



RAMA UNIVERSITY

www.ramauniversity.ac.in

FACULTY OF ENGINEERING

ARTIFICIAL INTELLIGENCE
LECTURE-15

Mr. Dharendra

Assistant Professor

Computer Science & Engineering

OUTLINE

- ❖ **Adversarial Search**
- ❖ **Types of Games in AI**
- ❖ **Zero-Sum Game**
- ❖ **Zero-sum game: Embedded thinking**
- ❖ **Formalization of the problem**
- ❖ **Game tree**
- ❖ **MCQ**
- ❖ **References**



Adversarial Search

- Adversarial search is a search, where we examine the problem which arises when we try to plan ahead of the world and other agents are planning against us.
 - In previous topics, we have studied the search strategies which are only associated with a single agent that aims to find the solution which often expressed in the form of a sequence of actions.
 - But, there might be some situations where more than one agent is searching for the solution in the same search space, and this situation usually occurs in game playing.
 - The environment with more than one agent is termed as multi-agent environment, in which each agent is an opponent of other agent and playing against each other. Each agent needs to consider the action of other agent and effect of that action on their performance.
 - So, Searches in which two or more players with conflicting goals are trying to explore the same search space for the solution, are called adversarial searches, often known as Games.
 - Games are modeled as a Search problem and heuristic evaluation function, and these are the two main factors which help to model and solve games in AI.

Types of Games in AI

Perfect information: A game with the perfect information is that in which agents can look into the complete board. Agents have all the information about the game, and they can see each other moves also. Examples are Chess, Checkers, Go, etc.

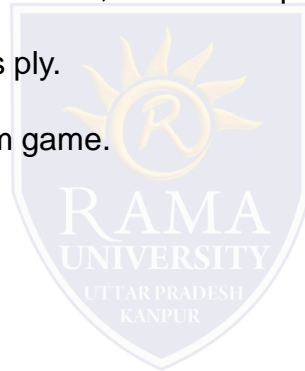
Imperfect information: If in a game agents do not have all information about the game and not aware with what's going on, such type of games are called the game with imperfect information, such as tic-tac-toe, Battleship, blind, Bridge, etc.

Deterministic games: Deterministic games are those games which follow a strict pattern and set of rules for the games, and there is no randomness associated with them. Examples are chess, Checkers, Go, tic-tac-toe, etc.

Non-deterministic games: Non-deterministic are those games which have various unpredictable events and has a factor of chance or luck. This factor of chance or luck is introduced by either dice or cards. These are random, and each action response is not fixed. Such games are also called as stochastic games. Example: Backgammon, Monopoly, Poker, etc.

Zero-Sum Game

- Zero-sum games are adversarial search which involves pure competition.
- In Zero-sum game each agent's gain or loss of utility is exactly balanced by the losses or gains of utility of another agent.
- One player of the game try to maximize one single value, while other player tries to minimize it.
- Each move by one player in the game is called as ply.
- Chess and tic-tac-toe are examples of a Zero-sum game.

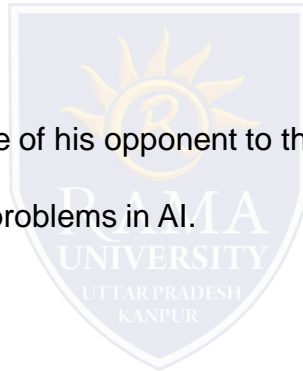


Zero-sum game: Embedded thinking

The Zero-sum game involved embedded thinking in which one agent or player is trying to figure out:

- What to do.
- How to decide the move
- Needs to think about his opponent as well
- The opponent also thinks what to do

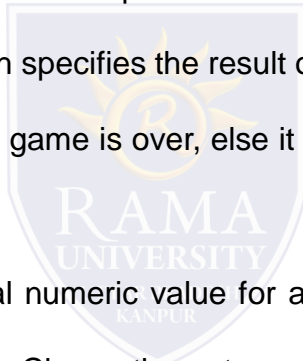
Each of the players is trying to find out the response of his opponent to their actions. This requires embedded thinking or backward reasoning to solve the game problems in AI.



Formalization of the problem

•A game can be defined as a type of search in AI which can be formalized of the following elements:

- Initial state: It specifies how the game is set up at the start.
- Player(s): It specifies which player has moved in the state space.
- Action(s): It returns the set of legal moves in state space.
- Result(s, a): It is the transition model, which specifies the result of moves in the state space.
- Terminal-Test(s): Terminal test is true if the game is over, else it is false at any case. The state where the game ends is called terminal states.
- Utility(s, p): A utility function gives the final numeric value for a game that ends in terminal states s for player p. It is also called payoff function. For Chess, the outcomes are a win, loss, or draw and its payoff values are +1, 0, $\frac{1}{2}$. And for tic-tac-toe, utility values are +1, -1, and 0.



Game tree

•A game tree is a tree where nodes of the tree are the game states and Edges of the tree are the moves by players. Game tree involves initial state, actions function, and result Function.

➤Example: Tic-Tac-Toe game tree:

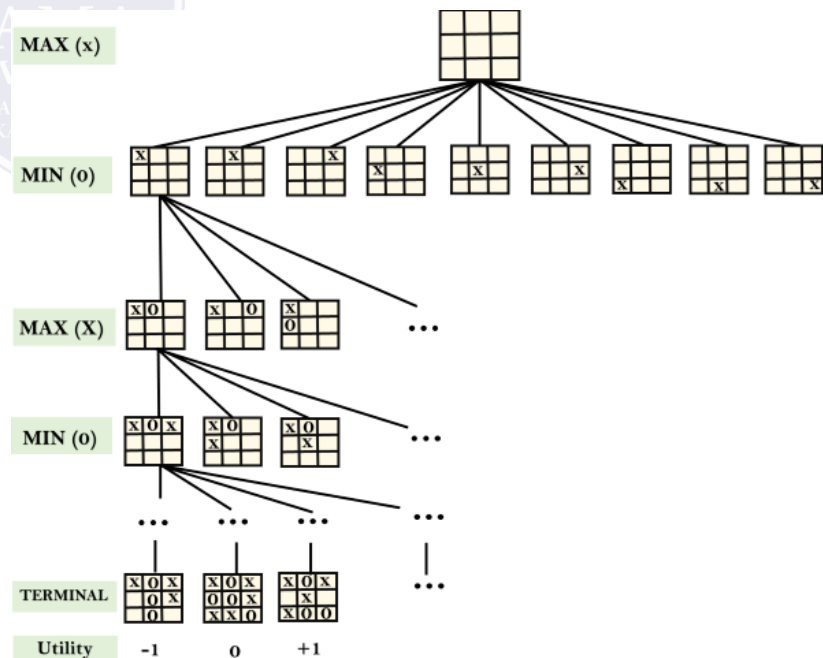
➤The following figure is showing part of the game-tree for tic-tac-toe game. Following are some key points of the game:

➤There are two players MAX and MIN.

➤Players have an alternate turn and start with MAX.

➤MAX maximizes the result of the game tree

➤MIN minimizes the result.



1. What is a top-down parser?

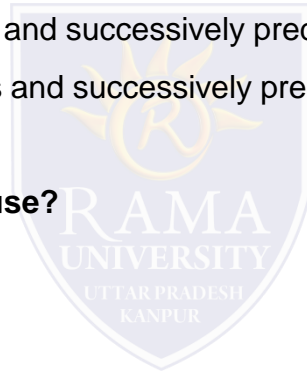
- a) Begins by hypothesizing a sentence (the symbol S) and successively predicting lower level constituents until individual preterminal symbols are written
- b) Begins by hypothesizing a sentence (the symbol S) and successively predicting upper level constituents until individual preterminal symbols are written
- c) Begins by hypothesizing lower level constituents and successively predicting a sentence (the symbol S)
- d) Begins by hypothesizing upper level constituents and successively predicting a sentence (the symbol S)

2. Among the following which is not a horn clause?

- a) p
- b) $\emptyset p \vee q$
- c) $p \rightarrow q$
- d) $p \rightarrow \emptyset q$

3. The action 'STACK(A, B)' of a robot arm specify to _____

- a) Place block B on Block A
- b) Place blocks A, B on the table in that order
- c) Place blocks B, A on the table in that order
- d) Place block A on block B

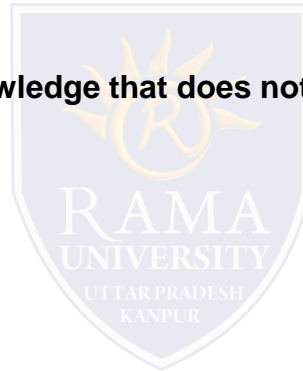


4. Different learning methods does not include?

- a) Memorization
- b) Analogy
- c) Deduction
- d) Introduction

5. In language understanding, the levels of knowledge that does not include?

- a) Phonological
- b) Syntactic
- c) Empirical
- d) Logical



References

- <https://www.javatpoint.com/digital-image-processing-tutorial>
- <https://www.tutorialpoint.com/>
- Stuart Russell, Peter Norvig, "Artificial Intelligence - A Modern Approach", Pearson Education.
- Elaine Rich and Kevin Knight, "Artificial Intelligence", McGraw-Hill.
- E Charniak and D McDermott, "Introduction to Artificial Intelligence", Pearson Education.
- Dan W. Patterson, "Artificial Intelligence and Expert Systems", Prentice Hall of India.

