National University of Singapore
School of Computing

Semester 1, AY2021-22

CS4246/CS5446 AI Planning and Decision Making

Issued: 1 Oct, 2021

Tutorial Week 8: Mid-Term Review

Guidelines

You may discuss the content of the questions with your classmates. But everyone should work on and be ready to present ALL the solutions. Note: Materials in this Tutorial cover only **some** of the topics relevant to the Mid-Term Exam on 8 October 2021.

Problem 1: Classical Planning

Modified from [RN 10.3 Kindle Edition]

$$
\begin{aligned}
&Init(At(C_1,\ SFO) \land At(C_2,\ JFK) \land At(P_1,\ SFO) \land At(P_2,\ JFK)\\
&\quad \land Cargo(C_1) \land Cargo(C_2) \land Plane(P_1) \land Plane(P_2)\\
&\quad \land Airport(JFK) \land Airport(SFO))\\
&Goal(At(C_1,\ JFK) \land At(C_2,\ SFO))\\
&Action(Load(c,\ p,\ a),\\
&\ \ \text{PRECOND: } At(c,\ a) \land At(p,\ a) \land Cargo(c) \land Plane(p) \land Airport(a)\\
&\ \ \text{EFFECT: } \neg\, At(c,\ a) \land In(c,\ p))\\
&Action(Unload(c,\ p,\ a),\\
&\ \ \text{PRECOND: } In(c,\ p) \land At(p,\ a) \land Cargo(c) \land Plane(p) \land Airport(a)\\
&\ \ \text{EFFECT: } At(c,\ a) \land \neg\, In(c,\ p))\\
&Action(Fly(p,\ from,\ to),\\
&\ \ \text{PRECOND: } At(p,\ from) \land Plane(p) \land Airport(from) \land Airport(to)\\
&\ \ \text{EFFECT: } \neg\, At(p,\ from) \land At(p,\ to))
\end{aligned}
$$

a) Given the action schemas and initial state from the figure above, what are all the applicable concrete instances of $Fly(p, from, to)$ in the state described by

$$At(P_1, JFK) \land At(P_2, SFO) \land Plane(P_1) \land Plane(P_2) \land Airport(JFK) \land Airport(SFO)?$$

Solution:
Applicable means satisfying preconditions. The applicable actions are: $Fly(P_1, JFK, SFO)$, $Fly(P_1, JFK, JFK)$, $Fly(P_2, SFO, JFK)$, $Fly(P_2, SFO, SFO)$. Flying from one airport to itself is allowable, if not useful.

b) What is the result of executing the action $Load(C_2, P_2, JFK)$ from the initial state?

Solution:
First check that the action is applicable at the initial state. Since it is applicable, we can execute the action. To compute the result, remove the literal that appears in the delete list, $At(C_2, JFK)$, and add the literal that appears in the add list, $In(C_2, P_2)$ giving a new state

$$At(C_1, SFO) \wedge At(P_1, SFO) \wedge At(P_2, JFK) \wedge Cargo(C_1) \wedge Cargo(C_2)$$
$$\wedge Plane(P_1) \wedge Plane(P_2) \wedge Airport(JFK) \wedge Airport(SFO) \wedge In(C_2, P_2).$$

c) In regression or relevant-state search, we use *description* instead of state.
True or false: the goal in STRIPS is a description instead of a state. Why?

Solution:
True, a state s satisfies a goal g if s contains all literals in g. This is the same condition for a state to satisfy a description.

d) What actions are relevant to the description $In(C_2, p)$?

Solution:
$Load(C_2, p, a)$.

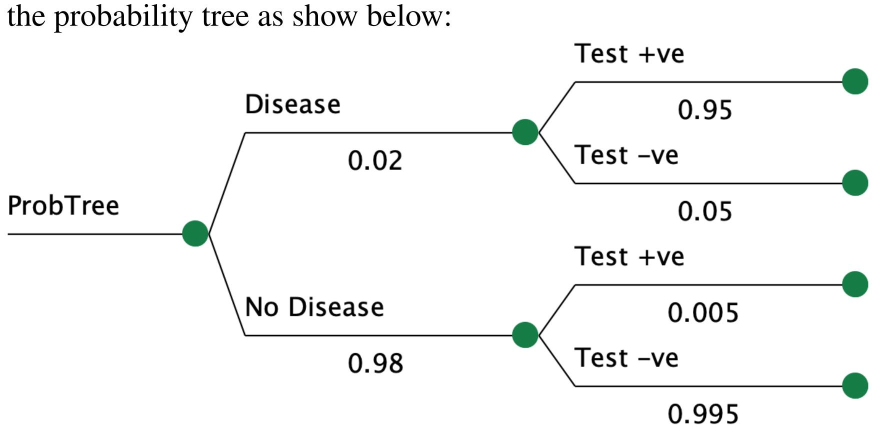
e) What is the outcome of regressing the description $At(C_1, JFK) \wedge At(C_2, SFO)$ over action $Unload(C_1, p, JFK)$?

Solution:

$$In(C_1, p) \wedge At(p, JFK) \wedge Cargo(C_1) \wedge Plane(p) \wedge Airport(JFK).$$

Problem 2: Decision Analysis

Flip the probability tree as show below:

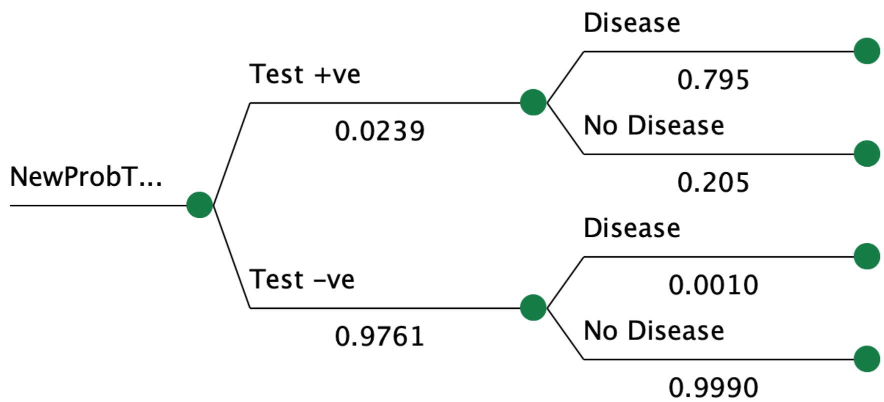


Solution:

Let:
$Pos = positive$
$Neg = negative$
$D = disease$
$Not\_D = no\ disease$

Using Law of total probability:
$P(Pos) = 0.0239$
$P(Neg) = 1 - P(Pos) = 0.9761$

Using Baye's Theorem:
$P(D|Pos) = 0.795$
$P(D|Neg) = 0.0010$



Problem 3: Markov Decision Process

Adpated from [RN 17.10]

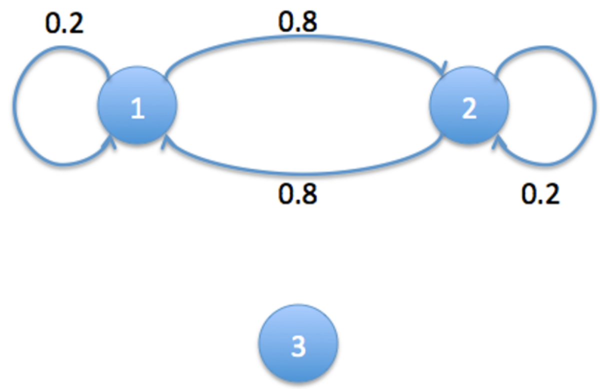
Consider an undiscounted MDP having three states, $(1, 2, 3)$, with rewards $-1, -2, 0$, respectively. State 3 is a terminal state. In states 1 and 2 there are two possible actions: A and B. The transition model is as follows:

• In state 1, action A moves the agent to state 2 with probability 0.8 and makes the agent stay put with probability 0.2.

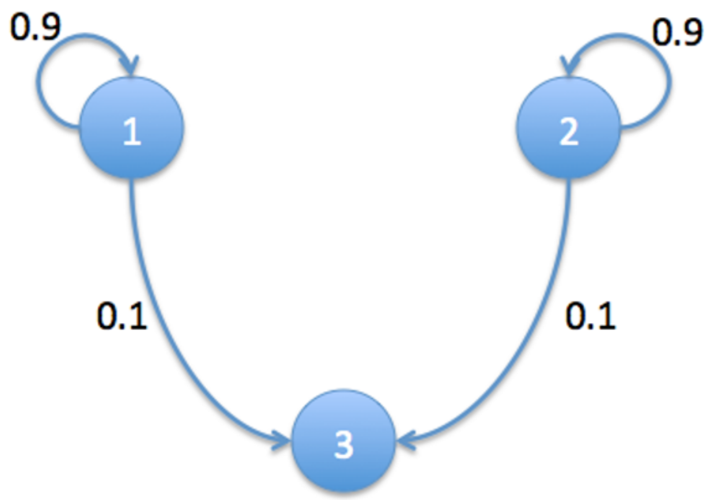
• In state 2, action A moves the agent to state 1 with probability 0.8 and makes the agent stay put with probability 0.2. • In either state 1 or state 2, action B moves the agent to state 3 with probability 0.1 and makes the agent stay put with probability 0.9.

a) Draw the state transition diagram for action a and b respectively.

Solution:



For action A:



Fora action B:

b) What can be determined qualitatively about the optimal policy in states 1 and 2?

Solution:

Agent should try to get to state 3 as soon as possible. In state 1, would be better to try action B, even with low probability of success. In state 2, would be better to try action A, to try to get to state 1, and then try to get to state 3 from there. This is because the negative reward of staying in state 2 is twice that of staying in state 1.

c) Apply policy iteration, showing each step in full, to determine the optimal policy and the values of state 1 and 2. Assume that the initial policy has action B in both states.

Solution:

Start with action B in both states
After policy evaluation: preferred action in State 1: B, preferred action in State 2: A
Set Unchanged to false

Next policy: execute B in State 1, A in state 2
After policy evaluation: preferred action in State 1: B, preferred action in State 2: A
Set Unchanged to True, Terminate

Optimal policy: in State 1: take action B, in State 2: take action A
Optimal policy matches intuition in part b)

Problem 4: Decision Theory

A season investor, Alice, has a utility function $U(x) = ln(x)$, where x is total wealth, has a choice between the following two alternatives:

A: Win $\$10,000$ with probability 0.2
Win $\$1,000$ with probability 0.8
B: Win $\$3,000$ with probability 0.9
Lose $\$2,000$ with probability 0.1

a) If Alice current wealth is $\$2,500$, should she choose A or B?

b) If Alice current wealth is $\$5,000$, should she choose A or B?

c) If Alice current wealth is $\$10,000$, should she choose A or B?

d) Do you think this pattern of choices between A and B is reasonable? Why or why not?

Solution:

a) If wealth = $2,500$, calculate EU(A) and EU(B). Choose A
b) If wealth = $5,000$, calculate EU(A) and EU(B). Choose B
c) If wealth = $10,000$, calculate EU(A) and EU(B). Choose A
d) This seems strange. One might think that as wealth level increases, a decision maker might change from one gamble to another, but never back to the first. (This is called Bell's one-switch rule, see reference). Only certainty utility functions have this "one-switch" property, but the logarithmic utility function is not one of them.

Ref: Bell, D. E. (1988) One-switch utility functions and measure of risk. Management Science, 34, 1416-1424.