Student Number:

The University of Melbourne

Semester 1 Assessment 2017

School of Computing and Information Systems

COMP30024 Artificial Intelligence

Reading Time: 15 minutes.

Writing Time: 3 hours.

This paper has 13 pages including this cover page.

Common Content Papers: None

Authorised Materials: None.

Instructions to Invigilators:
Each student should initially receive one standard script book.
Students must hand in **both** their **exam paper** and their **script book(s)**.
Students may **not** remove any part of the examination paper from the exam room.

Instructions to Students:

- This paper counts for 70% of your final grade, and is worth 70 marks in total.
- There are 7 questions, with marks as indicated. Attempt all questions.
- Answer questions 1, 2 and 3 **on the exam paper**, and answer questions 4, 5, 6, and 7 on the lined pages in your **script book**. If you need more space for questions 1, 2, or 3, then use the spare page at the end of the exam paper.
- You must hand in **both** your **exam paper** and your **script book(s)**.
- Start your answer to each question in the script book on a new page.
- Answer the questions as clearly and precisely as you can.
- Your writing should be clear. Unreadable answers will be deemed wrong. Excessively long answers or irrelevant information may be penalised.
- For numerical methods, marks will be given for applying the correct method. Students will not be heavily penalised for arithmetic errors.

Question 1 (10 marks) [Write your answers on this page]

Pick the most appropriate answer to each of the following questions. Please write your answer to each question in the boxes below. Each question is worth 2 marks.

Question	(a)	(b)	(c)	(d)	(e)
Answer					

(a) Which of the following statements is true in general:

1. both minimax and expectiminimax give perfect play
2. minimax gives perfect play, but expectiminimax does not
3. expectiminimax gives perfect play, but minimax does not
4. none of the above

(b) Consider a graph search problem where for every action, the cost is at least ϵ, with $\epsilon > 0$. In general which one of the following statements is true (based on the definition of the search algorithms used in the lectures).

1. Greedy graph search is guaranteed to return an optimal solution.
2. Depth-first graph search is guaranteed to return an optimal solution.
3. Uniform-cost graph search is guaranteed to return an optimal solution.
4. Breadth-first graph search is guaranteed to return an optimal solution.

(c) Let h_1 be an admissible heuristic, and let h_2 be an inadmissible heuristic, which one of the following is guaranteed to be admissible:

1. $\alpha * h_1 + (1 \quad \alpha) * h_2$, for $\alpha \in [0, 1]$
2. $h_1 * h_2$
3. $max(h_1, h_2)$
4. $min(h_1, h_2)$

(d) Considering English, Dutch, and first-price sealed-bid auctions, which one of the following is true in general:

1. English might be efficient, but Dutch and first-price sealed-bid auctions are always efficient.
2. English, Dutch, and first-price sealed-bid auctions are susceptible to collusion.

3. First-price sealed-bid is simpler in terms of communication overhead than English and Dutch auctions.

4. First-price sealed-bid auctions overcome the winner's curse, but English and Dutch auction don't.

(e) Figure 1-1 shows a robot arm that is made up of two sections Arm_1 and Arm_2. Arm_1 can rotate around the shoulder joint A, while Arm_2 is connected to Arm_1 at the elbow joint B, and can rotate around the joint B. The configuration of the robot arm can be specified by the angle θ_1 between the horizontal axis and Arm_1, and the angle θ_2 between Arm_1 and Arm_2. Both angles are measured in radians. There is also 1 fixed obstacle shown, which restricts the movement of the robot arm.

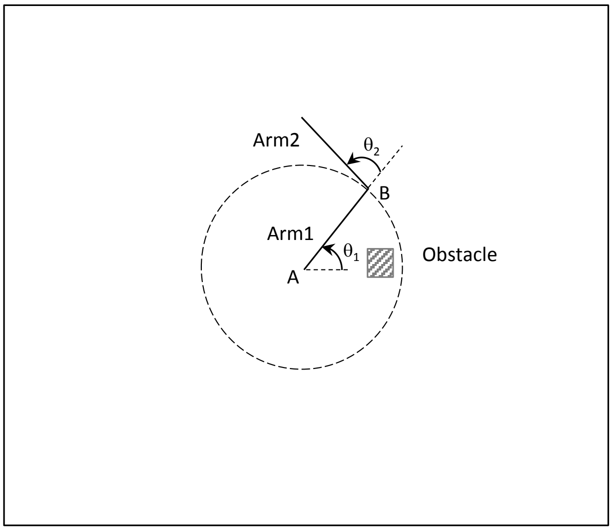


Figure 1-1

Which of the following four figures best represents the *configuration space* for this robot? The figures are labelled (1) to (4).

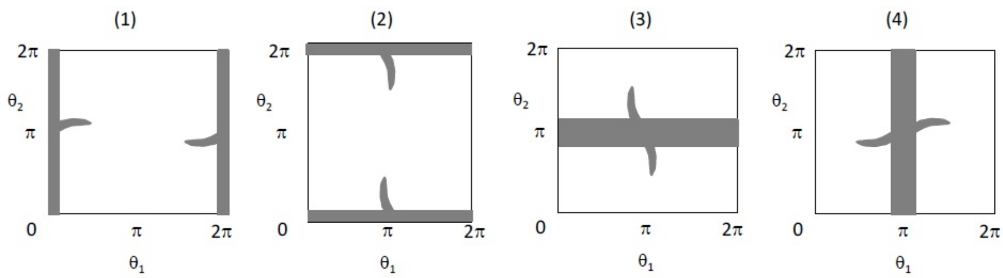


Figure 1-2

Question 2 (10 marks) [Write your answers on this page]

For each part of the following question you should write a brief answer in the box provided.

The following is the weight update rule for gradient decent learning as described in the lectures

$$w_i \leftarrow w_i + \eta(z \quad t)f_i(s)$$

where η is the learning rate, and w is the vector of weights in the evaluation function.

(a) [2 marks] What is the role of the learning rate parameter η?

(b) [2 marks] What are the two main problems with using gradient decent learning to learn the weights of an evaluation function in game playing?

(c) [6 marks] For each of the 3 following environments, what is the most appropriate learning approach (supervised learning or temporal difference learning)? Briefly justify your answer.

(i) Learning chess by playing against a skilled human opponent
(ii) Learning backgammon from a large number of labelled examples
(iii) Learning tic-tac-toe by playing against yourself

Question 3 (10 marks) [Write your answers on this page]

Consider the 3-ply game tree shown in Figure 3-1. Each node has an identifier (e.g., the root of the tree is node 1; it has three successor nodes 2, 12 and 22), and each terminal node has an associated value (e.g., the value of node 4 is 7).

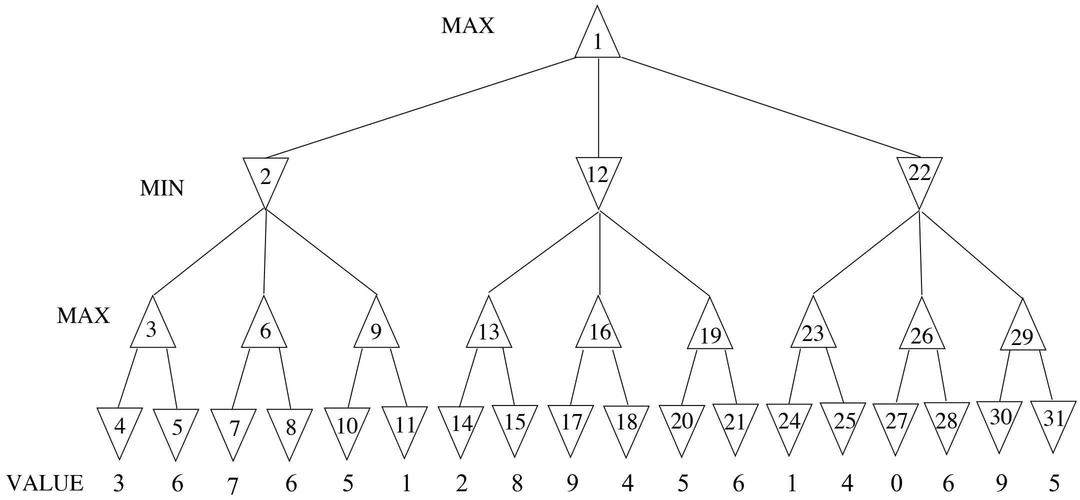


Figure 3-1

In the following questions, you are NOT required to redraw the search tree in your answer.

(a) [1 mark] What is the minimax value at node 1 after applying the minimax algorithm to this search tree? **Answer:**

(b) [4 marks] If the nodes are examined in the order shown by the identifier in each node in Figure 3-1, which nodes would be pruned if alpha-beta pruning is used? For each node that would be pruned, place a cross in the corresponding box below.

Answer:

Node	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Pruned																

Node	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Pruned															

(c) [2 marks] If we come up with the perfect ordering of the nodes, what is the maximum number of nodes that we can prune? **Answer:**

Question 3 (continued) [Write your answers on this page]

(d) [3 marks] What are the 3 main components that define an auction mechanism?

Write your answer in the space provided below. If you require more space, please use the last page of the exam paper, and clearly indicate on this page that you have used the last page.

Answer:

Question 4 (10 marks) [Write your answers in your script book]

A* is a well-known informed search algorithm which is guaranteed to return an optimal solution when given an admissible heuristic. Often in practical applications it is too expensive to find an optimal solution, so instead we search for good suboptimal solutions.

Weighted A* is a variant of A* commonly used for suboptimal search. Weighted A* is exactly the same as A* but where the f value is computed differently:

$$f(n) = g(n) + \varepsilon\, h(n)$$

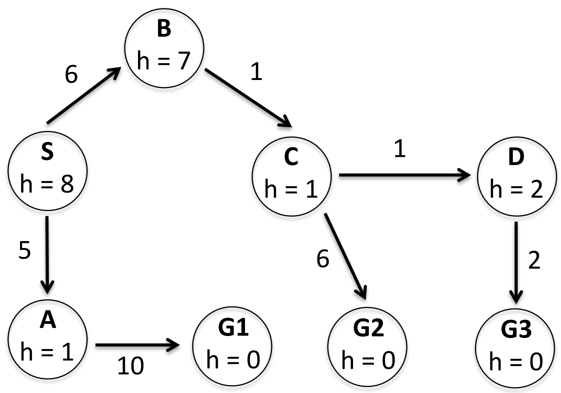
where $\varepsilon \geq 1$ is a parameter given to the algorithm, $g()$ is the path cost, and $h()$ is a heuristic. In general, the larger the value of ε, the faster the search is, and the higher the cost of the goal found.

Pseudocode for weighted A* tree search is given below. **NOTE:** The only differences from the A* tree search pseudocode presented in the lectures are: (1) $fringe$ is assumed to be initialized with the start node before this function is called (this will be important later), and (2) now INSERT takes ε as a parameter so it can compute the correct f-value of the node.

Algorithm 1 Weighted A*

function WEIGHTED-A*-TREE-SEARCH($problem$, $fringe$, ε)
    loop do
        if $fringe$ is empty **then return** failure
        $node \leftarrow$ REMOVE-FRONT($fringe$)
        if GOAL-TEST($problem$, STATE[$node$]) **then return** $node$
        $successors \leftarrow$ EXPAND($node, problem$)
        for each $s \in successors$ **do**
            $fringe \leftarrow$ INSERT(s, $fringe$, ε)
        end for
    end loop
end

(a) [2 m



Execute weighted A* on the above graph with $\varepsilon = 2$ and start node S, completing the following table in your script book. To save time, you can optionally just write the nodes added to the fringe, with their g and f values. An example of the table you should write in your script book is given below.

node	Goal?	*fringe*
-	-	$\{S : g = 0, f = 16\}$
S	No	$\{S \to A : g = 5, f = 7;\ S \to B : g = 6, f = 20\}$
$\vdots$	$\vdots$	$\vdots$

(b) [3 marks] Weighted A* includes a number of other algorithms as special cases. For each of the following, name the corresponding algorithm.

(i) $\varepsilon = 1$.

(ii) $\varepsilon = 0$.

(iii) $\varepsilon \to \infty$ (i.e., as ε becomes arbitrarily large).

(c) [5 marks] After running weighted A* with weight $\varepsilon \geq 1$ a goal node G is found, of cost $g(G)$. Let C^* be the optimal solution cost, and suppose the heuristic is admissible. Select the strongest bound below that holds, and provide a proof (partial credit for reasonable proof sketches.).

(i) $g(G) \leq \varepsilon\, C^*$

(ii) $g(G) \leq C^* + \varepsilon$

(iii) $g(G) \leq C^* + 2\varepsilon$

(vi) $g(G) \leq 2^{\varepsilon}\, C^*$

Question 5 (10 marks) [Write your answers in your script book]

Consider the general less-than chain CSP below. Each of the N variables X_i has the domain $\{1...M\}$. The constraints between adjacent variables X_i and $X_i + 1$ require that $X_i < X_i + 1$. The corresponding constraint graph for this problem is shown in Figure 5-1.

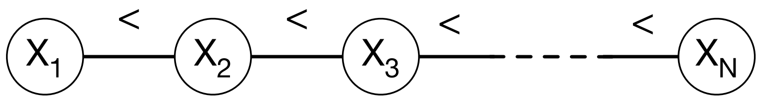


Figure 5-1

For now, assume N = M = 5.

(a) [1 mark] How many solutions does the CSP have?

(b) [2 marks] What will the domain of X_1 be after enforcing the consistency of only the arc $X_1 \rightarrow X_2$?

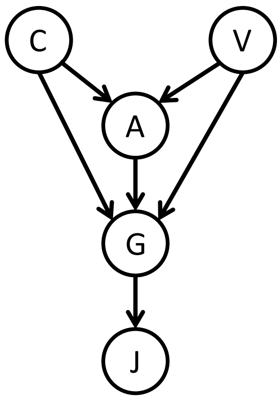
(c) [2 marks] What will the domain of X_1 be after enforcing the consistency of only the arcs $X_2 \rightarrow X_3$ then $X_1 \rightarrow X_2$?

(d) [2 marks] What will the domain of X_1 be after fully enforcing arc consistency?

(e) [3 marks] What is the minimum number of arcs (big-O is ok) which must be processed by AC-3 (the algorithm which enforces arc consistency) on this graph before arc consistency is established?

Question 6 (10 marks) [Write your answers in your script book]

Consider the Bayes network shown in Figure 6-1, where C = CommitCrime, A = Arrested, G = FoundGuilty, J = SentToJail and V = TelevisedStory are all Boolean random variables, i.e., they take the value either *true (t)* or *false (f)*. Also note that $P(c)$ is shorthand for $P(C = \text{true})$ and $P(\neg c)$ is shorthand for $P(C = \text{false})$.



P(C)	P(V)
0.9	0.1

C	V	P(A|C,V)
t	t	0.9
t	f	0.5
f	t	0.5
f	f	0.1

G	P(J|G)
t	0.9
f	0.0

C	A	V	P(G|C,A,V)
t	t	t	0.9
t	t	f	0.8
t	f	t	0.0
t	f	f	0.0
f	t	t	0.2
f	t	f	0.1
f	f	t	0.0
f	f	f	0.0

Figure 6-1

(a) [4 marks] Briefly explain whether or not J is conditionally independent of A given C.

(b) [6 marks] Calculate the value of $P(g|\neg j, a, c)$. If you cannot easily calculate the final value, try to simplify the expression as best you can.

Question 7 (10 marks) [Write your answers in your script book]

Consider a robot that patrols a circular field, where the field is divided into 3 equal-sized zones labelled z_1, z_2 and z_3 as shown in Figure 7-1. The robot continually moves in a clockwise circular path around the centre of the field at a constant speed, so that it always moves through the zones in the order $\ldots z_3, z_1, z_2, z_3, z_1, \ldots$. Note that the robot never changes direction.

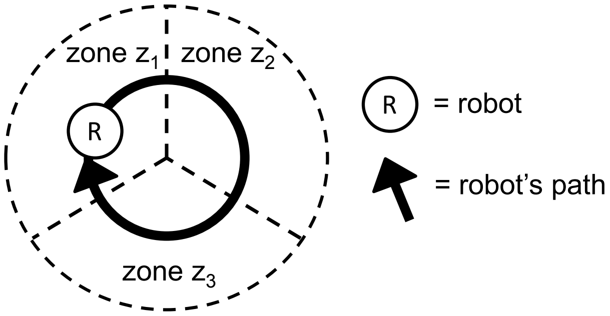


Figure 7-1

The robot has a sensor to detect its location, i.e., which zone it is in. However, the sensor is only 50% reliable. When the robot is in any given zone z_i, it has a 50% chance of correctly reporting that it is in zone z_i, otherwise the sensor will incorrectly report that it is in one of the other two zones each with equal probability. For example, if the robot is actually in zone z_3, then there is a 50% chance of the sensor correctly reporting that it is in zone z_3, a 25% chance of incorrectly reporting that it is in zone z_1, and a 25% chance of incorrectly reporting that it is in zone z_2.

At regular time intervals $t = 1, 2, 3, \ldots$ the sensor reports what zone it detects the robot to be in. You can assume that the speed of the robot is fixed so that each time the robot reports its location $t = 1, 2, 3, \ldots$, the robot will have moved into the next zone in the sequence. You can also assume that the robot will never sit on the border between two zones when the sensor reports the location of the robot. For example, if the robot is actually in zone z_3 at time $t = 2$, then it must be in zone z_1 at time $t = 3$.

Assume that when the robot is activated at $t = 1$, it does not know which zone it is in, so that it initially can be in any of the 3 zones with equal probability. At time $t = 1$ the sensor reports that it is in zone z_1. Then at time $t = 2$ the sensor reports that it is in zone z_1 again. Clearly, one or both of these reported locations must be incorrect.

Given the robot's sensor reports that it is in zone z_1 at time $t = 1$, and zone z_1 at time $t = 2$:

(a) calculate the probability that the robot is actually in zone z_2 at time $t = 2$, and

(b) calculate the probability that the robot is actually in zone z_3 at time $t = 2$.

Mathematically justify your answers in each case by using the *incremental* form of Bayes rule, and clearly show your reasoning and assumptions.

END OF EXAM QUESTIONS

Extra space if needed to answer questions 1, 2 or 3. If you write part of your answer here, please write the question number, and indicate at the corresponding question that you have used this space.

LAST PAGE OF EXAM