272SM: Introduction to Artificial Intelligence
Homework Assignment 3: CSP's, Games, MDP's

Monday 27th November, 2023 - Tuesday 12th December, 2023

Instructions

Report your answers to the following exercises and submit them. Submitting homework solutions is highly recommended; Correct solutions will be counted as bonus points towards the final grade. You can team up with your colleagues into groups of max. 5 people.

CSPs

Exercise 1: Formulating a CSP

Suppose you have a state-space search problem defined by the usual stuff:

- a set of states s;
- an initial state s0;
- a set of actions A including the NoOp action that has no effect;
- a transition model Result(s, a);
- a set of goal states G.

Unfortunately, you have no search algorithms! All you have is a CSP solver. How could you reformulate this as a CSP? You may assume that you are given the maximum number of steps, T that any plan can have. Make sure that your formulation makes it easy to see what the plan is.

Exercise 2: Algorithmic map coloring - Performance analysis

Generate random instances of map-coloring problems as follows: scatter n points on the unit square; select a point X at random, connect X by a straight line to the nearest point Y such that X is not already connected to Y and the line crosses no other line; repeat the previous step until no more connections are possible. The points represent regions on the map and the lines connect neighbors. Now try to find k-colorings of each map, for both k=3 and k=4, using min-conflicts, backtracking, backtracking with forward checking, and backtracking with arc consistency. Construct a table of average run times for each algorithm for values of n up to the largest you can manage. Comment on your results.

Exercise 3: Critical ratio

Using a CSP solver program and another program to generate random problem instances of CSPs, report on the time to solve the problem as a function of the ratio of the number of constraints to the number of variables.

Games

Exercise 1: Minimax tree pruning

In a full-depth minimax search of a tree with depth D and branching factor B, with alpha-beta pruning, what is the minimum number of leaves that must be explored to compute the best move?

Exercise 2: Simulating tic-tac-toe

Describe

- state descriptions
- move generators
- terminal tests
- utility functions
- evaluation functions

for tic-tac-toe game. Look at the implementation available at https://github.com/aimacode/aima-python/blob/master/games4e.ipynb. Illustrate the average score of games between a player who is using alpha-beta pruning, a player using minmax search, and a random player.

Multi-agent Decision Making

Exercise 1: Gridworld navigation

For the 4x3 world shown in Figure 1, calculate which squares can be reached from (1,1) by the action sequence [Right, Right, Right, Up, Up] and with what probabilities.

Hint: compute the occupancy probabilities at each step by filling in Table .

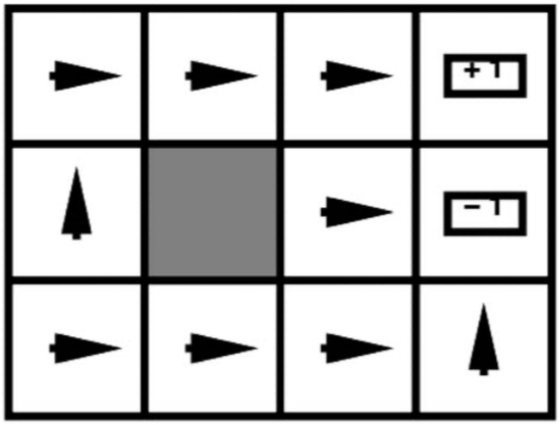

Figure 1: Gridworld

		Right	Right	Right	Up	Up
(1, 1)	1	.1	.02			
(1, 2)		.1	.09			
(1, 3)						
(2, 1)		.8				
(2, 3)						
(3, 1)						
(3, 2)						
(3, 3)						
(4, 1)						
(4, 2)						
(4, 3)						

tableAn example table

Exercise 2: Threshold Policy Optimization

For the environment shown in Figure 1, find all the threshold values for $R(s)$ such that the optimal policy changes when the threshold is crossed. You will need a way to calculate the *optimal policy* and its value for fixed $R(s)$.

Hint: Prove that the value of any fixed policy varies linearly with $R(s)$.