Department of Electrical and Computer Engineering

ECE-457A Adaptive and Cooperative Algorithms
Final Examination
Spring 2021

Instructor Dr. Otman A. Basir

Student Name: Student ID: Section:

Exam start time: 12:00 PM(Eastern), Thursday, August 12, 2021
Exam end time: 12:00 PM(Eastern), Monday, August 16, 2021
Duration: 27 hours

Exam type: Open-book [You must not form any group even though Crowdmark may show you an option to invite group members]

Other instructions:

- **As you write the exam, you must maintain academic integrity. This is important for learning.**
- **For this exam, you must not communicate with or seek help from anyone.**
- **You must also not share this exam or any part of it in any form with anyone who has not yet submitted this exam or a replacement of it for grading.**
- **By submitting the exam for grading, you confirm that you have followed the above rules and committed no academic offences such as cheating and plagiarism.**
- **Penalty for late submission – 5 % per minute.**
- **Answer all questions in this exam. Clearly state any assumptions you make when answering any questions.**
- **Show your work—no magic answers are acceptable. Use additional pages as needed.**
- **Upload your responses to Crowdmark (1st option), Learn: Final Exam DropBox(2nd option) or email to obasir@uwaterloo.ca (as a last resort)**

- **If you believe the exam has errors/mistakes, please bring them to the instructor's attention by sending an email to obasir@uwaterloo.ca. However, do not contact the instructor seeking clarification of questions. Please, do not ask for clarification. If in doubt make assumptions.**

  You can document your answers in hand writing, word processing, ect. You can submit as a pdf, or scanned images, etc.

Question:	1	2	3	4	5	6	7	Total
Points:	25	10	15	10	15	15	10	100
Score:								

Question 1 (points)

(a) True/False

i. (1 point) ___ A* search will always expand fewer search nodes than uniform cost search.

ii. (1 point) ___ ES might be more appropriate for a problem where exploration is needed, but a GA might be more appropriate for a problem where exploitation is needed.

iii. (1 point) ___ A heuristic that always evaluates to h(s) = 1 for non-goal search nodes s is always admissible.

iv. (1 point) ___ Breadth-first search and iterative-deepening search always find the same solution.

v. (1 point) ___ MDP instances with small discount factors tend to emphasize near-term rewards.

vi. (1 point) ___ Breadth-first search is always a complete search method, even if all of the actions have different costs.

vii. (1 point) ___ If two admissible heuristic functions evaluate the same search node n as $h_1(n) = 6$ and $h_2(n) = 8$, we say h_1 dominates h_2, because it is less likely to overestimate the actual cost.

viii. (1 point) ___ A$*$ search will always expand fewer search nodes than uniform cost search.

ix. (1 point) ___ True / False – Value Iteration always find the optimal policy, when run to convergence.

(b) (3 points) Which search algorithm (BFS, DFS, UCS, A*, ID, etc.) best managed the worst-case asymptotic time and space complexity tradeoff? How did it achieve this result?

(c) (4 points) You are given sentences in a language that you know nothing about, where the word order has been lost. Given the words, you are requested to put them back in a sequence, that has a sensible meaning.

For example, in English language, the words { terrible, exams, Dr-Basir, are, not, but, fun } can be put together in this valid ordering: "exams Dr-Basir are terrible but not fun"

As stated, the language used is completely unknown to you. Luckily, you can use a function that assigns a score to every sequence you introduce, according to how ridiculous it is (for example, in English, the sequence "exams not fun" is considered less ridiculous than "fun not exams"!).

i. Given a set of n words, what is the problem search space size? (you may assume that all words are unique.)

ii. Define a neighborhood (a 'moveset', that can be reached in a single step of search). Give two examples of neighbors for the sequence "fun exams term are mid".

iii. Will hill-climbing search always find a valid sentence?

iv. Will Tabu Search always find a valid sentence?

(d) (4 points) Given the following two parents with permutation representation:
$p1 = (475318692)$
$p2 = (524836971)$

i. Compute the first offspring with Cycle Crossover.

ii. Compute the first offspring with PMX, using crossover points between the 2nd and 3rd loci and between the 6th and 7th loci.

(e) (2 points) We have two parents in a binary GA, each with n bits. The i -th bit in parent 1 is different than the i -th bit in parent 2 for $i \in [1, n]$. We randomly select a crossover point $c \in [1, n]$. What is the probability that the children are clones of (that is, identical to) the parents?

(f) (3 points) In what type of problem would exploration be more desirable, and in what type of problem would exploitation be more desirable? Justify tour answer.

Question 2 (points)

Sam finally defeats the dragon, but the cave starts the crumble. Sam flees from the cave and reconvenes with Sandor. Sandor asks Sam if there was any treasure in the cave. Sam replies that there was a lot of treasure, but that he didn't bring any back. Before Sam even finishes his sentence, Sandor rushes into the cave to see what treasure he can salvage before the cave collapses.

Sandor pulls out the treasure map of the cave, shown below. There are 6 treasures, and the location of each is marked with a number, which represents the utility of getting that treasure. Upon moving into a treasure square, Sandor immediately picks it up. Each treasure can only be picked up once. The square labeled T marks Sandor's starting location. Assume there is no discounting ($\gamma = 1$), and there is no time penalty. Sandor may only take one of the actions (North, South, East, West) and all actions are deterministic. To survive, Sandor must get back to his starting location by the stated maximum number of timesteps left (e.g. if two timesteps are left, Sandor has time only to move one cell and come right back). If he fails to get back to his starting location, his utility is $-\infty$. The game ends when (1) Sandor makes it back to his starting location or (2) the maximum number of timesteps has passed.

The map also indicates that a boulder could be present in the squares marked with the letter B in the map. The presence of a boulder means you cannot move onto the boulder square. Sandor doesn't know if the boulder is actually in the maze or not; he observes whether it is present or not if he moves to a square adjacent to the boulder (B). The boulder is present with probability 0.5.

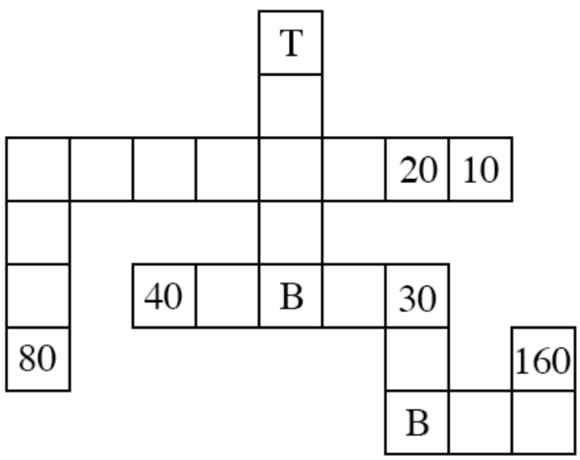


Figure 1:

Sandor wants to maximize the sum of utilities gained. Let S_K be the starting state for Sandor when he has just entered at position T and there are K timesteps left. For each scenario, calculate the optimal $V^*(S_K)$ values.

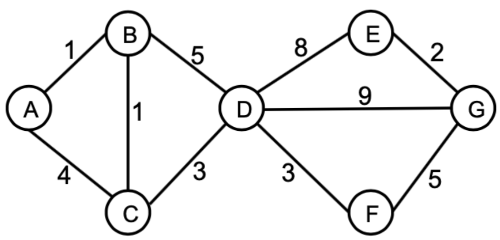
(a) (2 points) [1pt]$V^*(S_9) =$

(b) (3 points) [2pts]$V^*(S_{13}) =$

(c) (5 points) [2pts]$V^*(S_\infty) =$

Question 3 **(points)**

Consider the state space graph shown above. A is the start state and G is the goal state. The costs for each edge are shown on the graph. Each edge can be traversed in both directions. Note that the heuristic h_1 is consistent but the heuristic h_2 is not consistent.



Node	h_1	h_2
A	9.5	10
B	9	12
C	8	10
D	7	8
E	1.5	1
F	4	4.5
G	0	0

(a) (8 points) Possible paths returned For each of the following graph search strategies, mark which, if any, of the listed paths it could return. Note that for some search strategies the specific path returned might depend on tie-breaking behavior. In any such cases, make sure to mark all paths that could be returned under some tie-breaking scheme.

Search Algorithm	A-B-D-G	A-C-D-G	A-B-C-D-F-G
Depth first search			
Breadth first search			
Uniform cost search			
A* search with heuristic h_1			
A* search with heuristic h_2			

(b) (7 points) Heuristic function properties Suppose you are completing the new heuristic function h_3 shown below. All the values are fixed except $h_3(B)$. For each of the following conditions, write the set of values that are possible for $h_3(B)$. For example, to denote all non-negative numbers, write $[0, \infty]$, to denote the empty set, write $\varnothing$, and so on.

Node	A	B	C	D	E	F	G
h_3	10	?	9	7	1.5	4.5	0

i. What values of $h_3(B)$ make h_3 admissible?

ii. What values of $h_3(B)$ make h_3 consistent?

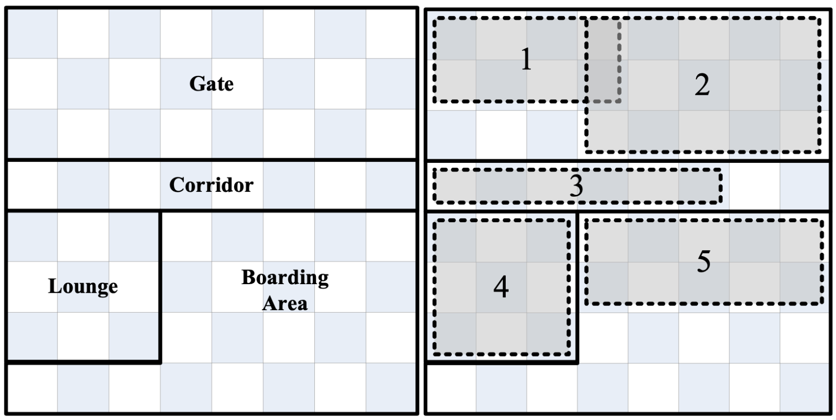
iii. What values of $h_3(B)$ will cause **A*** graph search to expand node **A**, then node **C**, then node **B**, then node **D** in order?

Question 4 (points)

(a) (2 points) What should be observed in selecting the initial and final temperatures of the SA cooling schedule? Provide guidelines to select these temperatures.

(b) (3 points) Mention some of the approaches that have been proposed to improve the computational cost of the basic SA method.

(c) (5 points) In a simple sensor placement problem, it is required to place 5 pan-tilt cameras in such a way that maximizes the coverage of an airport. The airport consists of 8×8 cells divided into four areas of interest, namely: gate, corridor, lounge and boarding area as illustrated in Figure 2 a, The gate area is considered as airport perimeter sensitive area, and accordingly uncovered cells in this area are assigned double the cost of uncovered cells in other areas. The sensing range of each camera is given in Table in Figure 3 . This sensing range represents the union of all fields of view that a pan-tilt camera can view. Figure 1-b shows a possible placement for the cameras that is obtained by placing the cameras in the areas in sequence.



(a) Airport Plan (b) Possible Placement of Cameras

Figure 2:

Camera	Sensing Range	
	L1 (Cells)	L2 (Cells)
1	2	4
2	3	5
3	2	6
4	3	4
5	2	7

Table 1 – Cameras Sensing Ranges

Figure 3:

- Any camera can be placed to get a sensing range of L1-by-L2 or L2-by-L1
- The sensing ranges of two cameras may intersect.

i. Define a suitable problem representation.
ii. Define suitable neighborhood operators,
iii. Define the objective function for calculating the cost of a solution,
iv. Starting from the solution in Figure 2 b, apply simulated annealing for 2 iterations using an initial temperature of 500, a linear decrement rate of 50, and one iteration at each temperature. Assume the following random numbers in sequence $[0.0071, 0.6799, 0.9027]$.

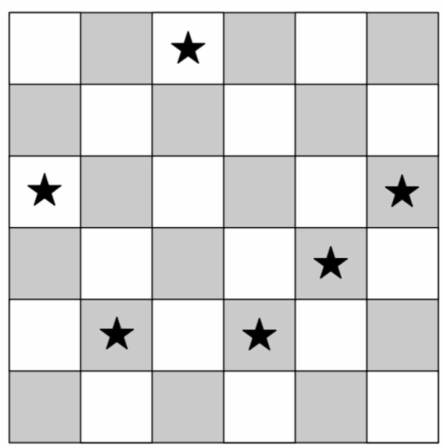
Question 5 (points)

The N-queens problem requires you to place N queens on an N -by-N chessboard such that no queen attacks another queen. (A queen attacks any piece in the same row, column or diagonal.) Here are some important facts:
- We define the states to be any configuration where the N queens are on the board, one per column.
- The moveset includes all possible states generated by moving a single queen to another square in the same column.
The function to obtain these states is called the successor function.
- The evaluation function Eval(state) is the number of non-attacking pairs of queens in this state. (Please note it is the number of NON-attacking pairs.)
In the following questions, we deal with the 6-queens problem (N = 6).

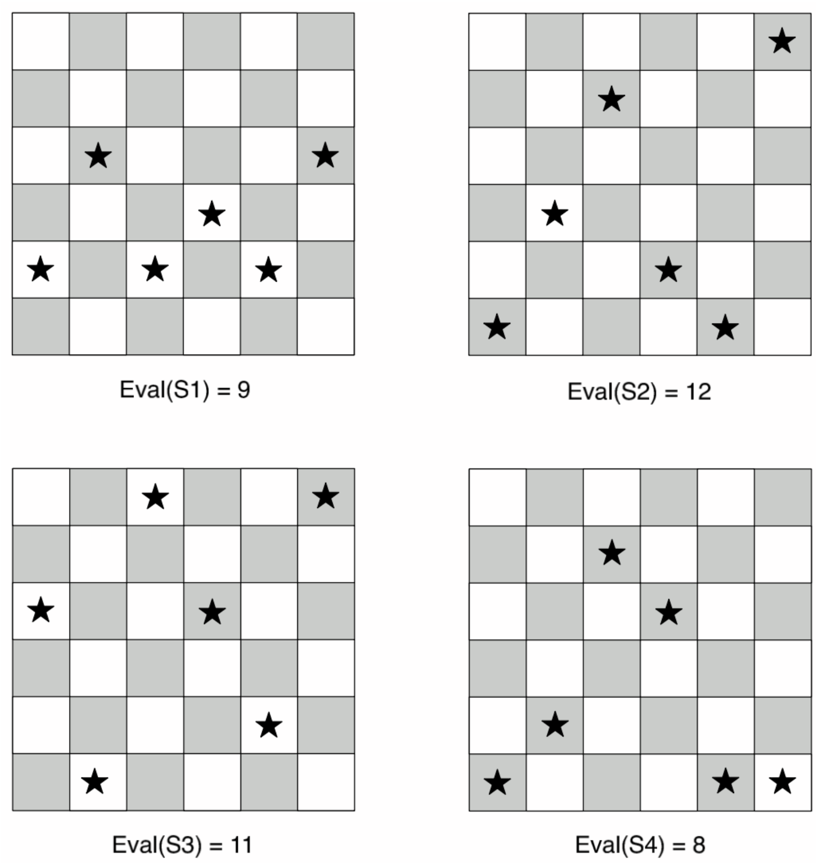
(a) (2 points) How many possible states are there totally?

(b) (2 points) For each state, how many successor states are there in the moveset?

(c) (3 points) What value will the evaluation function Eval() return for the current state shown below?



(d) (4 points) Suppose you use a Genetic Algorithm. The current generation includes four states, $S1$ through $S4$. The evaluation values for each of the four states are: $\text{Eval}(S1) = 9, \text{Eval}(S2) = 12, \text{Eval}(S3) = 11, \text{Eval}(S4) = 8$. Calculate the probability that each of them would be chosen in the "selection" step (also called "reproduction" step).



(e) (4 points) In a Genetic Algorithm, each state of 6-queens can be represented as 6 digits, each indicating the position of the queen in that column. Which action in genetic algorithm (among { selection, cross-over, mutation }) is most similar to the successor function described above?

Question 6 (points)

(a) (3 points) What is the difference between the concept of mutation in evolutionary algorithms and the velocity update in particle swarm optimization?

(b) (2 points) How does Evolution Strategies achieve adaptation?

(c) (10 points) Simple ACO is being used in a traveling salesman problem. Assume 4 cities $\{A, B, C, D\}$, which are represented by a fully connected graph. The following tables represent the pheromone levels on each edge of the graph and the distances between each city (assume the pheromone levels and distances are symmetric).

	Pheromone Levels			
	A	B	C	D
A				
B	0.25			
C	0.11	0.98		
D	0.34	0.54	0.67	

	Distances			
	A	B	C	D
A				
B	12			
C	10	6		
D	8	15	3	

i. Assume an ant started its journey at city A and has travelled to city C. Assume alpha and beta are set to 1 . Using the formula described in the lectures

a) What is the probability that the ant will travel to city A ?

b) What is the probability that the ant will travel to city B ?

c) What is the probability that the ant will travel to city D ?

ii. Assume the number of ants is initialized to 5 . At each iteration of applying ACO how many times the update rule is used if
a) Online step-by-step update rule is applied

b) Online delayed update rule is applied

c) Offline update rule is applied (Global update)

iii. Online step-by-step update rule can use an ant density model or an ant quantity model. Briefly explain the difference between these two models.

Question 7 (points)

(a) (5 points) In PSO, each particle shares its personal best information with other particles. Selecting a proper neighborhood affects the convergence and also helps in avoiding getting stuck at local minima. What topology each statement below is describing? Choose from the list:
List: Star topology, ring topology, Von Neumann Model, physical neighbors, social neighbors

i. Choose the particles that are the closest to particle m in the search space. The notion of closest is based on the distance in the cartesian space. This approach is computationally expensive.

ii. This topology has the fastest propagation of information in a population. Particles can easily get stuck in local minima.

iii. The particles are kept in a matrix data structure. Particles next to particle m in the matrix are picked as neighbors.

iv. The most successful neighborhood structure which is formed by arranging the particles in a grid and connecting each to 4 other particles.

v. The propagation of information is the slowest but does not easily get stuck in local minima. It might increase computational cost.

(b) (5 points) PSO can be used for permutation problems by re-defining three operations: adding velocity to a position, subtracting two positions, and multiplying a velocity by a constant. Suppose that particle m is currently at position X1 and the velocity is V1. Calculate the updated velocity V2 and position X2 given the information below:
X1 = [14285673]
V1 = $\{(1,3),(5,8),(2,7)\}$
Pbest = [16285473]
Nbest = [14275683]
$\mathrm{W} = 1/3, \mathrm{c1} = \mathrm{c2} = 2, \mathrm{r1} = \mathrm{r2} = 0.5$