Spécialité informatique
2^{e} année

Artificial Intelligence

Tutorials

Régis CLOUARD

Chapter: Uninformed searching

Exercise 1. The Maze problem

Your agent will find a path in a maze to reach a particular location. The maze is completely known. Its is represented by a grid.

1) Define the problem as a search problem.

2) Consider the following 6x6 empty maze where each letter corresponds to a possible location. Give the first levels of the search tree regarding the three uninformed search algorithms and a hand left right-hand rule:

(1) Breadth-First Search

(2) Depth-First Search

(3) Iterative Deepening Search

```
# # # # A #
# B C D E #
# F # G H #
# # # I J #
# K L M N #
# O # # # #
```

Exercise 2. The Jug problem

Given two full jugs with 3 and 5 liter capacities and an empty jug with 8 liters, find a sequence of pours leaving 4 liters in the two largest jugs.

1) Define the problem as a search problem. The description should be precise enough to be directly implemented.

2) Compute the maximum space complexity for the BFS algorithm.

3) Draw a solution as a search tree.

Exercise 3. Put 'OURS' in 'CAGE'

The goal is to produce the word CAGE from the word OURS using only words in a specified dictionary, and changing only one letter from two consecutive words.

The dictionary contains 21 words of 4 letters:

```
(baie brie bris cage came cime cire cure
 dues dure durs gage gaie mage mare mars
 murs ours pris pues puis)
```

1) Formalize the problem as a search problem.

2) Compute the maximum space complexity the BFS algorithm.

3) Suppose we are using an informed search strategy, propose an admissible heuristic.

Exercise 4. State representation

NASA are planning to send an autonomous planetary rover to explore the surface of Mars. The rover's task is to collect useful scientific data and keep itself operational for as long as possible.

The rover is at Site A and need to visit Site B and Site C. Its batteries currently hold 30 units of charge. To recharge the batteries to their full capacity (100 units), it needs to return to Base. The distance between the sites are shown in Figure 1.

It takes 10 units of charge to travel 1 km. So if the rover has 30 units of charge left and it decides to move from Site A to Site B, it will have 10 units of charge left when it arrives at B.

A step in the plan consists in moving the rover from one location to another. The actions of recharging the batteries and picking up data are completed automatically when the rover visits the appropriate location. The rover's goal is to collect data both from Site B and from Site C. Note that it does not need to return to base once the data is collected - it can, transmit the data back to Earth for anywhere on the surface.

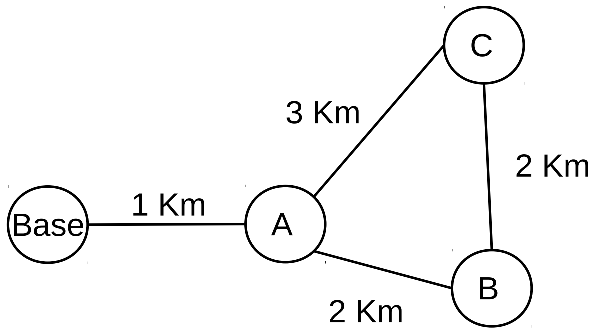

Figure 1: Distance map for the Rover problem.

1) Formulate this problem as a search problem.

2) Which of the uninformed algorithms can be applied to solve this problem?

3) Apply breadth-first to this problem, showing the order in which the nodes are visited in the tree. Find the plan that completes in the fewest number of steps.

4) Draw the search tree for this problem down to depth of 4.

5) Is your plan guaranteed to be optimal in terms of the traveled distance?

6) Which algorithm can be used to find the optimal distance?

Exercise 5. The Wolf, goat and cabbage problem

You are on the bank of a river with a boat, a cabbage, a goat, and a wolf. Your task is to get everything to the other side.

Restrictions:

- only you can handle the boat;
- when you're in the boat, there is only space for one more item;
- you can't leave the goat alone with the wolf, nor with the cabbage (or something will be eaten).

1) Model this problem as a search problem. The description should be precise enough to be directly implemented.

2) Give the higher cost of a depth-first searching.

3) Using the previous modeling, give a solution as a sequence of actions.

Exercise 6. Combinatorial search

Suppose you are searching for a girl’s name written using only the letters D, N and A. You have the letters ordered alphabetically (A, D, N) and you start writing down possibilities:

A, D, N, AA, AD, AN, DA, DD, DN, NA, ND, NN, …

1) How many strings of 4 letters can you form?

2) In the above possibilities, are you searching in a depth first or breadth first way?

3) What are the next three possible names you would write down?

4) How many possibilities will you write down before getting to the name ANNA?

5) Are you guaranteed to find all girls names with letters D, N and A in this manner?

Exercise 7. Golf-Tree Puzzle

Assume the given initial position for the Golf-Tee Puzzle:

```
        0
      1   1
    1   1   1
  1   1   1   1
1   1   1   1  1
```

Legal move is: jump over an adjacent tee and removing the jumped tee. For the above configuration one possible play would result in the following board layout (tee in (0, 2) jump into (0, 0) and remove tee (0, 1):

```
        1
      0   1
    0   1   1
  1   1   1   1
1   1   1   1   1
```

Goal state:

```
        1
      0   0
    0   0   0
  0   0   0   0
0   0   0   0   0
```

1) Model this problem as a search problem.

2) Give the python code.

3) How does the solver work?

Chapter: Informed Search

Exercise 1. Search problem

Consider the 3-puzzle problem which is a simpler version of the 8-puzzle where the board is 2×2 and there are three tiles numbered 1, 2, and 3, and one blank. The start and goal states are given in Figure 2.

1) Show how the path to the goal can be found using A* search with the heuristic being the number of misplaced tiles (Hamming distance).

Assume that there is no possibility to remember states that have been visited earlier. Label each visited node with a number indicating the order in which they are visited.

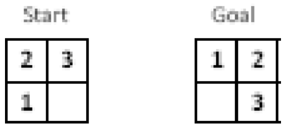

Figure 2: Start and goal states of the 3-puzzle.

Exercise 2. State Spaces and Search

A robot has to deliver identical packages to locations A, B, and C, in an office environment. Assume it starts off holding all three packages. The environment is represented as a grid of squares, some of which are free (so the robot can move into them) and some of which are occupied (by walls, doors, etc.). The robot can move into neighboring squares, and can pick up and drop packages if they are in the same square as the robot.

1) Formulate this problem as a search problem, specifying the state space, action space, goal test, and cost function.

2) Give a polynomial-time admissible heuristic for this domain, or argue that there isn't one.

3) Package B has to be delivered to the boss, and it's important that it be done quickly. But we should deliver the other packages promptly as well. How could we encode this in the problem?

4) Now consider the case where the robot doesn't start with the packages, but it has to pick up a package from location 1 to deliver to location A, a package from location 2 to deliver to B, and from 3 to deliver to C. What is an appropriate state space for this problem?

5) What is a good admissible heuristic for the previous question?

Exercise 3. Sokoban

Sokoban is a type of transport puzzle. The game is played on a board of squares, where each square is a floor or a wall. Some floor squares contain boxes, and some floor squares are marked as storage locations.

The player is confined to the board, and may move horizontally or vertically onto empty squares (never through walls or boxes). The player can also move into a box, which pushes it into the square beyond. Boxes may not be pushed into other boxes or walls, and they cannot be pulled. The puzzle is solved when all boxes are at storage locations.

If we assume that branching factor is 2.5 and the length of an average solution is about 200 moves, we would end up with a complexity in the order of 2.5^{200}.

1) Define the problem as a search problem.

2) Give an estimation of the worst-case complexity.

3) Give admissible heuristics and order them (dominance).

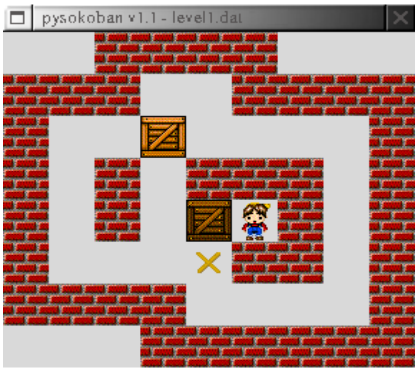

Figure 3: Example of a Sokoban puzzle.

Exercise 4. Rush Hour

Rush Hour is a sliding block puzzle. It is composed of a board and cars and trucks. A board is a 6x6 grid with grooves in the tiles to allow cars to slide, and an exit hole which only the red car can escape in the third row (marked with a red arrow in the figure). However, the cars and trucks obstruct the path which makes the puzzle harder. The game comes with 12 cars and 4 trucks, each colored differently. The cars take up 2 squares each, and the trucks take up 3.

Figure 4: Example of a Rush Hour puzzle.

Each configuration of the game shows where cars and trucks get placed on the board. Cars and trucks can only be moved within a straight line along the grid. They cannot be rotated. The goal of the game is to get the red car out by moving the other vehicles out of its way.

1) Define the problem as a search problem.

2) Give an estimation of the number of states.

3) Give admissible heuristics and order them (dominance).

Exercise 5. Algorithm A*

Consider the road map given Figure 5. The goal is to find the shortest path between A to I. The connection cost is indicated with the link in the map. The heuristic values are the following:

Node	A	B	C	D	E	F	G	H	I
h	10	4	5	10	10	3	3	4	0

1) Prove that heuristics h admissible?

2) Use the uniform cost algorithm to find a solution.

3) Use the greedy algorithm to find a solution with heuristic h.

4) Use the A* algorithm to find a solution with heuristic h.

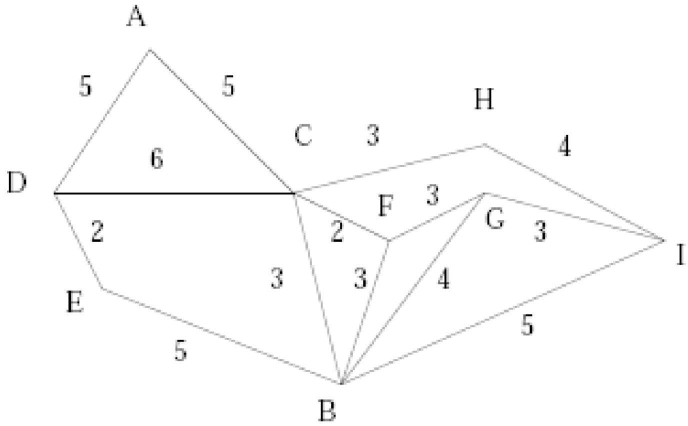

Figure 5: Distance map.

Exercise 6. Search algorithms

Consider the search space below Figure 6, where S is the start node and G1, G2 and G3 satisfy the goal test. Arcs are labeled with the actual cost of traversing them and the estimated cost (heuristic) to a goal state is reported inside the nodes.

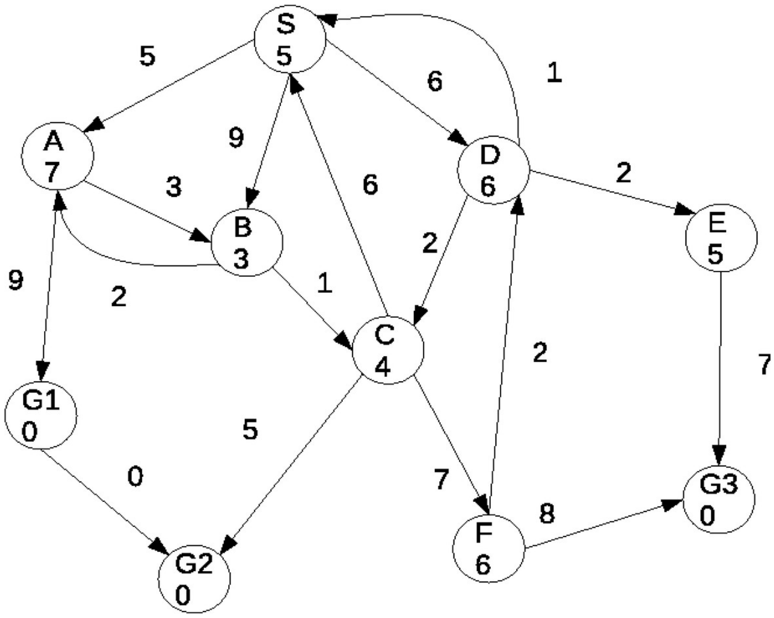

Figure 6: Search graph.

1) Apply the following search strategies to this problem, showing the agenda at each step in the search and the cost of the solution. When costs are equals, the nodes in the agenda should be visited in alphabetical order.

(1) Depth-first search

(2) Breadth-first search

(3) Greedy search

(4) A*

Chapter: Constraint Satisfaction Problem

Exercise 1. Setting Network

Consider the graph Figure 7 where each node is a variable and each arc is labeled with a number. Each variable takes values from the range [0, 9]. Each arc represents a constraint the two variables connected by the arc must satisfy. The constraint is that each variable must have the same value modulo the number on the arc.

For example, the arc connecting x, z with value 3 represents the constraint x mod 3 = z mod 3. This constraint can be satisfied by the following assignments (x = 2, z = 2) or (x = 5, z = 2) or (x = 7, z = 1).

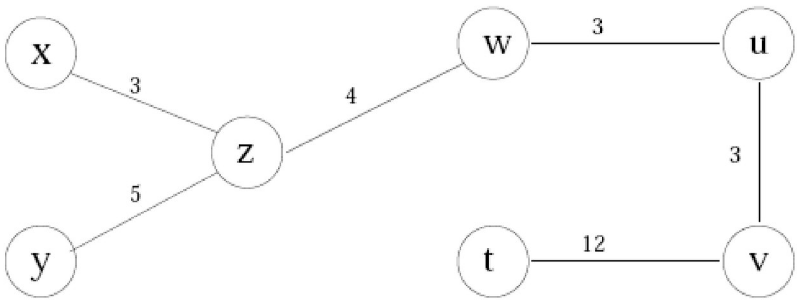

Figure 7: A constraint graph.

1) Model this problem as CSP.

2) Give variable values that would be inferred by forward checking algorithm after the first three assignments: x = 2, y = 0 and t = 0.

3) Give variable values that would be inferred by arc consistency algorithm after the first three assignments: x = 2, y = 0 and t = 0.

Exercise 2. Cryptarithmetic

Each letter stands for a distinct digit; the aim is to find a substitution of digits for letters such that the resulting sum is arithmetically correct, with the added restriction that no leading zeroes are allowed.

```
  TWO
+ TWO
 ----
 FOUR
```

1) Model the problem as CSP.

2) Which variable and which value would be used first? Which heuristics?

3) Give variable values that would be inferred by forward checking algorithm.

4) Use the arc consistency algorithm to prune the domain of the variables of the two arcs: O → R, and R → O.

Exercise 3. Sudoku satisfaction

Sudoku (meaning "*only one single number can fit*" in Japanese) is a puzzle in a 9x9 grid so that each column, each row, and each of the nine 3x3 boxes that compose the grid contains all the digits from 1 to 9 (e.g, Figure 8). The puzzle setter provides a partially completed grid, which typically has a unique solution.

1) Try to solve by hand the puzzle on the left of Figure 8.

2) Model this problem.

3) Which heuristics are available for choosing the variable.

		2	
2			4
		3	
	1		

1	4	2	3
2	3	1	4
4	2	3	1
3	1	4	2

Figure 8: A smaller version of the regular Sudoku puzzle.

Exercise 4. Takuzu satisfaction

Takuzu is a logic-based number placement puzzle. The objective is to fill a square grid with 1s and 0s, where there is an equal number of 1s and 0s in each row and column and no more than two of either number adjacent to each other. Additionally, there can be no identical rows or columns.

	1		0
		0	
	0		
1	1		0

0	1	1	0
1	0	0	1
0	0	1	1
1	1	0	0

Figure 9: Initial (left) and final (right) Takuzu puzzle.

1) Try to solve by hand the puzzle on the left of Figure 9.

2) Model this problem.

3) Forward checking: how to prune the domain of values?

4) Arc consistency: how to prune the domain of values?

5) Heuristics

Exercise 5. Logic puzzle (*Logigramme*)

Consider the problem of 3 couples man/woman at a party.

Each person is talking to another person of opposite sex, and that is not his husband or her wife.

(1) Marie is talking to Pascal.

(2) Georges is talking to Pascal's wife.

(3) Denise is talking to Marie's husband.

1) Model this problem as a CSP problem.

2) Apply the forward checking algorithm to find a solution.

Exercise 6. Zebra

The zebra puzzle is a well-known logic puzzle. It is often called Einstein's Riddle because it is said to have been invented by Albert Einstein as a toy; it is sometimes claimed that only 2 % of the population can solve it. However, there is no known evidence for Einstein's authorship; and the original puzzle cited below mentions brands of cigarette, such as Kools, that did not exist during Carroll's lifetime or Einstein's boyhood.

There are several versions of this puzzle. The version below is quoted from the first known publication in Life International magazine on December 17, 1962. The March 25, 1963 issue contained the solution given below and the names of several hundred solvers from around the world.

Consider the following logic puzzle: In five houses, each with a different color, live 5 persons of different nationalities, each of whom prefer a different brand of cigarette, a different drink, and a different pet.

1) Given the following facts, the question to answer is: “Where does the zebra live, and in which house do they drink water?”.

List of the facts:

- The Englishman lives in the red house.
- The Spaniard owns a dog.
- Coffee is drunk in the green house.
- The Ukrainian drinks tea.
- The green house is directly to the right of the ivory house.
- The Old-Gold smoker owns snails.
- Kools are being smoked in the yellow house.
- Milk is drunk in the middle house.
- The Norwegian lives in the first house on the left.
- The Chesterfield smoker lives next to the fox owner.
- Kools are smoked in the house next to the house where the horse is kept.
- The Lucky-Strike smoker drinks orange juice.
- The Japanese smokes Parliament.
- The Norwegian lives next to the blue house.

Exercise 7. Map coloring problem

The goal is to color a map such that no countries on the map that share a border are assigned the same color. The number of colors is limited--It has been demonstrated that 4 colors are sufficient-- For the purpose of assignment assume you have a map with 10 countries and you have three different colors: Green, Red and Blue.

1) Formulate the map coloring problem as a (graph) search by defining its initial state, operators and the goal condition.

2) What is the search space size of your formulation?

3) If the exact calculation of the search space size of your formulation becomes hard, give a reasonable upper bo

4) und estimate.

Exercise 8. Constraint-Satisfaction Problem

Johnny Seafood works at the aquarium in his town. The aquarium where Johnny works has recently got 5 new fish species that need to be allocated in the three available tanks. There are some constraints concerning what species can be together in a tank and also some specific requirements.

(1) Sharks and dolphins can’t be together in the same tank.

(2) Sharks and remoras (*poissons pilotes*) must be in the same tank.

(3) Dolphins can’t be in tank 3. It is not possible for them to perform their show in this tank.

(4) There are too many dolphins, so they cannot share the tank with the killer whale (*orque*).

(5) The tank closest to the food deposit is tank 2. Johnny wants to allocate the whale there, so he doesn’t have to carry its food too far.

(6) Remoras are so small that the whale could eat them. Thus, they can’t be in the same tank as the whale.

(7) Lobsters can be anywhere.

1) Define the problem as a CSP. Draw the constraint graph of the CSP.

2) Use the Forward Checking algorithm to solve the problem. Draw the search tree and the domain of the variables at each node.

Chapter: Adversarial Search

Exercise 1. Peg Solitaire

Peg solitaire is a board game. A valid move is to jump a peg horizontally or vertically over an adjacent peg into a hole two positions away and then to remove the jumped peg.

Consider a game similar to peg solitaire except that the game board is different (see Figure 10) and there are 2 players: Min tries to minimize the number of remaining pegs at the end and Max tries to maximize the number. The score for Max is equal to the number of remaining pegs.

1) Specify the game classification.

2) Formulate the problem as adversarial search.

3) Max starts: use the Minimax algorithm to determine the lowest score Max can expect to win.

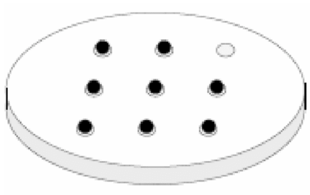

Figure 10: Plateau de jeu.

Exercise 2. Game Search Algorithms

Consider the game search tree in the Figure 11.

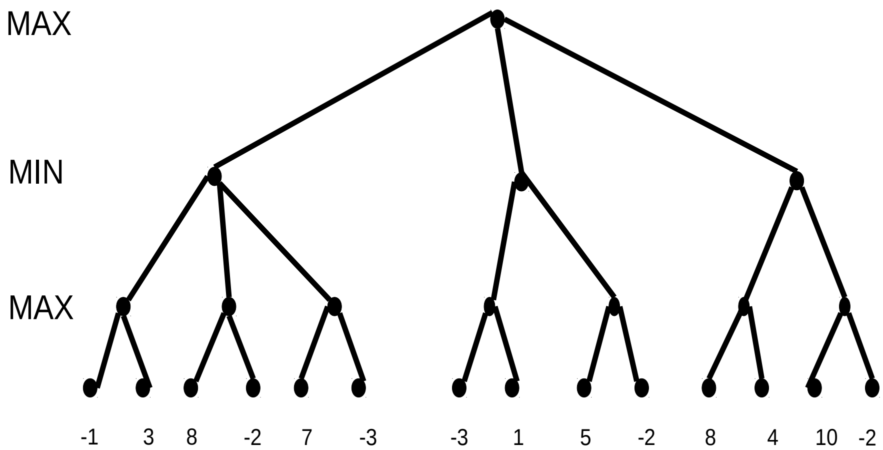

Figure 11: Search game tree.

Assume the first player is the max player and the values at leaves of the tree reflect his/her utility. The opponent wants the same utility to be minimized.

1) Compute the minimax values for each node in the tree? What move should the first player choose? What is the solution path the rational players would play?

2) Assume we use alpha-beta algorithm to explore the game tree, and we do this in the left-to-right order and determine the players strategies. List all nodes that are cut off from the tree and are never examined by the alpha beta procedure.

3) Assume we explore the tree in the right-to-left order. What nodes would not need to be examined by the alpha-beta algorithm and pruned away?

4) How to order children to maximize alpha-beta pruning?

Exercise 3. Reversi

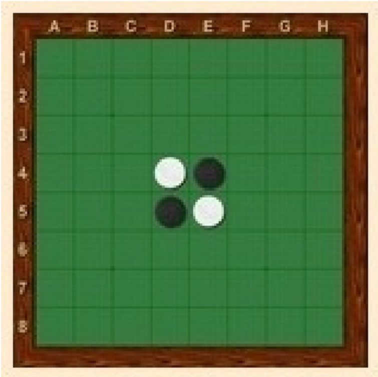

Figure 12: Initial state of the reversi board.

1) Give the game classification.

2) Formulate the problem as an adversarial problem.

3) The game tree is too large to find the exact solution. Utility function becomes evaluation function. How do you propose to represent the evaluation function?

Exercise 4. Pac-Man

1) Give the game classification.

2) Formulate the problem as an adversarial problem.

3) The ghosts act randomly. What is the best algorithm that can be used to solve the problem?

4) The game tree is too large to find the exact solution. Utility function becomes evaluation function. How do you propose to represent the evaluation function without knowing how the score is computed?

Exercise 5. Variations on games

1) Suppose that your opponent plays randomly. They no longer search for the minimum ply. What does the game tree look like?

2) Now suppose a game with skill and chance such as the monopoly (or backgammon). What does the game tree look like?

3) Now suppose a non-zero sum game like monopoly or backgammon. What does the game tree look like?

4) Finally, suppose a card game like French Tarot. What does the game tree look like?

Exercise 6. Alpha Beta algorithm

1) What letter max should play if the opponent is rational (use alpha-beta algorithm)?

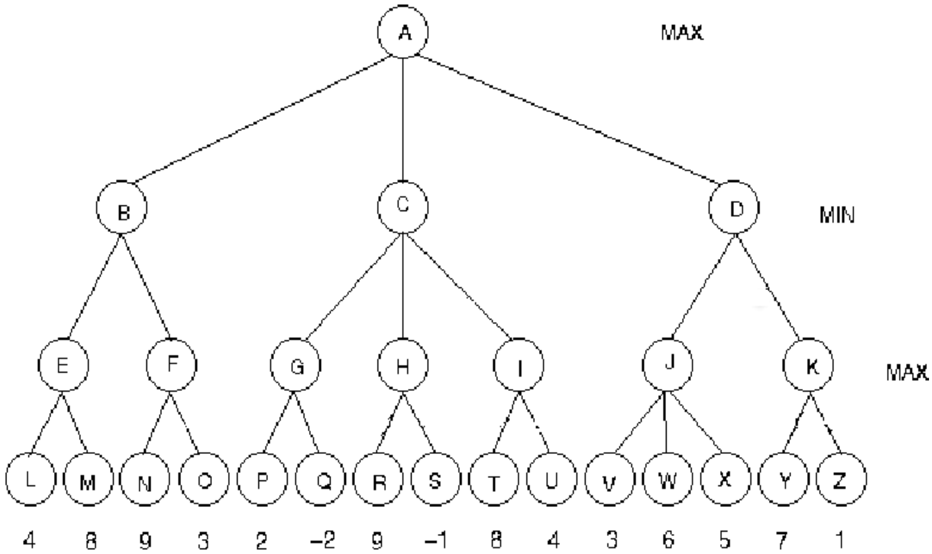

Figure 13: Game tree.

2) What letter max should play if the opponent acts randomly (uniform law)?

Exercise 7. Backgammon: game of chance

The goal of this game is to move all one's pieces off the board. White moves clockwise toward 25, and black moves counter clockwise toward 0. A piece can move to any position unless there are multiple opponent pieces there; if there is one opponent, it is captured and must start over. In the position shown, White has rolled 6-5 and must choose among four legal moves: (5-10, 5-11), (6-11, 19-24), (5-10, 10-16), and (5-11, 11-16).

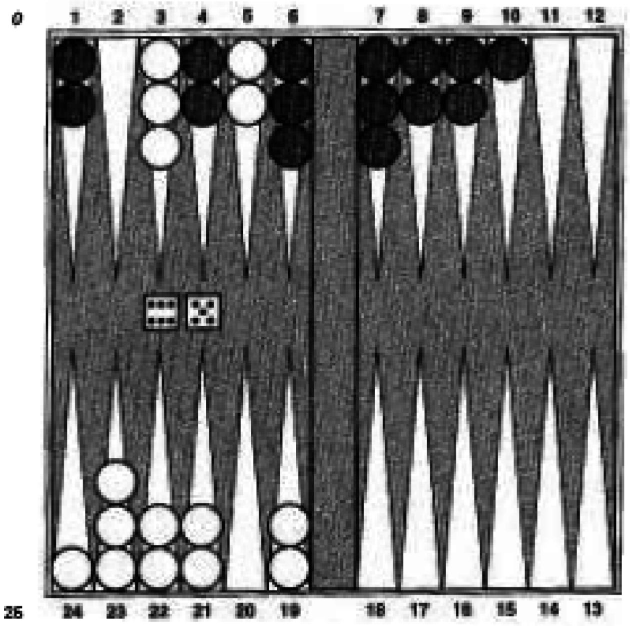

Figure 14: Backgammon.

1) How to model the game tree?

Exercise 8. Nim game

To play Nim, a number of tokens are placed in a pile between two opponents; at each move, the player must divide a pile into two nonempty piles of different sizes. Thus, a pile of six tokens may be divided into two piles of [5,1] or [4,2] but not [3,3]. The first player who can no longer move loses.

1) Write the Nim game search tree if we consider only 7 tokens.

2) Use the best algorithm to find the best branch among the tree.

Exercise 9. Minimax for several adversaries

1) Give the Minimax values for the game tree given in the following Figure 15.

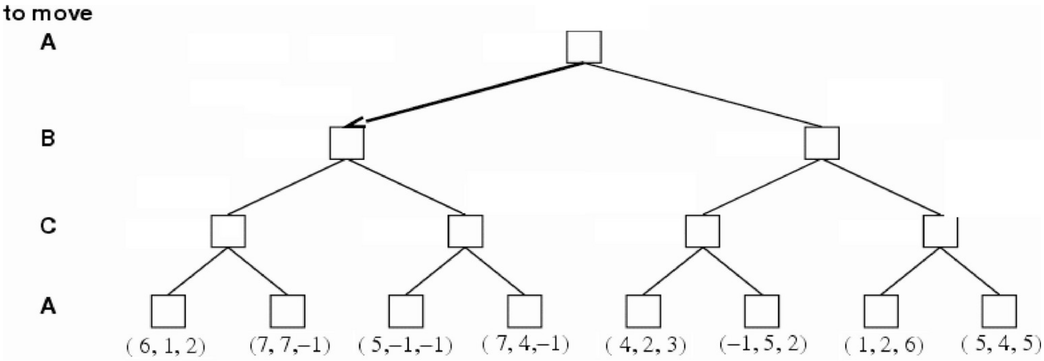

Figure 15: Game tree.

Chapter: Markov Decision Process

Exercise 1. Planification par les processus de décision de Markov

Soit le processus de décision markovien suivant, dont les transitions sont étiquetées par les noms des actions avec les probabilités de transitions et les états sont étiquetés par les récompenses correspondantes.

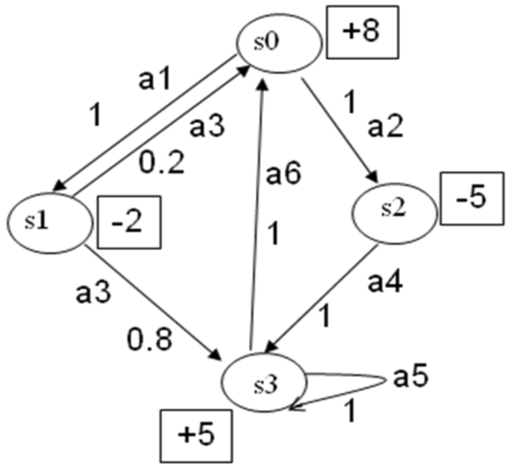

Figure 16: Exemple d'un processus de décision Markovien.

1. Indiquez les valeurs de la fonction d'utilité pour chaque état à la fin de chacune des trois premières itérations de l'algorithme value-iteration, en supposant qu'on utilise un facteur d'escompte (*discount factor*) de 0.5 et en partant initialement (*itération 0*) avec des valeurs toutes égales à 0.
2. Donnez la politique correspondant aux valeurs à chaque itération.

Exercise 2. Processus de décision de Markov

Consider the following Markov Decision Process, where transitions are labeled by the action name with the related transitions probabilities and the states are labeled by the immediate reward. It models the possible actions of a character being chased by others. Uncertainty come from other behavior.

In the quiet state, nothing happens –no enemy on the horizon. In this state, The character receives a reward of +2.

If he waits without doing anything, he is in danger of being localized and trapped with a probability of 0.2. If he is trapped, he receives a penalty of 5 points.

In the quiet state, if it moves, it has a 50% chance to be trapped and be surrounded by the enemies. Once surrounded or trapped, the character can attempt to run away or defend itself.

1. Indicate the values of the utility function for each state at the end of each of the first two iterations of the value iteration algorithm, assuming a discount factor of 0.9. and starting with initial values all equal to 0.
2. Give the best policy of action resulting from iteration 2.

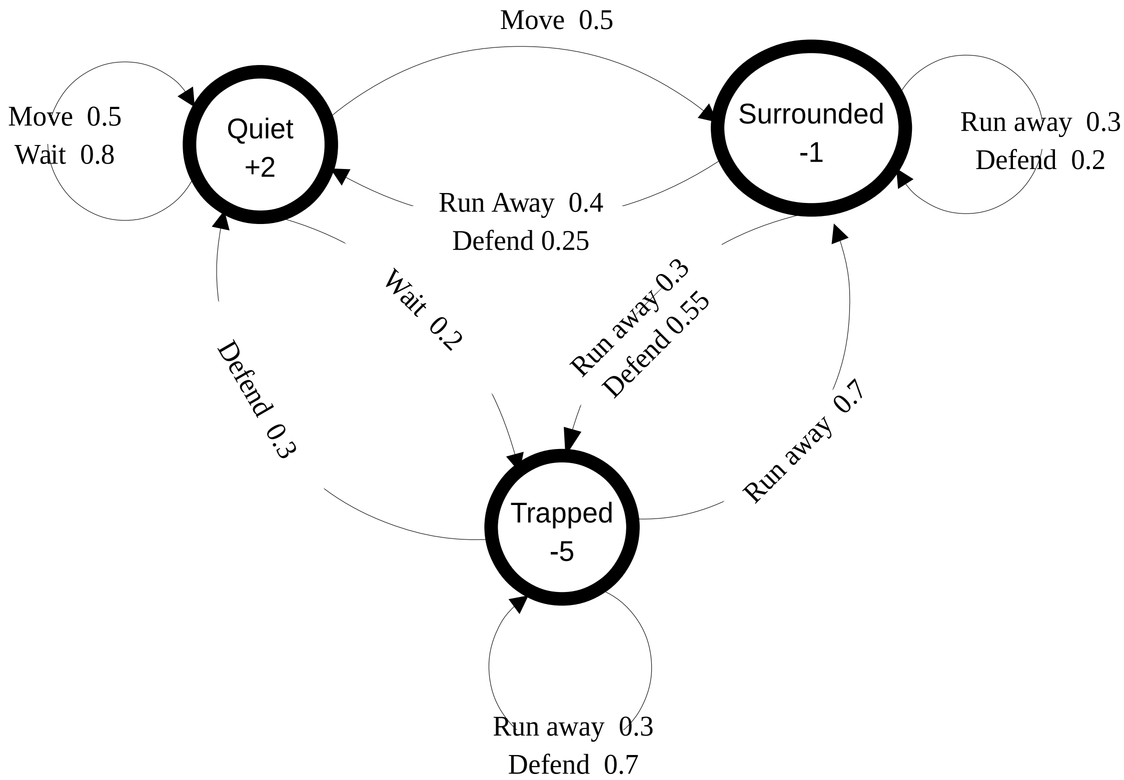

Figure 17. Markov Decision Process for the game.

Exercise 3. Automobile replacement problem

Suppose that we use a machine of a certain age. Should we keep it on or should we trade it in; in the latter case, how old machine should we buy.

To be more concrete, suppose that we review a car every month and that the decision is made either to keep the present car or to trade in the car for another car of a certain age. The age of a car is measured in months.

In order to keep the state space finite, we assume that there is a large stage N, i.e. a car of age N will always be reset by another car. Furthermore, we assume that a car of age i has a probability p_i of a breakdown in which case it ends up in state N.

Suppose that we have the following costs and rewards:

b_i : cost of buying a car of age i;

t_i : trade-in value of a car of age i;

c_i : expected maintenance cost in the next period for a car of age i.

Give the MDP data for this automobile problem.

Exercise 4. Learning by reinforcement

Consider a MDP with the set of states S = {s_0, s_1, s_2, s_3} where s_2 is terminal, the set of actions A={a_1, a_2, a_3} and the discount factor γ = 0.5.

It is assumed that all actions are possible from each state.

Now suppose that the following trials were generated by an agent doing training, following some exploration policy. Note that the number to the right of the state index is the immediate reward for that state.

$$(s_0)_1 \xrightarrow{a_1} (s_1)_1 \xrightarrow{a_2} (s_1)_1 \xrightarrow{a_2} (s_2)_{10}$$

$$(s_0)_1 \xrightarrow{a_3} (s_0)_1 \xrightarrow{a_3} (s_1)_1 \xrightarrow{a_1} (s_1)_1 \xrightarrow{a_1} (s_0)_1 \xrightarrow{a_3} (s_2)_{10}$$

$$(s_0)_1 \xrightarrow{a_2} (s_3)_2 \xrightarrow{a_1} (s_2)_{10}$$

$$(s_0)_1 \xrightarrow{a_1} (s_0)_1 \xrightarrow{a_1} (s_3)_2 \xrightarrow{a_1} (s_1)_1 \xrightarrow{a_2} (s_2)_{10}$$

1. Give the list of updates of the Q(action-value) function. Assume an initialization of Q(s, a) to 0 and use a learning rate $\alpha = 0.1$.
2. What would be the policy learned at the end? Give the action chosen by this policy for each state.

Chapter: Logic and inferences

Exercise 1. Propositional logic

1) Translate the following sentences into predicate logic sentences.

(1) “If it is hot and humid, then it is raining.”

(2) “If it is humid, then it is hot.”

(3) “It is humid.”

2) Using the truth-table approach, answer the question α: “Is it raining?”, i.e., is KB a model for a; $KB \models \alpha$?

3) Same question using the inference approach Forward chaining

4) Same question using the inference approach Backward chaining.

Exercise 2. Treasure Island

In the back of an old cupboard (*placard*) you discover a note signed by a pirate famous for his love of logical puzzles. In the note he wrote that he had hidden treasure somewhere on the property. He listed five true statements and challenged the reader to use them to figure out the location of the treasure.

(1) If the house is next to a lake, then the treasure is not in the kitchen.

(2) If the tree in the front yard is an oak (*chêne*) then the treasure is in the garage.

(3) If the treasure is not in the kitchen then tree in the front yard is not an elm.

(4) This house is next to a lake.

(5) The tree in the front yard is an elm (*orme*) or the treasure is buried under the flagpole.

1) Find where the treasure is hidden using the propositional logic and forward chaining inferences.

Exercise 3. First-Order Logic: Natural Deduction for Colonel Chabert

“The law says that it is a crime for a Frenchman to sell weapons to a hostile nation. The country Albion, an enemy of France, has some missiles, and all of its missiles were sold to it by Colonel Chabert, who is a Frenchman.”

1) Prove Colonel Chabert is a criminal using the backward chaining. This means to represent the sentences as Horn form.

Exercise 4. Who killed Victor?

Victor has been murdered, and Arthur, Bertrand, and Charles are the only suspects (meaning exactly one of them is the murderer). Arthur says that Bertrand knew the victim, and that Charles hated the victim. Bertrand says that he was out-of-town the day of the murder, and besides he didn’t even know the guy. Charles says that he saw Bertrand in town before the murder.

Besides, several witnesses said that: Victor was a friend of Bertrand, that Charles hated Victor and that Charles has been seen with Bertrand as well as with Victor just before the murder. Assuming that at least one suspect is guilty, that a suspect is telling the truth only if he is innocent, and that all other witnesses are telling the truth, we want to determine who is guilty.

1) Represent the above information in FOL (in Horn clause).

2) Prove that Bertrand is guilty using backward chaining.

Exercise 5. First-order logic: Proofs

Recall that $\alpha \models \beta$ iff β is true in all models in which α is true. A formula is *valid* if all interpretations make the formula true. For classical logics, it is also possible to reexpress the question of the validity as that: α is valid if and only if $\neg\alpha$ is unsatisfiable.

1) Prove the following enumerations:

(1) α is valid iff true $\models \alpha$.

(2) For all α, false $\models \alpha$.

(3) $\alpha \models \beta$ iff statement $(\alpha \Rightarrow \beta)$ is valid.

(4) $\alpha \models \beta$ iff statement $(\alpha \wedge \neg\beta)$ is unsatisfiable.

Exercise 6. Translation

Translate the following sentences into first-order logic:

(1) All dogs are mammals.

(2) Fido is a dog.

(3) Fido's mother is a mammal.

(4) All mammals have a mother who is a mammal.

(5) Fido is a mammal.

(6) All mammals produce milk.

(7) There exists a dog which doesn't produce milk.

Transform the sentence into Conjunctive Normal Form (even Horn Normal Form).

Exercise 7. Prove that Curiosity Killed the Cat

1) Translate the followings sentence into first-order logic and then into CNF:

"Everyone who loves all animals is loved by someone." "Anyone who kills an animal is loved by no one." "Jack loves all animals." "Either Jack or Curiosity killed the cat, who is named Tuna." "Tuna is a cat and cats are animals." "Did Curiosity kill the cat?"

2) Prove that curiosity kills the cat by using resolution.