National University

Of Computer & Emerging Sciences Chiniot- Faisalabad Campus

AI 2002 – Artificial Intelligence (Spring 2024)

Assignment 3

<table>
<tr><td><u>Topics Covered:</u> Local searches, adversarial searches</td><td><u>Submission Deadline:</u> Friday – March 22, 2024 by23.00 sharp</td></tr>
<tr><td colspan="2"><u>Submission Guidelines:</u>
• Group assignment: 2 members (at max.)
• Submit your source code and a PDF file on the Google Classroom (of your respective section).
• Students are expected to submit their own code and answers, any help from internet/chatgpt/fellows is not allowed.</td></tr>
</table>

Basic overview:

In this assignment, students will work on two environments: “nqueen” and “tictactoe” implemented with a graphical user-interface. The code repository is available at Github Repository, and is adapted from the AIMA, necessary changes are made by the teaching assistants to meet our needs.

Credits: Saad Zafar, Dawood Ahmad, Asad Ullah

Question 1

Implement the following local search methods on the “8-Queen” environment, see points (a) to (c) for some guidelines.

i. Hill climbing
ii. Stochastic hill climbing
iii. Simulated annealing (for temperature schedule: see part (f) below)

a. The objective function is the total number of queens attacking each other at time t. Implement this function, that is, given an 8-Qeen board configuration, compute its value. By running local search algorithm, we want to find the global minimum i.e., a state with objective function value of 0 (or no queen attacks the other).

b. Implement a successor function which will generate a new board configuration by randomly moving any queen to another place in its column.

c. To compare the performance of the three algorithms, it is necessary that all algorithms start from the same random state. That is, generate a state randomly with a controlled seed value and regenerate the same random state to run the second algorithm, and so on.

d. Compare the three algorithms based on:
   - Time complexity
   - Optimality (which algorithm found a global minimum or a local minimum or a plateau)

   Write your answers in a table, along with the seed values (for reproducibility).

e. Perform 10 runs of each algorithm for a detailed analysis, each with a completely random initial state (store the seed value for reproducibility).
   - Follow the instructions stated above in part (b) and (c).
   - For part (c), you table will contain the results of 10 runs for the three algorithms.
   - Based on above, which algorithm has performed the best overall? Write your answer and provide reasoning.

f. Temperature schedule for simulated annealing: Python has a library called "mlrose" for implement different optimization algorithms. The following link provides information of the available temperature functions (https://mlrose.readthedocs.io/en/stable/source/decay.html), use *ExpDecay* in your code with default values.

   A tutorial for solving the TSP problem using mlrose is available at: https://mlrose.readthedocs.io/en/stable/source/tutorial2.html. DO NOT use algorithms from this library, you will implement all local searches yourself.

Question 2

Implement the following adversarial methods on the "TicTacToe" environment, the game will be played between two computer agents: (a) *rational agent:* will always maximizes its utility and will take the best move, (b) *random agent:* will take a move randomly between the "two best" moves, that is, among the two moves with the lowest utility, it will make a move randomly. For leaf nodes (when the tictactoe is filled), set the utility=1 if it is a win for the rational agent, utility = -1 if the random agent wins, and utility=0 for a draw. We assume that the rational agent takes the first turn.

i. Minimax algorithm
ii. Alpha-beta pruning

Compare the performance of the above algorithms based on:

a. Space complexity
b. Time complexity
c. Who won the game? Rational agent or random agent?

Write your answers in a table.