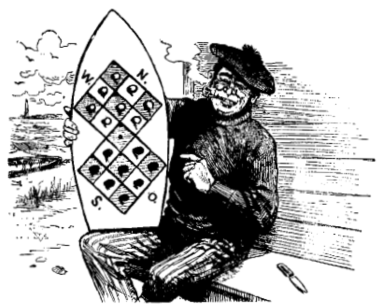
Künstliche Intelligenz I (HWS 2010)
Übungsblatt 9 (7 + 5 + 1)

Christian Meilicke & Heiner Stuckenschmidt
{christian,heiner}@informatik.uni-mannheim.de



Ausgabe: 24.11.2010 Uhr
Abgabe: 06.12.2010 09:00 Uhr

Aufgabe 1 (3 Punkte)

Wir haben das Spiel Seemannssolitär bereits auf einem der vorigen Blätter kennengelernt. Die Regeln des Spiels und weitere Informationen finden sich auf der letzten Seite dieses Übungsblatts. Im folgenden betrachten wir die Variante des Spiels, bei der es nicht notwendig ist schwarze und weiße Steine abwechselnd zu ziehen (entspricht dem Original-Regelwerk). Es gehts nun darum das Spiel als Planungsproblem zu verstehen und in der STRIPS Notation zu beschreiben.

a) (1,5 Punkte) Die Prädikate einer geeigneten Darstellung sagen etwas über die einzelnen Felder des Spielfelds aus (ist das Feld von einem schwarzen oder weißen Stein besetzt, sind Felder benachbart, sind Felder in einer Reihe):

```
(define (domain sailor)

   (:requirements :strips)
   (:predicates
      (clear ?square)
      (black ?square)
```

```
        (white ?square)
        (adjacent ?square1 ?square2)
        (row ?square1 ?square2 ?square3)
    )
    (:action move
        ...
    )
    (:action jump
        ...
    )
)
```

Vervollständige hierauf basierend die Handlungsspezifikationen!

b) (1,5 Punkte) Erstelle ein Fakten-File für eine verkleinerte Variante des Problems (Spielfeld mit 7 Feldern, 2 $\times$ 4 Felder, die sich in einem Feld überlappen). Enthalten sein muss der Startzustand, Zielzustand, und eine Beschreibung der Topologie des Spielfeldes.

```
(define (problem sailorsmall)
    (:domain sailor)
    (:objects ...)
    (:init ...)
    (:goal ...)
)
```

Aufgabe 2 **(2 + 2 Punkte)**

Im folgenden seht ihr das Planungsproblem aus Abbildung 11.2 (2nd edition) aus *Russel and Norvig: AI - a modern approach* (Abbilung 10.1 in der 3rd edition). Konstruiere die Ebenen 0, 1 und 2 des Planungsgraphen für dieses Problem! Hinweis: Der Graph wird recht umfangreich, sollte allerdings auf ein DIN A4 Blatt passen.

$Init(At(C_1, SFO) \wedge At(C_2, JFK) \wedge At(P_1, SFO) \wedge At(P_2, JFK)$
$\quad \wedge Cargo(C_1) \wedge Cargo(C_2) \wedge Plane(P_1) \wedge Plane(P_2)$
$\quad \wedge Airport(JFK) \wedge Airport(SFO))$
$Goal(At(C_1, JFK) \wedge At(C_2, SFO))$
$Action(Load(c, p, a),$
$\quad$ PRECOND: $At(c, a) \wedge At(p, a) \wedge Cargo(c) \wedge Plane(p) \wedge Airport(a)$
$\quad$ EFFECT: $\neg At(c, a) \wedge In(c, p))$
$Action(Unload(c, p, a),$
$\quad$ PRECOND: $In(c, p) \wedge At(p, a) \wedge Cargo(c) \wedge Plane(p) \wedge Airport(a)$
$\quad$ EFFECT: $At(c, a) \wedge \neg In(c, p))$
$Action(Fly(p, from, to),$
$\quad$ PRECOND: $At(p, from) \wedge Plane(p) \wedge Airport(from) \wedge Airport(to)$
$\quad$ EFFECT: $\neg At(p, from) \wedge At(p, to))$

Da das Einzeichnen aller Mutex Links unzumutbar wäre, müssen nicht alle Mutex Links markiert werden. Auf den Folien werden vier Arten von Mutex Links unterschieden.

a) Inconsistent Effects

b) Competing Needs

c) Interference (prec-effect)

d) Inconsistent Support

Markiere für jeden Typ einen entsprechenden Mutex Link im Graphen (insofern vorhanden) und mache durch eine kurze Erklärung ersichtlich, wieso es sich um diesen Typ handelt (2 Punkte)!

Programmieraufgabe VII (5 Punkte)

Recherchiere im Internet nach Programmen, mit denen man Planungsprobleme, die in PDDL formuliert sind, lösen kann, und teste mindestens eines der Programme. Deine eigentliche Aufgabe besteht darin, dass Original Seemanns-solitär-Problem mit Hilfe des Programms zu lösen! Wieviel Schritte werden benötigt und wie lange dauert es, eine Lösung zu finden?

Erkläre in einem Absatz den in dem System implementierten Algorithmus (nachlesen auf der Webseite oder in einem Paper)! Erläutere weiterhin in einem Absatz wie man das System installieren und benutzen kann (Downloadaddresse, Kommandozeilenaufruf oder GUI, Inputparameter, ...)!

Alternative Programmieraufgabe (3,5 Punkte) Solltest du Probleme haben das Seemanssolitär als STRIPS Problem zu formulieren, dann kannst du alternativ nach einer STRIPS Formulierung des 'Towers-of-Hanoi' Problems suchen und dieses von der Planungssoftware lösen lassen. Berichte kurz über Problemgröße und Laufzeit! Auch hier gilt es, das verwendete System kurz zu beschreiben (vgl. oben)!

Bonusaufgabe (1 Punkte)

(Gilt als Programmierbonuspunkt) Ändere das Regelwerk so ab, dass weiß und schwarz immer abwechselnd ziehen müssen und löse das resultierende Problem mit dem Planungssystem. Wieviele Züge werden nun benötigt?

Ich möchte die Gelegenheit wahrnehmen und Ihre Aufmerksamkeit auf die Herkunft eines ganz besonders hübschen Spiels, eine Art Solitaire, lenken, das sich in Europa ziemlicher Beliebtheit erfreut. Es handelt sich dabei um eine englische Erfindung, denn der Seemann, von dem es stammt, war ein Engländer, der 40 Jahre seines Lebens in »Sailor's Snug Harbor« auf Staten Island verbrachte und dessen größter Stolz es war, unter Kapitän Randall, dem Begründer dieser Institution, zur See gefahren zu sein.

Der alte Seemann verdiente sich mit dem Spiel, das er an Besucher verkaufte, einen schönen »Tabak-Batzen«, wie er es nannte. Sein Klappmesser, mit dem er die Spiele zurechtschnippelte, konnte gar nicht schnell genug sein, um allen Nachfragen gerecht zu werden. Das Spiel kam dann auch in London heraus und erfreute sich unter dem Namen »English Sixteen Puzzle« großer Beliebtheit. Auf unserer Seite des großen Teichs war es allerdings nie auf dem Markt.

Ziel des Spiels ist es, die schwarzen und weißen Steine durch die geringste Anzahl Züge miteinander zu vertauschen. Jeder Stein kann von seinem Feld auf ein angrenzendes leeres gerückt werden, oder aber über einen Stein (ganz gleich, welcher Farbe) im danebenliegenden Feld springen – vorausgesetzt, das Feld, auf das er gelangt, ist ebenfalls leer. Die Züge dürfen (wie der Turm beim Schach) nur in waagerechter oder senkrechter Richtung erfolgen, nicht aber (wie beim Dame-Spiel) diagonal.

Nach Aussage eines Augenzeugen war der alte Seebär sehr stolz auf seine Erfindung. Stets gab er den Käufern Tips, wie sie mit möglichst wenig Spielzügen ans Ziel gelangen könnten. Allerdings sieht es so aus, als hätte er sich doch ein wenig geirrt. Oder aber wir haben es hier mit einer jener Künste zu tun, die im Verlauf der Zeit einfach verloren gegangen sind. Denn die Lösungen, die in englischen Rätselbüchern sowie in mathematischen Abhandlungen angegeben werden und angeblich die kürzeste Möglichkeit anbieten, weisen Mängel auf und lassen sich jeweils um mehrere Züge verkürzen.

[Seine Lösung verrät Loyd nicht. Die meisten Rätselbücher, sagt er, bieten eine Lösung mit 52 Zügen an, obgleich in Wirklichkeit nur 47 Züge nötig seien. Der britische Rätselexperte H. E. Dudeney übertrumpfte Loyd sogar um einen Zug und verringerte die Zahl auf 46. Dudeneys herrlich symmetrische Lösung findet sich in W. Rouse Balls *Mathematical Recreations and Essays*, S. 125. – M. G.]