CS4700 Fall 2011: Foundations of Artificial Intelligence

Homework #2

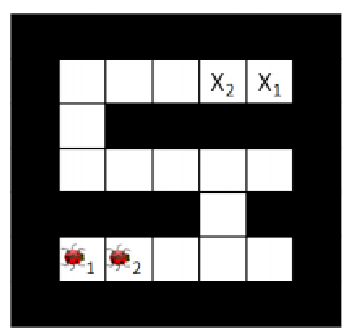
Due Date: Monday Oct 3 on CMS (PDF) and in class (hardcopy)

Submit paper copies at the beginning of class. Please include your NetID on the front of your assignment. Late submissions drop 10% each 24 hour period (minus slack days), until the solutions become available, at which point submissions will no longer be accepted. Your submission should include answers and an appendix with code printout. Code in `9pt courier` single spacing. Include only code you wrote and only for the solving the problem (no GUI code). Your assignments should reflect your individual work.

Question 1: Search Heuristics (30 Points)

A. A frog wants to jump from one side of the river bank to another side. The goal is to minimize the number of hops. There are N rocks connecting the two sides. The frog wants to cross the river as quickly as possible. It needs your help to come up with a plan. Here are the rules: In the first hop, it must jump to the next stone from the land; If in hop t-1 the frog jumped over X stones, then in hop t, the frog is allowed to jump X, X+1, X-1 or X+2 stones forwards or backwards; and in the last hop, it must arrive exactly at the destination.
   1. Describe how to construct the states for your state space, and the successor function in your state space.
   2. Is the number of stones left for the frog to jump an admissible heuristic? Briefly explain.
   3. State and justify a non-trivial, admissible heuristic, which are not the number of stones left for the frog to jump. Prove whether your heuristic with A* is admissible.

B. You control one or more insects in a rectangular maze-like environment with dimensions M x N, as shown. At each time step, an insect can move into an adjacent square if that square is currently free, or the insect may stay in its current location. Squares may be blocked by walls, but the map is known. Optimality is always in terms of time steps; all actions have cost 1 regardless of the number of insects moving or where they move. You control K insects, each of which has a special target ending location X_k. No two insects may occupy the same square. In each time step all insects move simultaneously to a currently free square (or stay in place); adjacent insects cannot swap in a single time step. Provide:



   1. State space description:
   2. State space size:
   3. Which of the following heuristics are admissible and why (if any)?
      i. Max of Manhattan distances from each insect's location to its goal.
      ii. Max of costs of optimal paths for each insect to its goal if it were acting alone in the environment, unobstructed by the other insects.

Question 2: CSP Formulation (20 Points)

Formulate the following problem as a CSP. Explain what variables you are using and their domains. Formulate all constraints using only Boolean operators (and, or, not, equality), and basic binary operations (+,-). You might also use the global constraint *alldifferent*($x_1,x_2,\ldots,x_n$), that ensures that the variables $x_1,x_2,\ldots,x_n$ take different values.

Your favorite horse race will take place tomorrow. Five horses will be racing: Look Behindyou, Showme Themoney, Ucan Doit, Catchme Ifyoucan, and Gotta Pee. As usual, all horses will be wearing a different color, and will have a different number. You also know that they all have a distinctive physical characteristic, although you do not know which.

You plan to place a bet, and you wander around in the stable, hoping for some insider information. You talk to twelve horse owners and, luckily, each of them reveals to you a precious piece of information. Here is the report of what they tell you:

Owner 1) “Catchme Ifyoucan”, who is not the number 13, is not the one with big ears.
Owner 2) The blue horse will arrive right before “Gotta Pee”.
Owner 3) The spotted horse is not the number 10.
Owner 4) The horse with the short tail is not called “Catchme Ifyoucan” and is not the number 2.
Owner 5) “Showme Themoney” will arrive immediately before the horse with the number 13, and the horse with the number 13 will arrive immediately before the purple horse.
Owner 6) The green horse is not the number 10.
Owner 7) “Gotta Pee” is not the number 7.
Owner 8) The tall horse is not called “Gotta Pee” and is not purple.
Owner 9) The horse with big ears is not red.
Owner 10) “Showme Themoney” is not gold.
Owner 11) The tall horse, “Catchme Ifyoucan”, and the horse with the number 2 will arrive in consecutive order, first-to-last.
Owner 12) “Ucan Doit”, the gold horse, the horse with the short tail, the horse with the number 8, and the horse that will finish last, are in fact different horses.

Given all these details, can you determine the order in which the horses will arrive?

Question 3: CSP Manual Analysis (20 Points)

Mr. Higgins is the manager of the most popular zoo that’s part of a rising zoo franchise “wild beasts R us”. He recently got into a situation that’s somewhat complicated and will need your help to figure things out. His zoo is expanding by buying some land from the neighbors and the board of shareholders has determined that three new sites need to be built each of which needs to hold exactly two different kinds animals. They have also decided that they are going to populate the new sites with the following six kinds of animals: lions, tigers, jaguars, leopards, black panthers and bobcats.

Normally, Mr. Higgins would have no problem finding a way to accommodate all the animals. However, the 4 biggest shareholders have placed some constraints because on their children's phobias. Mr. Higgins has to accommodate these constraints because if the children of these shareholders visit the zoo and they are afraid to visit all of the newly built sites or they can only visit the ones they are not interested in they will complain to their parents and then Mr. Higgins risks losing his job. Colleen is the daughter of the biggest shareholder Mr. Oag. She's afraid of bobcats and tigers. Furthermore she wants to be able to see the lions. Alex is the son of the second biggest shareholder and is afraid of tigers, jaguars and black panthers. Madison is the daughter of the third most influential shareholder and she is afraid of lions, bobcats and leopards. Finally Tom is the son of the fourth biggest shareholder and he is afraid of lions, jaguars, and bobcats. Can you help Mr. Higgins formulate the correct pairs of animals for the 3 new sites so that every child can visit at least one of the newly built sites?

In any constraint satisfaction problem, there may be ambiguity as to what variable is assigned next. Whenever this is the case, assign variables in alphabetical order. Similarly, assume values are tested in order, i.e., 1 before 2 before 3.

a. Redefine this problem more formally as a Constraint Satisfaction Problem. Explain what variables you are using and their domains, and formulate all constraints using only the Boolean operators $\vee$(or), $\wedge$(and), $\neg$ not), and = (tests equality of variable to a value or another variable). In situations where there are many of the same type of parallel rule, describe the general form of the rule with Boolean operators instead of listing all of the instances of it.
b. Perform a basic backtracking search for the problem. Demonstrate the process by drawing the search tree created previous to a solution being found. Clearly indicated which nodes are discarded due to them being ruled out (by direct violation of constraints, or indirectly through backtracking). Be careful not to use inference not given directly by the listed constraints. Also be sure to use the ordering rules listed above.
c. Solve the problem again, but this time use forward checking. Give your answer in the form of a table as follows: each line (row) adds a new tested assignment. Each column is a variable, with all of its remaining available values (at that step) listed. Each row will contain all of the available values for all of the variables for the current set of assignments. Circle variable values that have been assigned. In the event that a line is discarded due to backtracking, put an arrow from the line where backtracking starts, back to the point where the backtracking ends.
d. For this specific problem, does adding the minimum remaining value heuristic to your forward checking save time? Describe as precisely as possible why or why not.

Question 4: Coding Local Search (30 Points)

A set of M professors and N students are at a Computer Science cocktail party. The party is held in 10 meter by 10 meter square room, which is conveniently tiled with one hundred 1 m^2 tiles.

Being socially-awkward computer scientists, each student prefers to stand as far as possible from each professor, and vice versa. Further, the professors have been recently embroiled in a bitter fight over a corner office and prefer to stand as far as possible from each other. The students don't care how far apart they are standing. Assume both the students and professors stand exactly in the center of their own tiles.

The discontent created by a professor / student pair is described by

$$C = 1 / d(s_i, p_j)$$

for student s_i and professor p_j, where d(x, y) is the Euclidian distance between x and y. Similarly, the discontent for each professor / professor pair is

$$C = 1 / d(p_i, p_j)$$

The discontent of the room as a whole is the sum of the discontent created by each pair.

You are to write a computer program that minimizes the total discontent by choosing locations for all professors and students, given the following number of each:

Problem A: M = 1, N = 1 (an awkward party, by any measure)
Problem B: M = 3, N = 10
Problem C: M = 15, N = 10
Problem D: M = 20, N = 20

As with many local search problems, it will not be possible to prove that your solution is the global optimum, but you are to find as good of a solution as you can, using the following forms of local search: random restart hill climbing and simulated annealing. The goal of the search is to find as happy a room as possible.

Programming and submission tips: Don't create anything more complicated than you need to. This is a simple program. Implement it simply. Help your grader help you: be clear with comments and naming. In the printout, highlight every function declaration in bold and have a brief comment that explains what it does. Do not include printout of code you didn't write or of code that is not part of solving the problem, e.g. GUI (you can include screenshots if it's critical). The shorter the printout, the better.

Describe the general characteristics of your algorithm. You should be able to get away with about 10 word responses for each. Using more than 50 words on each response is strongly discouraged.

a. What are you using for a state?
b. How are you setting your initial state?
c. Given a state, what next states do you consider?
d. How do you compute the value of a state?
e. What temperature schedule are you using for simulated annealing?

In addition to these responses, hand in complete printouts of your source code, and the output of your program for each of the 4 sets of professors and students. This last should include, for random restart and simulated annealing,

- The total discontent
- The number of states visited

- An ASCII-art depiction of the locations of professors and students. Use an S for student, P for professor, and . (a period) for unoccupied locations
- For random restart, the number of times you ran the search from scratch.

For example, one (fairly bad) solution for A might be

```
Random restart
. . . . . . . . . .
. . . . . . . . . .
. . . . . . . . . .
. . P . . . . . . .
. . . . . . . . . .
. . S . . . . . . .
. . . . . . . . . .
. . . . . . . . . .
. . . . . . . . . .
. . . . . . . . . .
Discontent = 0.5
States evaluated = 50
Restarts = 10
```