Universität Karlsruhe (TH)
Forschungsuniversität · gegründet 1825

Institut für Anthropomatik (IFA)
Lehrstuhl Prof. Dr.-Ing. Rüdiger Dillmann

Robotik II SS 2009

Musterlösungen zum 2. Übungsblatt

Prof. Dr.-Ing. R. Dillmann
Haid-und-Neu-Str. 7, Geb. 07.21
D-76131 Karlsruhe

Dipl. Inform. Sven R. Schmidt-Rohr
Dipl. Inform. Rainer Jäkel
http://wwwiaim.ira.uka.de

Lösung 1 (Planungsverfahren)

Objekte: Kiste, Kühlschrank, Flasche, Roboterhand, Kühlschrank_Auf, Neben_Kühlschrank, Besitzerhand

Aktionen: Greife_Von(x, y), Bewege(x, y)
Prädikate: Frei(x), Inhand(x), Blockiert(x, y)

a) Die Aktionen lauten (Prädikate aus der Delete-Liste sind mit ¬ gekennzeichnet, die positiven sind aus der Add-Liste):

$Greife(x)$
P: $Frei(x) \wedge Frei(\text{Roboterhand})$
E: $\neg\, Frei(x) \wedge \neg\, Frei(\text{Roboterhand}) \wedge Inhand(x)$

$Greife\_Von(x, y)$
P: $Frei(x) \wedge Frei(\text{Roboterhand}) \wedge Blockiert(x, y)$
E: $\neg\, Frei(x) \wedge \neg\, Frei(\text{Roboterhand}) \wedge Inhand(x) \wedge Frei(y) \wedge \neg\, Blockiert(x, y)$

$Bewege(x, y)$
P: $Inhand(x) \wedge Frei(y)$
E: $Frei(x) \wedge \neg\, Inhand(x) \wedge Frei(\text{Roboterhand}) \wedge \neg\, Frei(y) \wedge Blockiert(x, y)$

b) Der Startzustand lautet:

Start: $Frei(\text{Kiste}) \wedge Frei(\text{Roboterhand}) \wedge Blockiert(\text{Kiste, Kühlschrank}) \wedge Blockiert(\text{Kühlschrank, Flasche}) \wedge Frei(\text{Neben\_Kühlschrank}) \wedge Frei(\text{Kühlschrank\_Auf}) \wedge Frei(\text{Besitzerhand})$

c) Das Ziel kann modelliert werden als:

Ziel: Blockiert(Flasche, Besitzerhand)

d) Ein effizienter Plan wäre:

ZUSTAND *Frei*(Kiste) ∧ *Frei*(Roboterhand) ∧ *Blockiert*(Kiste, Kühlschrank) ∧ *Blockiert*(Kühlschrank, Flasche) ∧ *Frei*(Neben_Kühlschrank) ∧ *Frei*(Kühlschrank_Auf) ∧ *Frei*(Besitzerhand)
AKTION Greife_Von(Kiste, Kühlschrank)
ZUSTAND *Inhand*(Kiste) ∧ *Frei*(Kühlschrank) ∧ *Blockiert*(Kühlschrank, Flasche) ∧ *Frei*(Neben_Kühlschrank) ∧ *Frei*(Kühlschrank_Auf) ∧ *Frei*(Besitzerhand)
AKTION Bewege(Kiste, Neben_Kühlschrank)
ZUSTAND *Frei*(Roboterhand) ∧ *Frei*(Kiste) ∧ *Blockiert*(Kiste, Neben_Kühlschrank) ∧ *Frei*(Kühlschrank) ∧ *Blockiert*(Kühlschrank, Flasche) ∧ *Frei*(Kühlschrank_Auf) ∧ *Frei*(Besitzerhand)
AKTION Greife_Von(Kühlschrank, Flasche)
ZUSTAND *Inhand*(Kühlschrank) ∧ *Frei*(Flasche) ∧ *Frei*(Kiste) ∧ *Blockiert*(Kiste, Neben_Kühlschrank) ∧ *Frei*(Kühlschrank_Auf) ∧ *Frei*(Besitzerhand)
AKTION Bewege(Kühlschrank, Kühlschrank_Auf)
ZUSTAND *Frei*(Roboterhand) ∧ *Frei*(Kühlschrank) ∧ *Blockiert*(Kühlschrank, Kühlschrank_Auf) ∧ *Frei*(Kiste) ∧ *Blockiert*(Kiste, Neben_Kühlschrank) ∧ *Frei*(Flasche) ∧ *Frei*(Besitzerhand)
AKTION Greife(Flasche)
ZUSTAND *Inhand*(Flasche) ∧ *Frei*(Kühlschrank) ∧ *Blockiert*(Kühlschrank, Kühlschrank_Auf) ∧ *Frei*(Kiste) ∧ *Blockiert*(Kiste, Neben_Kühlschrank) ∧ *Frei*(Besitzerhand)
AKTION Bewege(Flasche, Besitzerhand)
ZUSTAND *Frei*(Roboterhand) ∧ *Frei*(Flasche) ∧ *Blockiert*(Flasche, Besitzerhand) ∧ *Frei*(Kühlschrank) ∧ *Blockiert*(Kühlschrank, Kühlschrank_Auf) ∧ *Frei*(Kiste) ∧ *Blockiert*(Kiste, Neben_Kühlschrank)

e) Ein mechanischer Planer kann Probleme dabei kriegen, Greife() und Greife_Von() korrekt anzuwenden. Außerdem können Zyklen entstehen, wobei die Kiste zwischen Neben_Kühlschrank, Kühlschrank und Kühlschrank_Auf hin- und hergeschoben werden kann.

f) Die Semantik von Kühlschrank_Auf ist weiterhin problematisch.

Dies sind typische Probleme der STRIPS-Notation im praktischen Gebrauch.

Lösung 2 (Planungsverfahren II)

a) Abbildung 1 zeigt den Vorranggraph.

b) Abbildung 2 zeigt den teilweise linearisierten Vorranggraph für 2 Roboterhände. Das Decken von Teller muss an einer der markierten Stellen geschehen. Damit ist der Plan etwas langsamer, als er es für 3 Roboterhände wäre.

c) Abbildung 3 zeigt den Vorranggraph. Auch der Serviceroboter mit den klobigeren Händen wird aber bei der Ausführung genauso lange brauchen, wie der Roboter aus a).

d) Abbildung 4 zeigt den HTN-Baum Alle Knoten auf Ebene 2 können parallel bearbeitet werden. Abstrakte Aktionen sind mit A: gekennzeichnet.

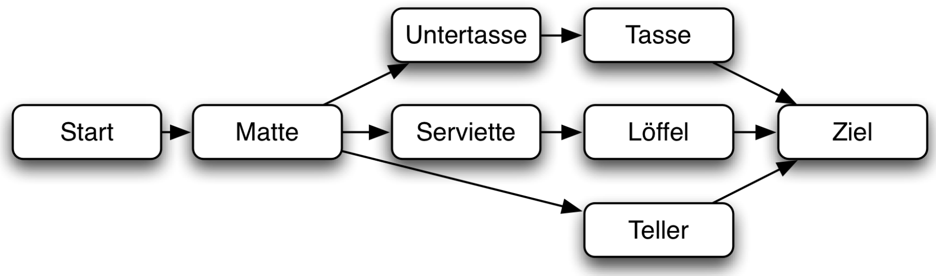

Abbildung 1: Der Vorranggraph aus a).

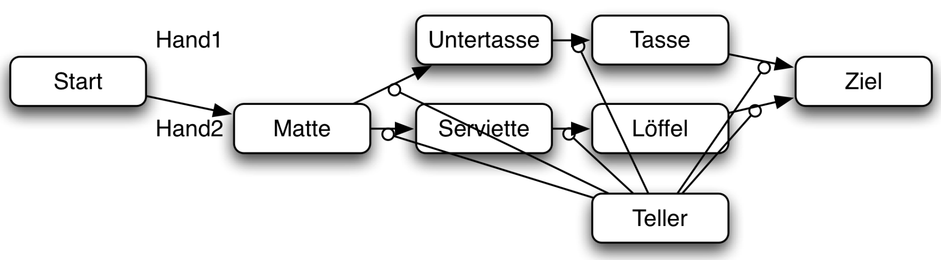

Abbildung 2: Der Vorranggraph aus b).

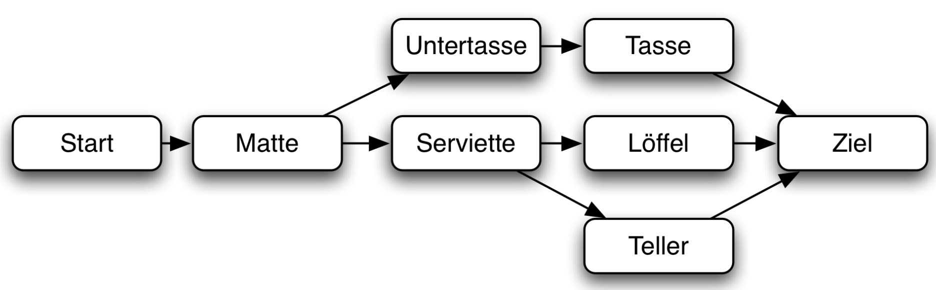

Abbildung 3: Der Vorranggraph aus c).

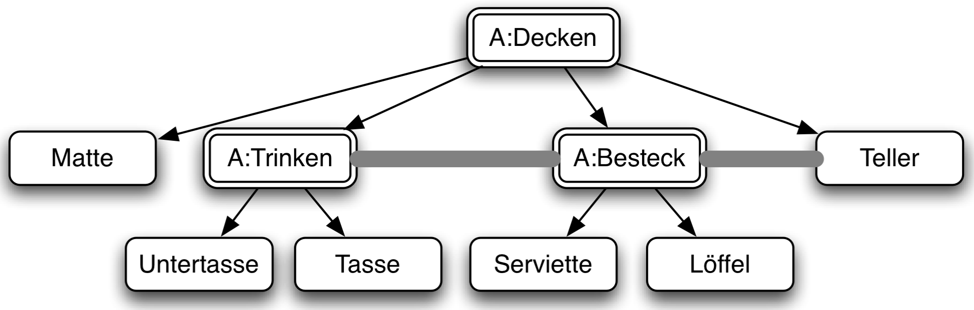

Abbildung 4: Der HTN aus d).

e) Abbildung 5 zeigt den abstrakten HTN-Baum. HTNs ermöglichen die einfache Erweiterung bzw. Synthese von grossen Szenarios bei verschiedenen konkreten Suchmethoden zur Planung. Sie sind also besonders modellierungsfreundlich.

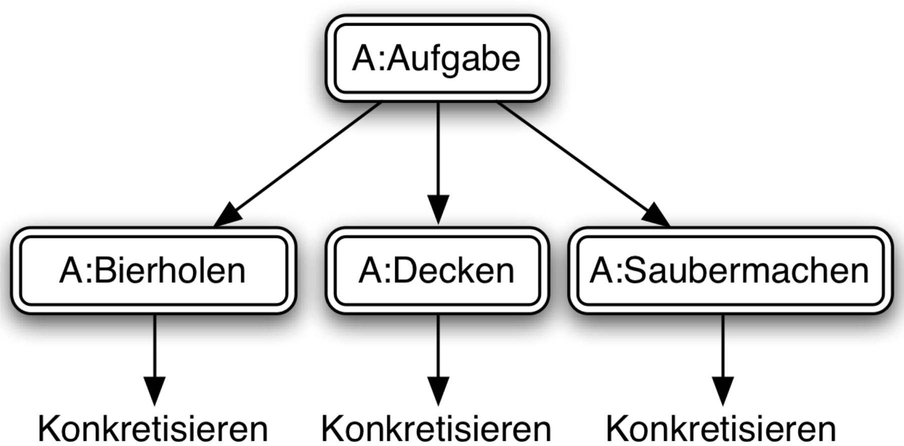

Abbildung 5: Der HTN aus e).

Lösung 3 (Markov Decision Processes)

a) Siehe Pseudocode des Übungsblattes.

b) Die value function für $\gamma = 0.8, 0.9, 0.95$, jeweils die ersten 10, der 20., 30. und 40. Schritte.

$\gamma = 0.8$:
-0.632 0.5648 3.56576
-0.26624 1.45242 4.09714
0.130739 1.94877 4.46148
0.468197 2.26664 4.7329
0.734246 2.49307 4.93889
0.939161 2.66195 5.09583
1.09589 2.78989 5.2155
1.21551 2.88728 5.30676
1.30676 2.96151 5.37636
1.37636 3.01812 5.42944

1.58511 3.18789 5.58865

1.59901 3.19919 5.59924

1.59993 3.19995 5.59995

$\gamma = 0.9$:
-0.804938 0.774042 4.03576
-0.369173 1.9983 4.79154
0.200426 2.7839 5.39467
0.761666 3.36778 5.9143
1.27187 3.8499 6.36929
1.72474 4.2655 6.76896
2.12394 4.62879 7.12027
2.47514 4.94766 7.42909
2.78394 5.22786 7.70059
3.05543 5.47417 7.93926

4.48546 6.77147 9.19643

4.87974 7.12916 9.54305

4.99346 7.23232 9.64302

$\gamma = 0.95$:
-0.89968 0.885213 4.2689
-0.430784 2.30183 5.15592
0.23975 3.27028 5.9151
0.947514 4.0412 6.61469
1.63489 4.72275 7.26922

2.28651 5.35097 7.88339
2.9 5.93777 8.46002
3.47647 6.48799 9.00147
4.01788 7.00445 9.5099
4.5263 7.48936 9.98731

8.18208 10.976 13.4201

10.1305 12.8342 15.2497

11.1689 13.8246 16.2248

Bei $\gamma = 0.8$ ist schon nach 40 Schritten die optimale value function (1.6, 3.2, 5.6) klar zu erkennen, die anderen Grenzwerte bei 0.9 und 0.95 sind nach 40 Schritten noch nicht zu erkennen.

c) Es gilt in diesem Szenario immer $V(s_1) < V(s_2) < V(s_3)$ für $\gamma < 1$. Das liegt daran, dass in s_3 der große positive Gewinn eingesammelt werden kann und im Transitionsmodell durch keine Handlung ein Übergang von s_1 nach s_3 möglich ist, also immer s_2 zwischen s_1 und s_3 liegt und damit s_2 immer näher (wegen $\gamma < 1$ wichtig) an s_3 liegt, als s_1 an s_3 liegt.

d) Der relative Unterschied ist bei kleinerem γ immer größer, da hier die *Entfernung* zum einzigen positiven Gewinn eine größere Rolle spielt. Bei größerem γ spielt langfristig mehrfach eingesammelter Gewinn (wegen dem unendlichem Horizont unendlich oft) eine größere Rolle.

e) Die absolute Differenz ist mindestens 4 (der positive Gewinn), bei größerem γ nahe 1 aber auch etwas mehr.

f) Bei T_b und T_c konvergiert die value function gegen (0, 0, 4).

g) Da der Mensch in T_b viel seltener folgt, verbraucht der Roboter nun viel mehr Sprechkosten bei seinen Versuchen, während sein Gewinn aus dem erfolgreichen Führen in T_b gerade noch reicht, um langfristig einen positiven Reward zu erhalten.

h) In T_c ist es für den Roboter nun günstiger, gar nicht mehr zu versuchen, einen Menschen zum Folgen aufzufordern. Er bleibt anstatt dessen passiv und wird nun keinen positiven Gewinn mehr einfahren, was aber immer noch besser ist, als langfristig negative Kosten zu akkumulieren. Der numerische Zusammenhang ist einfach: 3 erfolglose Versuche, den Mensch zum Folgen zu bewegen gibt $(3 * -1) = -3$, während $(4 + 1 * -1) = +3$ einen erfolgreichen Versuch beschreibt, nämlich die Kosten des Sprechens und die Belohnung des erfolgreichen Führens.

$T_x(S', U, S)$:

$$
U_1 : \begin{array}{cccc}
 & S'_1 & S'_2 & S'_3 \\
S_1 : & 0.75 & 0.25 & 0.0 \\
S_2 : & 0.0 & \mathbf{0.75} & \mathbf{0.25} \\
S_3 : & 0.25 & 0.25 & 0.5
\end{array}
$$

Bei T_x ist dadurch genau das Gleichgewicht erreicht - statistisch folgt auf 3 erfolglose Versuche ein Erfolgreicher. Bei T_b ist es für den Roboter gerade noch lohnenswert, Versuche zu unternehmen, bei T_c bleibt er lieber passiv.

i) Das Belohnungsmodell kann einfach geändert werden, indem die Belohnung für den Versuch, den Menschen erfolgreich zu führen auf 6 (-1 für die Kosten gibt 5 für den Eintrag) erhöht wird.

Lösung 4 (Partially Observable Markov Decision Processes)

a) Das Beobachtungsmodell lautet:

$O_a(m, s')$:

	S'_1	S'_2	S'_3
M_1 :	0.7	0.1	0.0
M_2 :	0.3	0.6	0.2
M_3 :	0.0	0.3	0.8

Als Bias ist sichtbar, dass der Roboter in diesem Beispiel erkennungstechnisch false positives bevorzugt, so sieht er eher nicht vorhandene Menschen, als dass er tatsächlich anwesende übersieht und er glaubt eher fälschlich an die Intention zu Folgen, als umgekehrt. Dies zeigt sich auch an den Zeilensummen von 0.8, 1.1, 1.1. Das Modell zeigt außerdem, dass die korrekte Erkennung des tatsächlichen Zustands überwiegt (Werte in der Diagonale).

b) Bei den vorliegenden linearen Funktionen Γ fällt auf, dass zur gleichen Handlung gehörende α ähnliche Werte haben. Diese Gruppen von ähnlichen Funktionen sind ein typisches Charakteristikum von value iteration POMDP policies. Es kann bei komplexeren Modellen auch sehr unterschiedliche Funktionen mit der gleichen Handlung geben, aber auch diese werden meistens jeweils in Gruppen anzutreffen sein. Dies liegt daran, dass solche Funktionen alle ein gleiches Handlungs/Geschehnismuster in der nahen Zukunft repräsentieren, während sie sich erst zu einem späteren Zeitpunkt in der projizierten Zukunft unterscheiden.

Diese Gruppen sieht man auch schön in der Visualisierung der policy (1 Horizontschritt weiter, aber fast identisch) in Abbildung 6.

c) Siehe Pseudocode Übungsblatt.

d) Die optimalen Handlungen für b_1, b_2 und b_3 entsprechen der policy aus dem MDP und sind auch direkt aus $|\Gamma|$ abzulesen: $b_1 : u_3$, $b_2 : u_1$, $b_3 : u_2$.

e) Die optimale Handlung für b_4 ist u_2. Das liegt vor allem daran, dass die Belohnung für das erfolgreiche Führen so hoch ist: $4 * 1/3 > -1 * 2/3$.

f) Wenn die Belohnung für den Versuch, den Menschen zu führen auf 3 (-1 für die Kosten gibt 2 für den Eintrag) verringert wird, ändert sich die optimale Handlung (bei $\gamma < 1$) für b_4 zu u_3 (nichts tun).

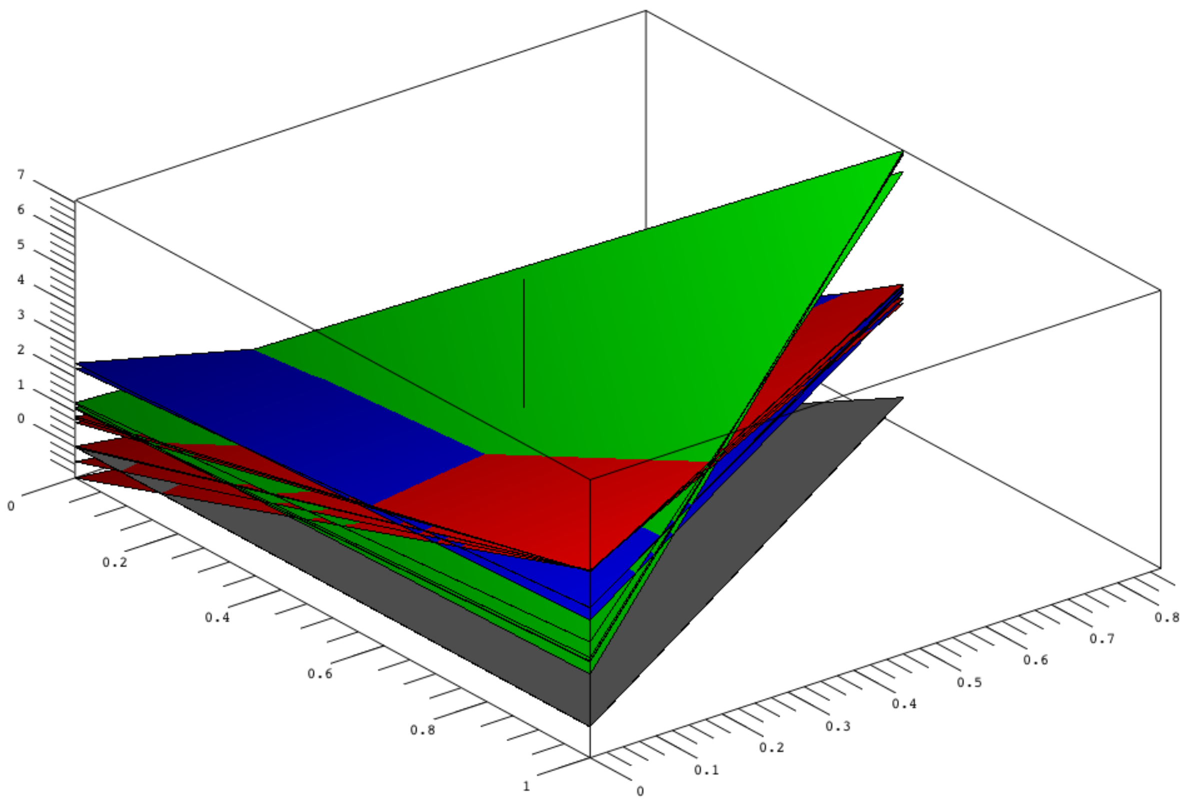

Abbildung 6: Visualisierung der value function Γ, b_4 ist markiert. u_1: rot, u_2: grün, u_3: blau.