Midterm: Th. Feb.16, 2pm - 3pm

Intro AI ICS 171
Instructor: Max Welling

- *This exam is closed book*
- *Spend your time wisely: get a shot at each of the 3 questions.*
- *You can get a total of 60 points.*
- *Good Luck !*

1.(10 pts) **Warming up** Answer the following questions with True or False.

a.(5pts) True/False: A rational agent will always achieve its goal.

a) answer: False: due to the unpredictability of the world it can only maximize its expected utility and at any given instance may fail if something unexpected happens.

b.(5pts) True/False: Alpha-Beta pruning in minimax search has polynomial time complexity.

a) answer: False, still exponential unfortunately.

1.(30pts) **Search** Consider a fantasy chess piece called "*jumper*". It can move up, down, left, right, or it can stay wherever it is. Consider k such jumpers on an infinite chessboard at positions $s_1, s_2, ..., s_k$. The goal is to move these jumpers as fast as possible to positions $g_1, g_2, ...g_k$. In each move, you are allowed to move any number of jumpers simultaneously, but 2 or more jumpers can never occupy the same square.

a.(5pts) Formulate the above problem as a search problem, i.e. describe a state, the initial state, an action, the goal test and a path-cost.

a) answer: State: the position of the jumpers on the chessboard. Initial state: the given initial positions $s_1, ..., s_k$. Action: Any allowed move of k jumpers to new positions. Goal test: Have the jumpers reached the positions $g_1, ...g_k$? Path-cost: number of moves up to current position.

b.(5pts) Set $k = 1$, i.e. a single jumper on the board. What is the maximal branching factor for this problem?

b) answer: $b = 5$, the jumper can move into 4 positions or it can stay where it is.

c.(5pts) Still $k = 1$. Describe an admissible heuristic for this problem. This could be used inside an A^* search algorithm.

c.) answer: Manhattan distance between s_1 and g_1.

d.(5pts) Now consider general k. What is now the maximal branching factor? Remember that up to k jumpers can move simultaneously.

d.) answer: 5^k, because if all jumpers are sufficiently far away from each other, they can all move and each one has 5 choices, including not moving.

e.(5pts) Let $h_i \geq 0$ be an admissible heuristic for jumper i if it were alone on the board (i.e. $k = 1$). Which of the following heuristics are admissible for k jumpers simultaneously: I) $\max\{h_1, ..., h_k\}$, II) $\max\{h_1, ..., h_k\} - \min\{h_1, ..., h_k\}$? Explain! (Recall that jumpers are constrained and can move simultaneously).

e) answer: I) is admissible because you need at least as many moves as the maximum of all the heuristics. Since II) is always smaller than I) it is also admissible.

f.(5pts) Assuming for a moment that both I) and II) are admissible, which heuristic will expand fewer nodes (or an equal number of nodes) when used in an A^* algorithm?

f) answer: I dominates II, so it is better.

3.(20pts) **CSPs** We now consider k knights on a $n \times n$ chess-board. Our goal is to place them in such a way that no two knights are attacking each other (knights can jump into 8 possible positions in the shape of an "L"). Two or more knights can not occupy the same position.

Consider the following two variables and their domains for this problem:

I) Variable $X_i = [x_i, y_i]$ is the 2-D position of knight i, $x_i = [1, .., n]$, $y_i = [1, .., n]$, $i = 1..k$.

II) Variable $Y_{x,y}$ is the occupancy of square $[x, y]$, $Y_{x,y} = [0, 1]$, $x = [1, .., n]$, $y = [1, ..., n]$.

a.(5pts) We define 2 different CSPs where the variables and domains correspond to the variables and domains for I) and II) above. Formulate the constraints for both I) and II).

a) answer: I) No two positions can be such that they occupy the same position or attach each other. II) No two squares can be occupied if their knights attack each other. Moreover, the total number of squares which are occupied should be equal to k.

b.(5pts) We will approach this problem using local search. Define a cost or an objective function for both formulations I) and II).

b) answer: I) For instance: (Number of knights that attach each other) + (some penalty for more than two knights on the same square). II) Number of knights attacking each other + $|k - \sum_i Y_i|$ (last term to enforce that only k squares should have 1.

c.(5pts) Define in both formulations of the CSP what a complete state looks like and define the notion of a local move (a sequence of such moves should gradually increase the objective function.)

c) answer: I) k knights at some position on the board. A local move could be to change the position of a single knight. II) Some assignment of 0 or 1 to the squares. A local move can be flipping a single assignment of a square.

d.(5pts) Are local search algorithms in general guaranteed to find the optimal solution? Explain!

d) answer: No, local search can get stuck in local minima.