Experiment No 1

Aim : Case Study On AI applications Published in IEEE/ACM/Springer or any prominent journal

Theory :

<u>Artificial Intelligence in Human Computing Games</u>

1. Introduction

    The introduction sets the stage by highlighting the burgeoning interest in character-based AI and its diverse applications beyond entertainment. It underscores the transformative potential of AI in military and medical training, educational games, and those advocating for social causes. The main aim of the research is emphasized: to develop AI techniques that profoundly impact the gaming industry

2. AI in Games:

    2.1. Traditional Approaches:

    This subsection briefly outlines traditional approaches to AI in games, such as rule-based systems and decision trees.

    2.2. Minimax Algorithm with Alpha-Beta Pruning:

    The Minimax algorithm is a decision-making algorithm widely used in two-player turn-based games. It aims to minimize the possible loss for a worst-case scenario. To improve its efficiency, Alpha-Beta pruning is employed to reduce the number of nodes evaluated in the minimax tree.

    2.2.1. Minimax Algorithm:

    In the context of gaming, the Minimax algorithm involves a recursive search through the game tree, where each node represents a possible game state. It assigns a value to each node based on the outcome of the game from that state.

    The algorithm alternates between maximizing and minimizing players until a terminal state is reached.

    2.2.2. Alpha-Beta Pruning:

    Alpha-Beta pruning is a technique used to cut off branches of the search tree that will not affect the final decision. By maintaining two values, alpha and beta, representing the minimum score the maximizing player is assured of and the maximum score the minimizing player is assured of, respectively, the algorithm can disregard subtrees that are known to be irrelevant.

    2.2.3. Application in Computing Games:

    Consider a simple example of applying Minimax with Algorithm to Tic-Tac-Toe. The game tree represents all possible moves and counter-moves until a terminal state is reached. The algorithm assigns scores to each leaf node, and as it traverses back up the tree, it selects the move that maximizes or minimizes the score, depending on the player.

    This algorithm allows an AI agent to make optimal moves in Tic-Tac-Toe, considering all possible outcomes and efficiently pruning irrelevant branches in the search space.

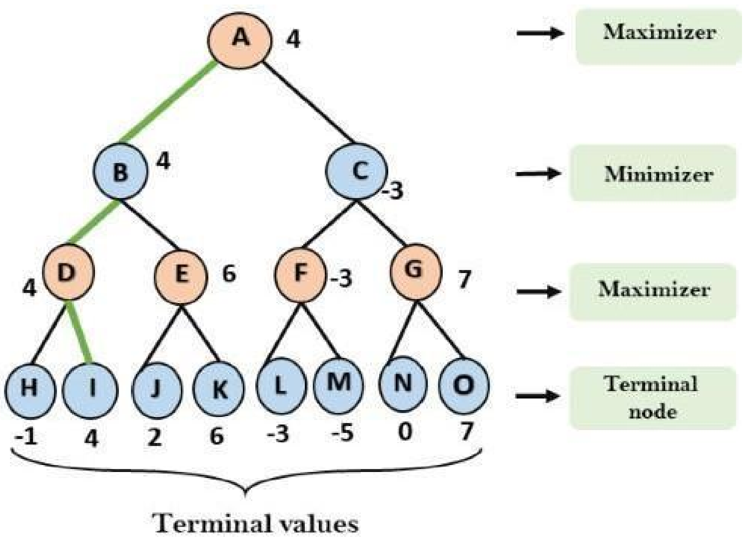
3. Challenges in AI:
   How Minimax addresses decision-making challenges in game AI and how Alpha-Beta pruning improves adaptability in the context of AI:
   3.1. Learning and Memory: Minimax in Game AI

Minimax is a decision-making algorithm commonly used in game AI to handle complex decision spaces, especially in games with two players and perfect information (meaning that the entire state of the game is visible to both players). It's often applied in games like chess, tic-tac-toe, and checkers.

Major Points:

- Tree-based Decision Model: Minimax creates a game tree that represents all possible moves and counter-moves by both players. This tree allows the AI to explore different decision paths and evaluate the potential outcomes.
- Maximization and Minimization: The algorithm alternates between maximizing and minimizing players. The maximizing player aims to achieve the highest possible score, while the minimizing player aims to minimize that score.
- Backtracking and Decision Evaluation: As the algorithm explores the tree, it backtracks and assigns scores to different game states. The AI evaluates the desirability of a move based on the potential outcomes and selects the move with the highest score for the maximizing player and the lowest score for the minimizing player.

Example : -The algorithm generates the entire game-tree and apply the utility function to get the utility values for the terminal states. In the below tree diagram



Properties of Mini-Max algorithm:

- **Complete-** Min-Max algorithm is Complete. It will definitely find a solution (if exist), in the finite search tree.
- **Optimal-** Min-Max algorithm is optimal if both opponents are playing optimally.

- o **Time complexity-** As it performs DFS for the game-tree, so the time complexity of Min-Max algorithm is **$O(b^m)$**, where b is branching factor of the game-tree, and m is the maximum depth of the tree.
- o **Space Complexity-** Space complexity of Mini-max algorithm is also similar to DFS which is **O(bm)**.

Limitation of the minimax Algorithm:

The main drawback of the minimax algorithm is that it gets really slow for complex games such as Chess, go, etc. This type of games has a huge branching factor, and the player has lots of choices to decide. This limitation of the minimax algorithm can be improved from **alpha-beta pruning**

Addressing Challenges:

- Handling Complexity: Minimax effectively manages complex decision spaces by systematically exploring the possibilities in a game tree. It ensures a methodical approach to decision-making, even in situations with numerous potential moves.
- Strategic Decision-Making: By evaluating potential outcomes and selecting the move with the highest score, Minimax helps the AI make strategic decisions, especially in games where long-term planning is crucial.

3.2. Adaptability: Alpha-Beta Pruning

Alpha-Beta pruning is an optimization technique used in conjunction with the Minimax algorithm to enhance adaptability and efficiency in decision-making.

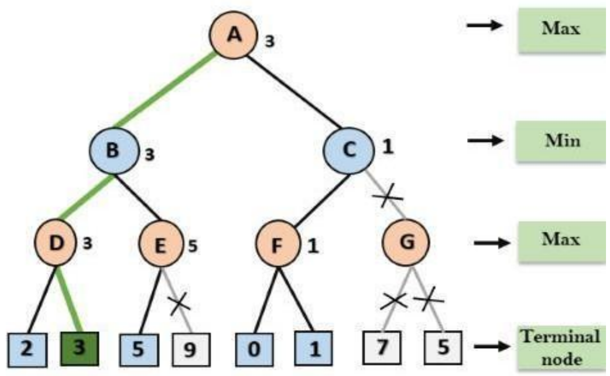
Major Points:

- Pruning Unnecessary Branches: Alpha-Beta pruning efficiently reduces the search space by eliminating branches in the game tree that are guaranteed to be irrelevant to the final decision. It does this by maintaining two values, alpha and beta, which represent the minimum score the maximizing player is assured of and the maximum score the minimizing player is assured of, respectively.
- Early Stopping: As the algorithm progresses, it stops exploring branches that cannot affect the final decision, saving computational resources and time.

Improving Adaptability:

- Faster Decision-Making: Alpha-Beta pruning significantly accelerates decision- making by avoiding unnecessary exploration of unpromising paths. This is particularly beneficial in games with large decision spaces, making the AI more adaptable in real-time scenarios.
- Resource Efficiency:The adaptability of the AI is improved by conserving computational resources. Alpha-Beta pruning allows the AI to focus on the most promising moves, leading to quicker and more resource-efficient decision-making.

Example: Let's take an example of two-player search tree to understand the working of Alpha-beta pruning



Move Ordering in Alpha-Beta pruning:

The effectiveness of alpha-beta pruning is highly dependent on the order in which each node is examined. Move order is an important aspect of alpha-beta pruning.

It can be of two types:

- **Worst ordering:** In some cases, alpha-beta pruning algorithm does not prune any of the leaves of the tree, and works exactly as minimax algorithm. In this case, it also consumes more time because of alpha-beta factors, such a move of pruning is called worst ordering. In this case, the best move occurs on the right side of the tree. The time complexity for such an order is $O(b^m)$.
- **Ideal ordering:** The ideal ordering for alpha-beta pruning occurs when lots of pruning happens in the tree, and best moves occur at the left side of the tree. We apply DFS hence it first search left of the tree and go deep twice as minimax algorithm in the same amount of time. Complexity in ideal ordering is $O(b^{m/2})$.

In summary, Minimax with Alpha-Beta pruning addresses challenges in AI by providing a systematic approach to decision-making in complex game scenarios and enhancing adaptability through efficient search space reduction.

4. Behavior Adaptation for Believable Characters:This section delves into the intricacies of creating characters with distinct personalities in interactive games. The action transformation system is explored through a vivid example involving a game of tag. Emotion-based personality constraints and reasoning modules are detailed, showcasing how they determine behavior revisions in response to personality contract violations.

5. Case-Based Planning for Strategy Games:

   The paper introduces a novel approach that extracts behavioral knowledge from expert demonstrations to address the vast search spaces in computer games, particularly in strategy games. A detailed explanation is provided on how the case-based planning engine retrieves suitable behaviors from expert demonstrations and adapts them to the current game state.

6. Drama Management in Interactive Stories:
   This section focuses on drama management, a crucial aspect of interactive storytelling. The goal is to give players a significant impact during interaction, deviating from pre-written scripts. The paper discusses the three-module approach involving a game engine, player modeling module, and drama management module. The player modeling module's construction of player models based on reactions and the drama management module's planning actions for a more appealing story direction are explored.

7. Proposed Approach and Future Work:
   The proposed approach, comprising a game engine, player modeling module, and drama management module, is elucidated in detail. The player modeling module's construction of player models based on reactions and the drama management module's planning
   actions for a more appealing story direction are explored. Future work, including extensive player evaluation and the expansion of player modeling modules, is outlined.

8. Conclusion:
   This section synthesizes the challenges and AI techniques discussed throughout the paper. It emphasizes the experiments showing the positive impact of Drama Management techniques on player experience. The belief in AI's significant impact on entertainment, education, and training is reiterated, emphasizing the need for continuous innovation and incorporation of AI in games for truly adaptive and immersive experiences.

9. Acknowledgment:
   A brief acknowledgment expresses gratitude to collaborators involved in the projects mentioned in the paper. Special thanks are extended for assistance in drama management work, real-time strategy games, and behavior modification work.

In conclusion, this comprehensive exploration of AI in human-computer interaction within gaming provides a detailed understanding of its challenges, innovative approaches, and potential future developments. The four-page document encapsulates the complexity and significance of AI in shaping the future of gaming experiences.