Coláiste na Tríonóide, Baile Átha Cliath
Trinity College Dublin
Ollscoil Átha Cliath | The University of Dublin

Faculty of Engineering, Mathematics and Science

School of Computer Science & Statistics

Integrated Computer Science Programme **Semester 2 2020**
B.A. (Mod.) Computer Science & Business
B.A. (Mod.) Computer Science & Language
Year 3 Annual Examinations

Artificial Intelligence I

Fri-Sat, 1-2 May 2020 **Take Home** **9:30 – 9:29+1day**

Dr Tim Fernando

Instructions to Candidates:

Answer two of the three questions, labelled Q1, Q2 and Q3. If you answer all three questions, your answer to question Q3 will be ignored (and left unmarked).

Each question is worth 50 marks; two questions together are 100 marks. To avoid penalties for overly long answers, please take the time to make your answers **concise**. *Use the allocated marks as a guide to the length of your answer. An answer to a part of a question worth N marks should not take more than N lines of prose, where a line has at most 15 words. N-lines-for-N-marks is not a hard and fast rule; it can be bent by lines of 5 words ... But for an entire question (of 50 marks), a word count of 800* ($\geq 50 \times 15$) *is a safe upper bound. As for a lower bound, try N-words-for-N-marks. These measures leave out pictures, which, where appropriate, are welcome.*

Submissions can be handwritten or typed, and uploaded as PDF or plain text files. (PDF converters are available online.) Please include a declaration based on the template at the end of this file.

Question Q1 Given

(i) a binary predicate arc/2 between nodes

(ii) a unary predicate goal/1 on nodes

(iii) and a node Start

we can search for a goal node along arcs from Start by the query

```
search(Start)
```

where

```
search(Node) :- goal(Node).
search(Node) :- arc(Node,Next), search(Next).
```

(a) Is there a way to ensure search terminates? Justify your answer.

[10 marks]

For an informed, intelligent search, two functions returning numbers are often introduced: a cost function on arcs, and a heuristic function h on nodes.

(b) How does A-star use these functions to guide search? Explain by outlining modifications to the predicate search. You may answer in succinct English, instead of Prolog.

[12 marks]

(c) Recall that A-star is said to be *admissible* if it returns a solution of minimal cost precisely when a solution exists. Must the heuristic function h underestimate the true cost for A-star to be admissible? Explain.

[8 marks]

(d) Focussing next on the goal predicate, suppose a goal node is given through a **constraint satisfaction problem** [Var,Dom,Con] consisting of

(i) a list Var = [X1,...,Xn] of variables Xi,

(ii) a list Dom = [D1,...,Dn] of lists Di (for variables Xi), and

(iii) a list Con of constraints mentioning variables from Var.

Let us agree that a goal node is a list [d1,...,dn] of members di of Di so that every constraint in Con is satisfied by instantiating Xi to di (for each i from 1 to n). More precisely, let us assume there is an *n*-ary predicate `satisfyCon/n` such that

```
goal([X1,...,Xn]) :- satisfyCon(X1,...,Xn).
```

How can we define

`Start`, `arc`, `cost` and `h`

so that calling A-star on `Start` searches for a goal node by backtracking from partial instantiations of variables Xi (for some i from 1 to n)? You need *not* give full definitions in Prolog. Concise descriptions in English are acceptable.

[20 marks]

Question Q2 If the aim of search is to find a goal node, the problem addressed by Markov decision processes (MDPs) is to make intelligent moves, understood in terms of arcs in a graph or actions. An action a_i may be followed by another a_{i+1}, yielding a sequence

$$a_1, a_2, \ldots, a_n, a_{n+1}, \ldots \tag{1}$$

of actions that may stretch indefinitely, along with a sequence

$$r_1, r_2, \ldots, r_n, r_{n+1}, \ldots \tag{2}$$

of real numbers r_i representing the immediate reward for performing action a_i at some state s_{i-1} leading to state s_i. That is, sequences (1) and (2) arise from a sequence

$$s_0 \xrightarrow{a_1} s_1, r_1 \xrightarrow{a_2} s_2, r_2 \cdots \xrightarrow{a_n} s_n, r_n \xrightarrow{a_{n+1}} s_{n+1}, r_{n+1} \cdots \tag{3}$$

of transitions $s_{i-1} \xrightarrow{a_i} s_i, r_i$ from state s_{i-1} to state s_i via action a_i with immediate reward r_i (for $i \geq 1$).

(a) What transitions $s_{i-1} \xrightarrow{a_i} s_i, r_i$ are possible in an MDP $\langle S, A, p, r, \gamma \rangle$?

[5 marks]

(b) What probability and reward functions p and r do the transitions in (3), taken together, suggest? Is there anything about (3) that could increase or decrease our confidence in this suggestion? Why might it be useful to consider more than one sequence (3) when trying to determine the functions p and r by sampling?

[15 marks]

Given an MDP $\langle S, A, p, r, \gamma \rangle$, the γ-**discounted value** of an action $a \in A$ at a state $s \in S$ is

$$Q(s, a) = \sum_{s' \in S} p(s, a, s')(r(s, a, s') + \gamma \max_{a' \in A} Q(s', a'))$$

which can be computed as $Q(s, a) = \lim_{n \to \infty} q_n(s, a)$ from iterates

$$q_0(s, a) := \sum_{s' \in S} p(s, a, s') r(s, a, s')$$
$$q_{n+1}(s, a) := \sum_{s' \in S} p(s, a, s')(r(s, a, s') + \gamma \max_{a' \in A} q_n(s', a')).$$

(c) For $S = \{s_1, s_2\}$ and $A = \{a_1, a_2\}$, compute the γ-discounted value $q_1(s_1, a_2)$ for $\gamma = \frac{1}{3}$ with probabilities p and immediate rewards r given by tables

a_1	s_1	s_2
s_1	.6, 7	.4, 0
s_2	.5, 0	.5, 3

Table a_1

a_2	s_1	s_2
s_1	.7, 0	.3, 15
s_2	.5, 0	.5, 2

Table a_2

specifying the pair $p(s, a_i, s'), r(s, a_i, s')$ at row s, column s' of Table a_i (e.g., $p(s_1, a_1, s_2) = .4$ and $r(s_2, a_2, s_2) = 2$).

[15 marks]

(d) For some MDPs, $Q(s, a)$ can be computed directly without resorting to iterates $q_n(s, a)$. Consider the MDP $\langle S, A, p, r, \gamma \rangle$ where $S = \{s_1, s_2, s_3\}$, $A = \{a_1, a_2\}$, $\gamma = \frac{1}{2}$ and p, r are given by

a_1	s_1	s_2	s_3
s_1	0,0	1,2	0,0
s_2	0,0	0,0	1,2
s_3	0,0	0,0	1,4

Table a_1

a_2	s_1	s_2	s_3
s_1	0,0	0,0	1,1
s_2	0,0	0,0	1,2
s_3	0,0	0,0	1,4

Table a_2

Compute $Q(s, a)$ for all $s \in S$ and $a \in A$.

[15 marks]

Question Q3 For each of the statements (a) to (e) below, answer "True" or "False" and briefly justify your answer.

(a) The exploration-exploitation trade-off in Q-learning depends on the learning rate α.

[10 marks]

(b) For a propositional knowledge base in Datalog (where all predicates have arity 0), any mechanical procedure for logical consequence is bound to be incomplete if it is goal-directed (as in Prolog). true

[10 marks]

(c) Abduction is the inverse of deduction. false

[10 marks]

(d) In a Bayes net, random variables are ordered so that causes come before their effects.

true

[10 marks]

(e) Conditional independences in a naive Bayes classifier are lost when its Bayes net is moralized to a Markov net. true

[10 marks]

Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

DECLARATION

I understand that this is an individual assessment and that collaboration is not permitted. I have not received any assistance with my work for this assessment. Where I have used the published work of others, I have indicated this with appropriate citation.

I have not and will not share any part of my work on this assessment, directly or indirectly, with any other student.

I have read and I understand the plagiarism provisions in the General Regulations of the University Calendar for the current year, found at http://www.tcd.ie/calendar.

I have also completed the Online Tutorial on avoiding plagiarism 'Ready Steady Write', located at http://tcd-ie.libguides.com/plagiarism/ready-steady-write."

I understand that by returning this declaration with my work, I am agreeing with the above statement.

Name:

Date: