittelten Pfaden kann sich der mobile Roboter kollisionsfrei bewegen.
Onlinefrage Nr. 3:
Der Graph besitzt 19 Kanten.

6.4.c

Planer 1 verwendet den Greedy-Algoritmus. Er ermittelt einen Weg über Start-G-D-E-Ziel mit einer Gesamtlänge von 236 (siehe Abb.4). Planer 2, der den A*-Algorithmus anwendet, ermittelt einen Weg über Start-B-E-Ziel mit einer Gesamtlänge von 115 (siehe Abb. 5). Mit * markierte Knoten werden nicht weiterverfolgt, da sie bereits weiter oben im Baum

durchlaufen wurden.

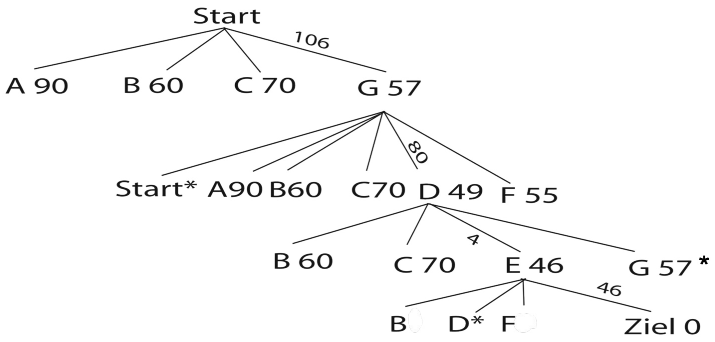


Abbildung 4: Suchbaum Planer 1

Onlinefrage Nr. 4:
Die Wegstrecke beträgt 115 Einheiten.

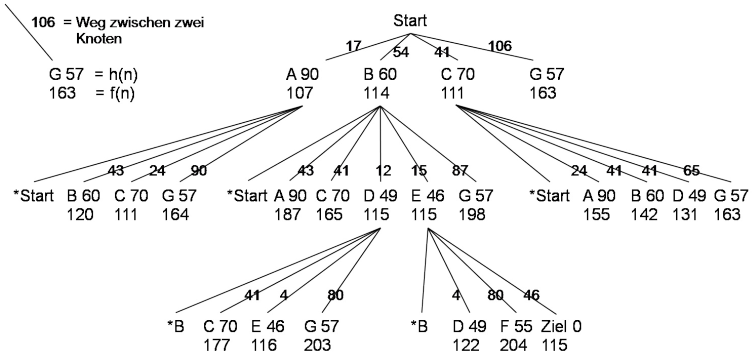


Abbildung 5: Suchbaum Planer 2

6.4.d

Siehe Abbildung 6.

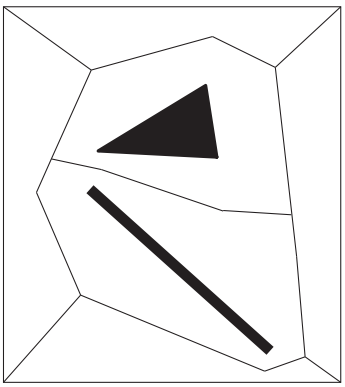


Abbildung 6: Voronoi-Diagramm

Aufgabe 6.5 - Quaternionen

6.5.a

Mit $\bar{q}_1 = (2, (-4, 3, 0))$ und $|q_1|^2 = 4 + 16 + 9 = 29$ errechnet sich das inverse Quaternion q_1^{-1} zu:

$$q_1^{-1} = \frac{\bar{q}_1}{|q_1|^2} = \left(\frac{2}{29}, (-\frac{4}{29}, \frac{3}{29}, 0)\right)$$

6.5.b

Mit $\vec{x}$ als Quaternion $\vec{x} = (0, (0, 0, 3))$ und $\bar{q}_2 = (\frac{\sqrt{2}}{2}, (-\frac{\sqrt{2}}{2}, 0, 0))$ berechnet sich der rotierte Punkt $\vec{x'}$ wie folgt:

$$\begin{aligned}
\vec{x'} &= q_2 x \bar{q}_2 \\
&= (\frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2}i) \cdot 3k \cdot (\frac{\sqrt{2}}{2} - \frac{\sqrt{2}}{2}i) \\
&= (\frac{3\sqrt{2}}{2}k - \frac{3\sqrt{2}}{2}j) \cdot (\frac{\sqrt{2}}{2} - \frac{\sqrt{2}}{2}i) \\
&= \frac{3}{2}k - \frac{3}{2}j - \frac{3}{2}j - \frac{3}{2}k \\
&= -3j \\
\Rightarrow \vec{x'} &= (0, -3, 0)
\end{aligned}$$

6.5.c

Mit Quaternionen lassen sich nur Rotationen darstellen, keine Translationen. Dies ist ein großer Nachteil von Quaternionen.
Onlinefrage Nr. 5: Nur Rotationen.