9. Dezember 2012

GKI - Hausaufgaben 4

Tao Xu, 343390 - Mitja Richter, 324680 - Björn Kapelle, 320438 - Marcus Weber, 320402

Aufgabe 1

1.a)

Wir wählen folgende Konstanten:

K={A, R1, R2, R3, H, C1, C2, C3}

Dabei steht:
A für den Agenten, sprich den Roboter;
R1, R2, R3 für die einzelnen Räume;
H für den Gang;
C1, C2, C3 für die Kisten.

Weiterhin haben wir folgende Prädikate:

P={Room/1, Item/1, Open/1, Closed/1, Carry/1, Clear/1, In/2}

Diese geben an:
Room(x) $\hat{=}$ x ist einer der drei Räume;
Item(x) $\hat{=}$ x ist ein Gegenstand;
Open(x) $\hat{=}$ x ist offen;
Closed(x) $\hat{=}$ x ist geschlossen;
Carry(x) $\hat{=}$ x wird vom Agenten getragen;
Clear(x) $\hat{=}$ x trägt keine Gegenstände;
In(x,y) $\hat{=}$ x befindet sich in y.

Daraus ergibt sich folgender Startzustand:

S_0 = {Room(R1), Room(R2), Room(R3), Item(C1), Item(C2), Item(C3), Closed(R1), Closed(R2), Closed(R3), In(A,R1), In(C1, R1), In(C2, R2), In(C3, R3), Clear(A)}

Und folgender Zielzustand:

S_z = {In(C1, R1), In(C2, R1), In(C3, R1)}

Folgende Aktionsschemata stehen zur Verfügung:

ACT: moveIn(r)
PRE: Room(r), Open(r), In(A,h)
ADD: In(A,r)
DEL: In(A,H)
Bem: Agent bewegt sich vom Gang in den Raum r

ACT: moveOut(r)
PRE: Room(r), Open(r), In(A,r)
ADD: In(A,H)
DEL: In(A,r)
Bem: Agent bewegt sich vom Raum r in den Gang

ACT: open(r)
PRE: Room(r), Closed(r), Clear(A)
ADD: Open(r)
DEL: Closed(r)
Bem: Agent öffnet Raum r

ACT: close(r)
PRE: Room(r), Open(r)
ADD: Closed(r)
DEL: Open(r)
Bem: Agent schließt Raum r

ACT: put(i)
PRE: Item(i), Carry(i), Room(r), In(A,r)
ADD: In(i,r), Clear(A)
DEL: Carry(i)
Bem: Agent legt Gegenstand i im Raum r ab

ACT: take(i)
PRE: Item(i), Room(r), In(i,r), In(A,r), Clear(A)
ADD: Carry(i)
DEL: In(i,r), Clear(A)
Bem: Agent nimmt Gegenstand i im Raum r auf

Dieses Modell nimmt zur Vereinfachung an, dass der Agent keine Gegenstände im Gang ablegen bzw. aufnehmen kann. Dies erscheint plausibel, da der Agent die Kisten lediglich zwischen den Räumen hin und her transportieren soll. Andernfalls müsste man noch zusätzliche Aktionsschemata für put und take sowie ein zusätzliches Prädikat für In/2 definieren.

1.b)

i) In S_0 anwendbar sind: {open(R1), open(R2), open(R3), take(C1)}

ii) Der Plan [take(C1), put(C1)] endet mit einer konsistenten und relevanten Aktion, denn put(C1) ist konsistent, denn es entfernt keines der drei Atome, welche im Zielzustand enthalten sind und put(C1) ist relevant, denn es fügt In(C1,R1) hinzu, welches zu diesem Zeitpunkt nicht im aktuellen Zustand enthalten ist.

iii) Der Plan [take(C1)] endet mit einer nicht konsistenten Aktion, denn take(C1) entfernt das Atom In(C1,R1), welches im Zielzustand enthalten ist.

1.c)

i) Prinzipiell kommen alle Aktionen in Frage, die ein In/2 im ADD haben. Davon scheiden aber moveIn und moveOut aus, denn deren In/2 sind bereits mit der Konstante A belegt. Es bleibt nur put übrig. Zielführende Aktionen sind daher put(C1,R1), put(C2, R2) und put(C3,R3).

ii) In unserem Modell resultieren keine Aktionen aus i) in unmögliche Vorgängerzustände S_{Z-1}.

iii) Für die letzte Aktion put(C1,R1) lautet der Vorgängerzustand S_{Z-1}:

S_{Z-1} = {In(C2,R2), In(C3,R3), Carry(C1)}

Für die vorletzte Aktion moveIn(R1) lautet der Vorgängerzustand S_{Z-2}:

S_{Z-2} = {In(C2,R2), In(C3,R3), Carry(C1), In(A,H)}

1.d)

Man bräuchte ein zusätzliches Prädikat Energy(x), welches den vorhandenen Energievorrat des Agenten beschreibt. Dazu köennte man Konstanten 0,..,M $\in \mathbb{N}$ einführen, die eine Skala für die Energie beschreiben. 0 hieße dann gar keine Energie und M maximale Energie. Problematisch ist die Modellierung der Energieabnahme, da STRIPS keine arithmetischen Operationen kennt. Eine (zugegebener Maßen unschöne) Möglichkeit dies zu lösen, wäre jede Aktion für jedes Energielevel zu definieren. Statt der oben definierten moveIn(r) erhielte man dann z.B. für das Energielevel 1:

ACT: moveIn(r)
PRE: Room(r), Open(r), In(A,h), Energy(1)

ADD: In(A,r), Energy(0)
DEL: In(A,H), Energy(1)
Bem: Agent bewegt sich vom Gang in den Raum r

Analog für jede andere Konstante k = 1,...,M und jede andere Aktion. Der gewünschte Effekt wäre dann, dass bei Energy(0) keine PRE's mehr erfüllt und der Agent damit handlungsunfähig wäre. Nachteil dieser Methode ist aber, dass sich die Anzahl der Aktionen ver-M-facht, was insbesondere bei großem M ein Problem darstellen könnte.

1.e)

Die Auswirkungen auf die Vorwärtsplanung wären gering, da das Energielevel im Startzustand bekannt sein muss und man dann in jedem Schritt nur genauso viele mögliche Aktionen wie im ursprünglichen Modell hätte. Es bestünde aber natürlich die Gefahr, dass der Agent handlungsunfähig wird, bevor er den Zielzustand erreicht.

Die Rückwärtsplanung hingegen hätte plötzlich einen viel größeren Verzweigungsgrad, da im Zielstand nichts über das Energielevel bekannt ist und demnach jede Aktion für alle M Energielevel durchprobiert werden müsste.

Aufgabe 2

2.a)

Wir starten im Zustand S_Z. Wir erfüllen das erste Ziel $g(p)$ mit Hilfe von A_2. $g(w)$ mit A_3, $g(b)$ mit A_5 und $SO(z)$ wird im Startzustand erfüllt. Da soweit keine Aktionen mit DEL-Eintrag verwendet wurden gibt es bis jetzt auch keine Bedrohungen.

Ziele von $A5$: $F(f,b)$ wird im Startzustand erfüllt und $SO(b)$ mit A_4. Da A_4 $SO(z)$ löscht, dieser aber im Zielzustand gebraucht wird und keine Möglichkeit einer Promo- oder Demotion besteht, gilt $SO(z)$ in A_Z wieder als zu erfüllendes Ziel.

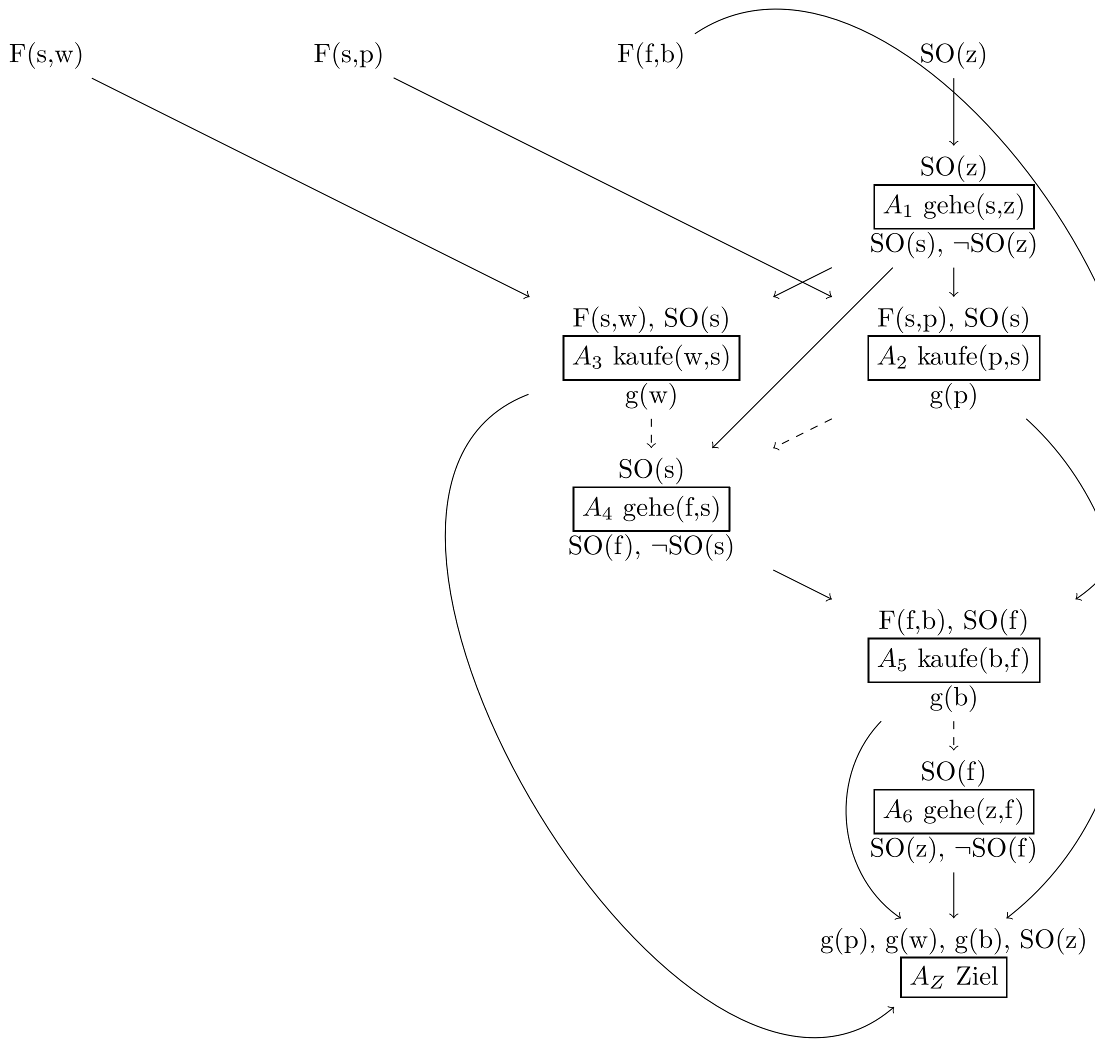
Ziel $SO(z)$ in A_Z wird erfüllt mit A_6. Zwischen A_5 und A_6 besteht eine Bedrohung, weswegen A_5 vor A_6 promoviert wird.

Das Ziel $SO(s)$ in A_4 wird durch A_1 erfüllt. Da nun Bedrohungen zwischen A_4-A_3 und A_4-A_2 entstanden sind müssen A_3 und A_2 vor A_4 promoviert werden.

Das Ziel $SO(s)$ von A_2 und A_3 wird von A_1 erfüllt. und die restlichen Ziele $F(s,w)$, $F(s,p)$, $SO(z)$ von A_2, A_3 und A_1 werden in A_0 erfüllt.

Daraus ergibt sich folgender Plan:

(Abkürzungen: F=Führt, SO=Standort, g=Gekauft, z=Zuhause, s=Supermarkt, f=Florist, w=Wein, p=Pralinen, b=Blumen)



2.b

Der erstellte Plan kann A_2 und A_3 parallel ausführen.