50.021 – Artificial Intelligence

Kwan Hui

Week 11 Theory Homework - Planning

[The following notes are compiled from various sources such as textbooks, lecture materials, Web resources and are shared for academic purposes only, intended for use by students registered for a specific course. In the interest of brevity, every source is not cited. The compiler of these notes gratefully acknowledges all such sources. ]

Due: 15th Apr 2024, 11:59pm

Smission: via eDimension

**Name: Sharryl Seto**

**ID: 1005523**

**Class: Cl03**

# Logistic Problem I

Consider the following logistic problem. There are three locations *a*, *b* and *c*, with a truck at *a* and package at *c*. The truck is able perform the following actions: (i) move(x,y): move from location *x* to *y*; (ii) load(x): load a package at location *x*; and (iii) unload(x): unload the package at location *x*. The truck can only move between adjacent locations, e.g., *a* to *b*, *b* to *c* (You can assume that these static facts are already modelled/defined).

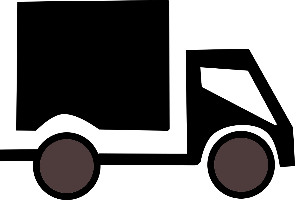
**b**

**a**

**c**

Package

Truck



Given the start state in the above diagram, your goal is to get the package to location *b*. Formulate this logistic problem using the STRIPS representation and answer the following:

STRIPS, formally (P,O,I,G)

P – set of propositional variables (facts), O – set of operators / actions,

I – initial state, true/false assignments to P, G – goal state of the world

1. List down the propositional variables (facts).
2. Specify the operators (actions), including the pre-conditions and post-conditions.
3. Specify the initial state.
4. List down the goal state/specification.
5. Facts/variables: truckAt(x), packAt(x), packInTruck (where x can be a, b, or c)
6. Actions: (:action move(x, y)

:precondition truckAt(x)

:postcondition not truckAt(x), truckAt(y) )

(:action load(x)

:precondition truckAt(x), packAt(x)

:postcondition not packAt(x), packInTruck)

(:action unload(x)

:precondition truckAt(x), packInTruck

:postcondition packAt(x), not packInTruck)

1. Initial state: truckAt(a), packAt(c)
2. Goal: packAt(b)

# Logistic Problem II

Based on your STRIPS formulation from Q1 (Logistic Problem I), answer the following:

1. What is the optimal solution to this problem?

* F0 = truckAt(a), packAt(c)
* A0 = move(a, b)
* F1 = truckAt(b), packAt(c)
* A1 = move(b, c)
* F2 = truckAt(c), packAt(c)
* A2 = load(c)
* F3 = truckAt(c), packInTruck
* A3 = move(c, b)
* F4 = truckAt(c), packInTruck
* A4 = unload(b)
* F5 = truckAt(b), packAt(b)

1. Make this a delete-relaxed problem. What are the changes to the original STRIPS formulation you made?

Remove negation of facts in all operators.

Actions: (:action move(x, y)

:precondition truckAt(x)

:postcondition ~~not truckAt(x)~~, truckAt(y) )

(:action load(x)

:precondition truckAt(x), packAt(x)

:postcondition ~~not packAt(x)~~, packInTruck)

(:action unload(x)

:precondition truckAt(x), packInTruck

:postcondition packAt(x), ~~not packInTruck~~)

1. Based on this delete-related problem, list down all the facts *Fx* and actions *Ax* at levels *x* = {0*,*1*,...,M*}.

* F0 = truckAt(a), packAt(c)
* A0 = move(a, b)
* F1 = truckAt(a), truckAt(b), packAt(c)
* A1 = move(b, c)
* F2 = truckAt(a), truckAt(b) ,truckAt(c), packAt(c)
* A2 = load(c)
* F3 = truckAt(a), truckAt(b) ,truckAt(c), packAt(c), packInTruck
* A3 = move(c, b)
* F4 = truckAt(a), truckAt(b) ,truckAt(c), packAt(c), packInTruck
* A4 = unload(b)
* F5 = truckAt(a), truckAt(b) ,truckAt(c), packAt(c), packInTruck, packAt(b)

# Logistic Problem III

Based on your answer from Q2 (Logistic Problem II), answer the following:

1. What is the optimal solution to this delete-relaxed problem? What is this heuristic called?

Optimal solution: F0 -A0-> F1 -A1-> F2 -A2-> F3 -A4-> F5 (skip F4). **It is called h+.**

o1

move(b, c)

o2

Can skip A3 and F4

F0

move(a, b)

F1

F2

F3

F5

Precondition is truckAt(b), packInTruck

o4

o3

unload(c)

load(c)

A4

A2

A1

A0

1. What is the value of *hadd*? Explain why.

The goal is packAt(b). So *hadd* = 1+1+1+1 = 4 for operator A0, A1, A2, A4 from initial state to goal.

1. What is the value of *hmax*? Explain why.

The goal is packAt(b). So = *hmax* = max(4) = 4, same 4 actions from initial to goal as well.

# Generic Planning I

Consider a STRIPS problem with propositional variables (facts) *m,n,o,p*, and the below STRIPS actions with their pre/post-conditions.

◦ *prea* - facts that must be true before the action can be performed  
◦ *adda* - facts that will change to true when/after the action can be performed

◦ *dela* - facts that will change to false when/after the action can be performed

|  |  |  |  |
| --- | --- | --- | --- |
| Action | Pre | Add | Del |
| A | m | n,o | ∅ |
| B | m,o | p | m |
| C | p | m | p |
| D | n,o | p | o |

Given an initial state *s* = {*m*} and goal specification *g* = {*m,n,o,p*}, answer the following questions:

1. What is the value of *h*+? Explain why.

D

F0

A0

A

Or B

A1

F2

F1

Delete-relaxed problem, ignore del column. Action A, D. So h+ is 1+1 = 2.

* F0 = m
* A0 = action A
* F1 = m, n, o
* A1 = action D
* F2 = m, n, o, p

1. What is the value of *hadd*? Explain why.

Delete-relaxed problem, ignore del column. Same as above

Action A, D. So *hadd* is 1+1 = 2.

1. What is the value of *hmax*? Explain why.

Delete-relaxed problem, ignore del column. Same as above

Action A, D. So *hmax* is max(2) = 2.

# Generic Planning II

Based on the same STRIPS formulation in Q4 (Generic Planning I). Now, based on initial state *s* = {*p*} and goal specification *g* = {*m,n,o,p*}, answer the following questions:

1. What is the value of *h*+ (if any)? Explain why.

F0

F1

A0

A1

F2

A

C

Delete-relaxed problem. Action C, A. So h+ is 1+1=2.

1. What is the value of *hadd* (if any)? Explain why.

Delete-relaxed problem. Action C, A. So add is 1+1=2.

1. What is the value of *hmax* (if any)? Explain why

Delete-relaxed problem. Action C, A. So max(2)=2.