Artificial Intelligence

Lab no 6

Alpha-Beta Puring vs Min-Max Algorithm:

Alpha-beta pruning is an improvement over the basic Minimax algorithm used in game theory and decision making. Minimax exhaustively explores all possible game states up to a certain depth, evaluating each state's utility. However, it can be inefficient as it explores branches that won't affect the final decision.

Alpha-beta pruning enhances Minimax by cutting off branches that are determined to be irrelevant based on the values of previously evaluated nodes. This pruning technique significantly reduces the number of nodes evaluated, thereby reducing computational overhead. By eliminating unnecessary branches, Alpha-beta pruning allows for a more focused search, resulting in faster and more efficient decision-making processes.

Overall, Alpha-beta pruning retains the accuracy of Minimax while drastically reducing the computational resources required, making it a preferred choice for scenarios where computational efficiency is crucial, such as in games with large state spaces.

Example:

In a Tic Tac Toe scenario, let's say we have the following board:

X		O
X	O	

If we're considering placing an X in the bottom-right corner, Minimax would continue exploring all possible moves from there, even though it's clear that placing an X there would be blocked by an O in the middle column, thus having no impact on final outcome.

In contrast, Alpha-beta pruning would recognize this scenario and prune the search tree, avoiding unnecessary exploration of branches that won't affect the game's result. This efficiency enables quicker decision-making in Tic Tac Toe.