Grado en Ingeniería Informática

Departamento de Informática
Universidad Carlos III de Madrid

Artificial Intelligence

Jun 2014

General instructions

- Time assigned to the exam is **2 hours**
- Teachers will not answer any question about the exam
- You cannot leave the classroom during the exam, unless you have finished it
- Exams cannot be answered using a pencil

Theory questions (4 points)

We expect very short answers.

1 point What are the main steps of a production system cycle? What knowledge representation paradigm should you use for the working memory? In case you would have to define a production system to solve the problem of computing the best path from any location to another location in the world, given a map, what information would you represent in the working memory?

1 point Suppose you are asked to use a search technique to compute the best path in terms of cost to go in a grid of 100×100 cells from one cell to another one. The allowed moves are only horizontal and vertical, one cell at a time. Moving to some cells has a cost of two, while moving to other cells has a cost of three. What heuristic search algorithm would you use? Which heuristic function would you use?

1 point What are the key elements that you have to define if you want to use a bayesian network to solve a problem?

1 point Name three different sensors that are commonly used in robots and explain what can they be used for.

Problem (3 points)

In the magic square puzzle, there is a 3×3 grid that should be filled with digits from 1 to 9. The goal is to assign a digit to each cell, such that there is a different digit at each cell. Also, the sum of the digits in each row, column or diagonal should be 15. Figure 1 shows a grid configuration at a given intermediate state. In this state, the sum of the digits in the second row and second column is already 15.

2	9	
7	5	3
	1	

Figure 1: State of the puzzle.

Let us assume that we start a search process in that state to generate a solution. Actions have unit cost. Show the expanded search tree for the following algorithms. You have to clearly present the expansion order of nodes and the path to the solution. Perform repeated states pruning whenever possible.

1. Breadth-first search.

2. A*, using as heuristic: $h(n)$= *number of rows + number of columns + number of diagonals, whose sum is not 15*. For instance, $h()$ for the state in Figure 1 is $h(n) = 6$. Clearly print the values of f, h and g of each node.

3. Hill-climbing, using the previous heuristic.

Reply to the following questions:

- Is it possible to obtain the optimal solution with the three search strategies?
- In the general case, are these strategies complete and optimal? If some conditions should hold, which are them?
- In this particular case, and in relation to the $h()$ presented, is A* complete and guarantee the optimal solution?

Question on the practice. (3 points)

We want to develop a system that decides whether it is convenient or not to attack a country in the game of Risk, given the current state of the game. This decision will be taken by a Bayesian network. There are three variables that the system will provide to the Bayesian network as evidence, from which the Bayesian will have to make the decision (variable **Attack**, values YES/NO). It will have to take into account two factors: if the opponent is weak or strong, and if we have some advantage or not in the attack. The three variables are:

- Variable **Aggressive**, YES/NO: whether the opponent has recently adopted an agressive strategy. We know that strong players take agressive strategies in a 70% of cases, while weak players in only a 40% of cases.
- Variable **Allies**, YES/NO: whether the opponent has allies. We know that strong players have allies in a 20% of cases, while weak players do in a 50% of cases.
- Variable **Advantage**, YES/NO: whether we have numerical advantage for this attack. We know that "a priori", in a 25% of cases we have advantage. Also, it is advised to attack in a 75% of cases in case of a weak opponent when there is no advantage, and always if there is advantage. When the opponent is strong, one should only attack when one has advantage, and in a 60% of cases.

Provide an answer to the following.

1. Draw the Bayesian network and show the conditional probability tables at each node with the previous data. If there is any variable whose distribution does not appear in the previous explanations, say it and assume all its values are equiprobable.

2. Describe how the Bayesian network in the previous item represents/factorizes the following distribution of joint probability:

$$P(\mathit{Attack}, \mathit{Advantage}, \mathit{Opponent}, \mathit{Aggressive}, \mathit{Allies})$$

3. How would be the following expression solved (you do not have to substitute the numerical values)? We do not know whether there is advantage or not. Use inference by enumeration and take into account the properties of conditional independence.

$$P(\mathit{Attack} = YES|\mathit{Aggressive} = YES, \mathit{Opponent} = Weak)$$

4. With the values you have defined in your Bayesian network, perform the numerical computation and determine if the decision would be to attack or not.

Solutions

Theory questions

1. The answers are:

   (a) Matching, conflict resolution, and rule execution.

   (b) You can use any representation paradigm for the working memory.

   (c) The map, the initial location and possibly the destination location.

2. Any algorithm that would generate optimal solutions, as A*, or IDA*. Since the minimum cost to move from one cell to another one is 2, the heuristic function could be the Manhattan distance multiplied by 2.

3. Nodes (variables), edges (conditional dependencies among variables), and CPTs (conditional probability tables) at each node.

4. Examples are:

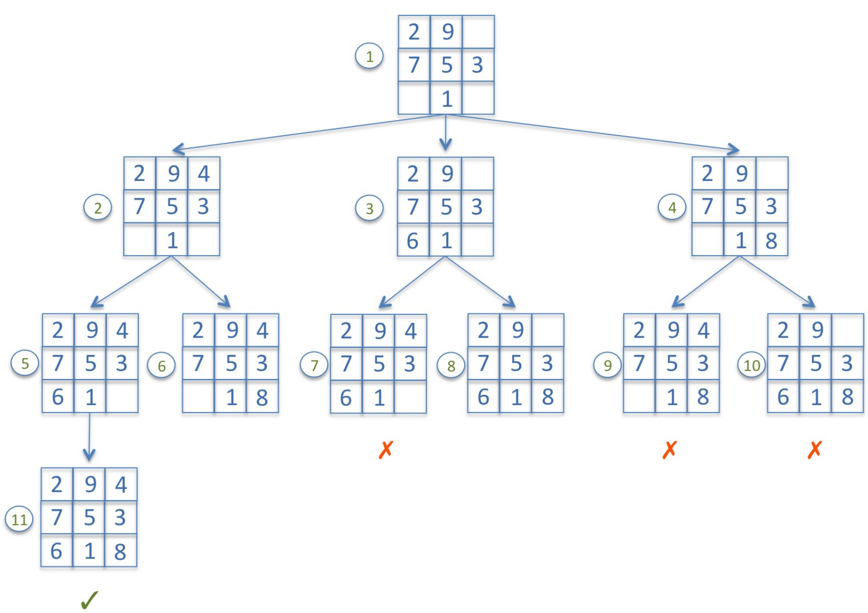
   (a) Cameras: for vision

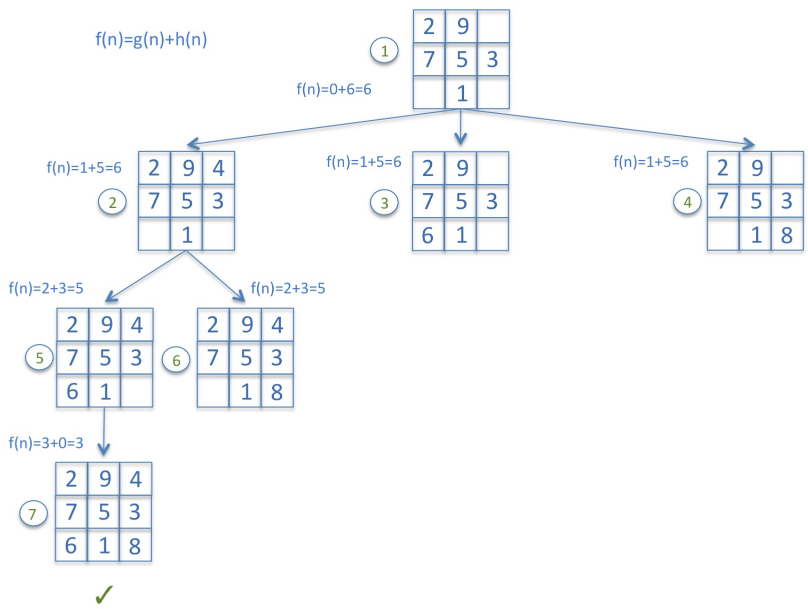
   (b) Infrared: to compute the distance to the closest obstacle

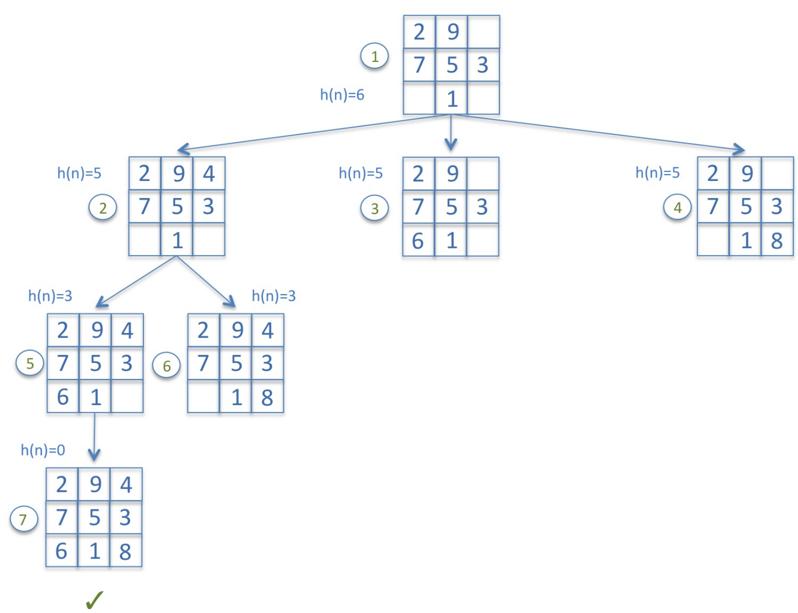
   (c) Laser: the same as above, but more precise

Problem

The corresponding trees are represented in the following figures.



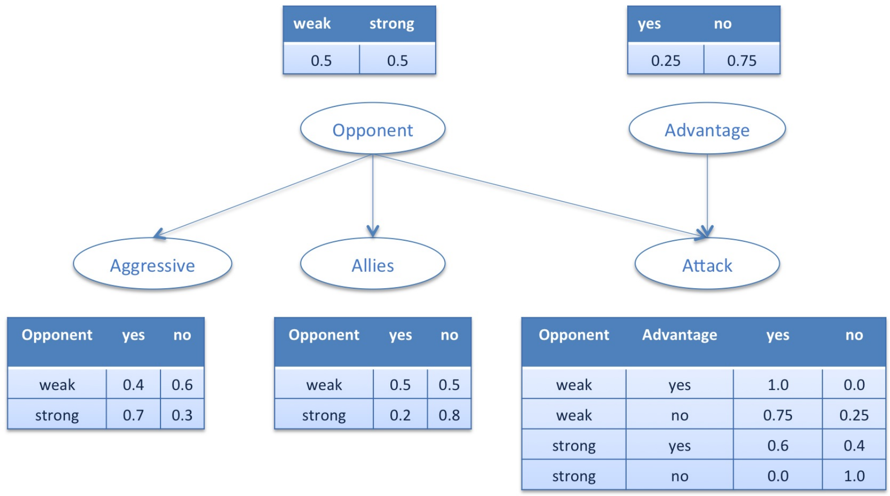




1. In this case, yes.
2. No. Breadth-first search and A* are both complete and optimal. Hill-climbing is not. Breadth-first requires unit cost and finite number of successors for optimality. A* requires finite number of successors, cost of edges greater or equal than a given threshold ϵ, such that $\epsilon > 0$, and an admissible heuristic ($h(n) \leq h^*(n)$ for all nodes n).
3. It is complete. Also, in this particular case, even if the proposed heuristic is not admissible, given that all solutions are optimal, A* also assures optimality.

Question on the practice

1. We show it in the following figure



2. It would be:

$$P(Attack, Advantage, Opponent, Aggressive, Allies) =$$
$$P(Opponent)P(Advantage)P(Aggressive|Opponent)P(Allies|Opponent)P(Attack|Opponent, Advantage)$$

3. It would be:

$$P(Attack = yes|Aggressive = yes, Opponent = weak) = \frac{P(Attack = yes, Aggressive = yes, Opponent = weak)}{P(Aggressive = yes, Opponent = weak)} =$$

Let us call $\alpha = \frac{1}{P(Aggressive=yes,Opponent=weak)}$

$=\alpha \sum_{Allies=\{yes,no\}} \sum_{Advantage=\{yes,no\}} P(Attack = yes, Aggressive = yes, Opponent = weak, Allies, Advantage) =$

$\alpha \sum_{Allies} \sum_{Advantage} P(Opponent = weak)P(Advantage)P(Aggressive = yes|Opponent = weak)P(Allies|Opponent = weak)$
$P(Attack = yes|Opponent = weak, Advantage) =$

$\alpha P(Opponent = weak)P(Aggressive = yes|Opponent = weak) \sum_{Allies} P(Allies|Opponent = weak)$
$\sum_{Advantage} P(Advantage)P(Attack = yes|Opponent = weak, Advantage)$

Since $\sum_{Allies} P(Allies|Opponent = weak)$ is independent of the rest of the formula and equal to 1, we can remove it.

$= \alpha P(Opponent = weak)P(Aggressive = yes|Opponent = weak)$
$\sum_{Advantage} P(Advantage)P(Attack = yes|Opponent = weak, Advantage)$

4. We have to compute $P(Attack = yes|Aggressive = yes, Opponent = weak)$ and $P(Attack = no|Aggressive = yes, Opponent = weak)$ and see which one is bigger.

$P(Attack = yes|Aggressive = yes, Opponent = weak) =$
$\alpha P(Opponent = weak)P(Aggressive = yes|Opponent = weak) \sum_{Advantage} P(Advantage)P(Attack = yes|Opponent = weak, Advantage) =$
$\alpha 0.5 \cdot 0.4 \cdot (0.25 \cdot 1.0 + 0.75 \cdot 0.75) = \alpha 0.2 \cdot (0.25 + 0.56) = 0.16\alpha$

$P(Attack = no|Aggressive = yes, Opponent = weak) =$
$\alpha P(Opponent = weak)P(Aggressive = yes|Opponent = weak) \sum_{Advantage} P(Advantage)P(Attack = no|Opponent = weak, Advantage) =$
$\alpha 0.5 \cdot 0.4 \cdot (0.25 \cdot 0.0 + 0.75 \cdot 0.25) = \alpha 0.2 \cdot 0.19 = 0.38\alpha$

We can see that it is better not to attack.