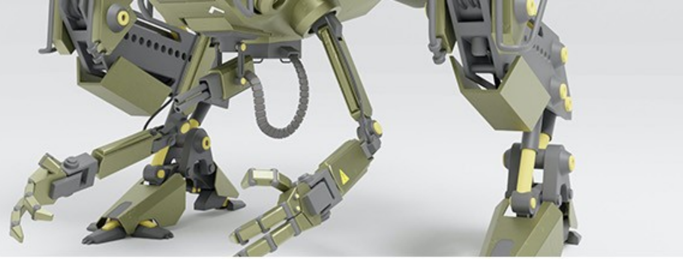
UNIT III

Adeversarial Search and Games

Contents

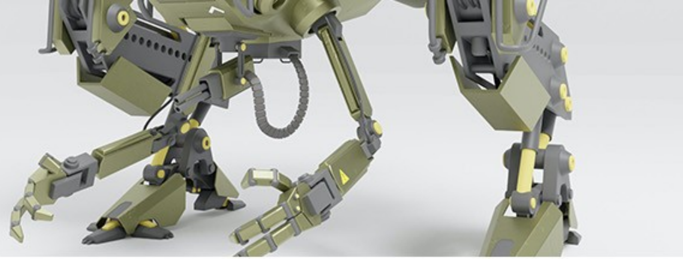
Unit III	Adversarial Search and Games	07 Hours
Game Theory, Optimal Decisions in Games, Heuristic Alpha–Beta Tree Search, Monte Carlo Tree Search, Stochastic Games, Partially Observable Games, Limitations of Game Search Algorithms, Constraint Satisfaction Problems (CSP), Constraint Propagation: Inference in CSPs, Backtracking Search for CSPs.		
#Exemplar/Case Studies	Machine Learning At Google: The Amazing Use Case Of Becoming A Fully Sustainable Business	
***Mapping of Course Outcomes for Unit III**	CO3, CO4	

Monte Carlo Tree Search (MCTS)



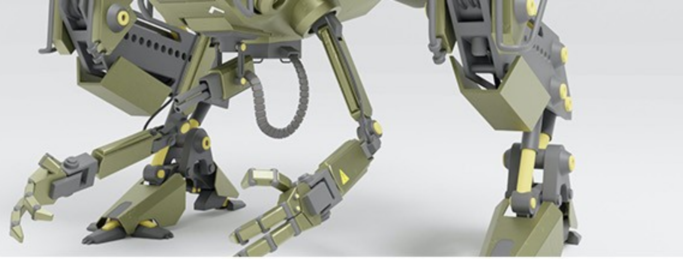
- Monte Carlo Tree Search (MCTS) is a heuristic search set of rules that has won big attention and reputation within the discipline of synthetic intelligence, specially in the area of choice-making and game playing.
- It is known for its ability to effectively handle complex and strategic video games with massive search areas, in which traditional algorithms may additionally struggle due to the full-size number of feasible actions or actions.

Monte Carlo Tree Search (MCTS)



- MCTS has been efficiently implemented in numerous domains, including board games (e.G., Go, chess, and shogi), card video games (e.G., poker), and video games.
- In tree search, there’s always the possibility that the current best action is actually not the most optimal action. In such cases, MCTS algorithm becomes useful as it continues to evaluate other alternatives periodically during the learning phase by executing them, instead of the current perceived optimal strategy.
  This is known as the ” exploration-exploitation trade-off “.
- It exploits the actions and strategies that is found to be the best till now but also must continue to explore the local space of alternative decisions and find out if they could replace the current best.
- Exploration helps in exploring and discovering the unexplored parts of the tree, which could result in finding a more optimal path.
- In other words, we can say that exploration expands the tree’s breadth more than its depth.

Monte Carlo Tree Search (MCTS)

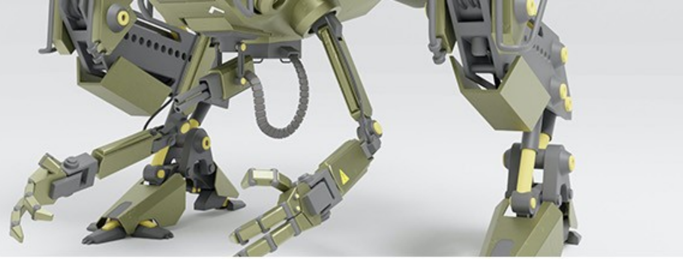


- Exploitation sticks to a single path that has the greatest estimated value. This is a greedy approach and this will extend the tree's depth more than its breadth.
- MCTS becomes particularly useful in making optimal decisions in Artificial Intelligence (AI) problems.

Monte Carlo Tree Search (MCTS)

- Why use Monte Carlo Tree Search (MCTS) ?
- ➔ Handling Complex and Strategic Games
- ➔ Unknown or Imperfect Information
- ➔ Learning from Simulations
- ➔ Optimizing Exploration and Exploitation
- ➔ Scalability and Parallelization
- ➔ Applicability Beyond Games
- ➔ Domain Independence

Monte Carlo Tree Search (MCTS)

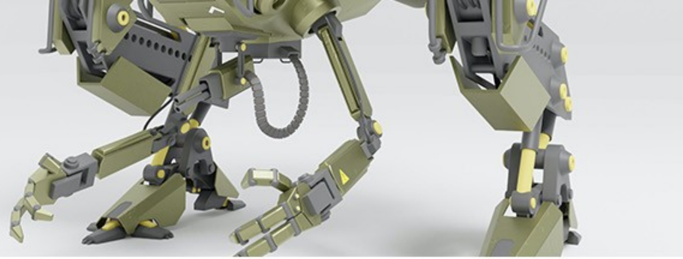


- The process of Monte Carlo Tree Search can be broken down into four distinct steps, viz., selection, expansion, simulation and backpropagation.
- **Selection:** In this process, the MCTS algorithm traverses the current tree from the root node using a specific strategy. The strategy uses an evaluation function to optimally select nodes with the highest estimated value.

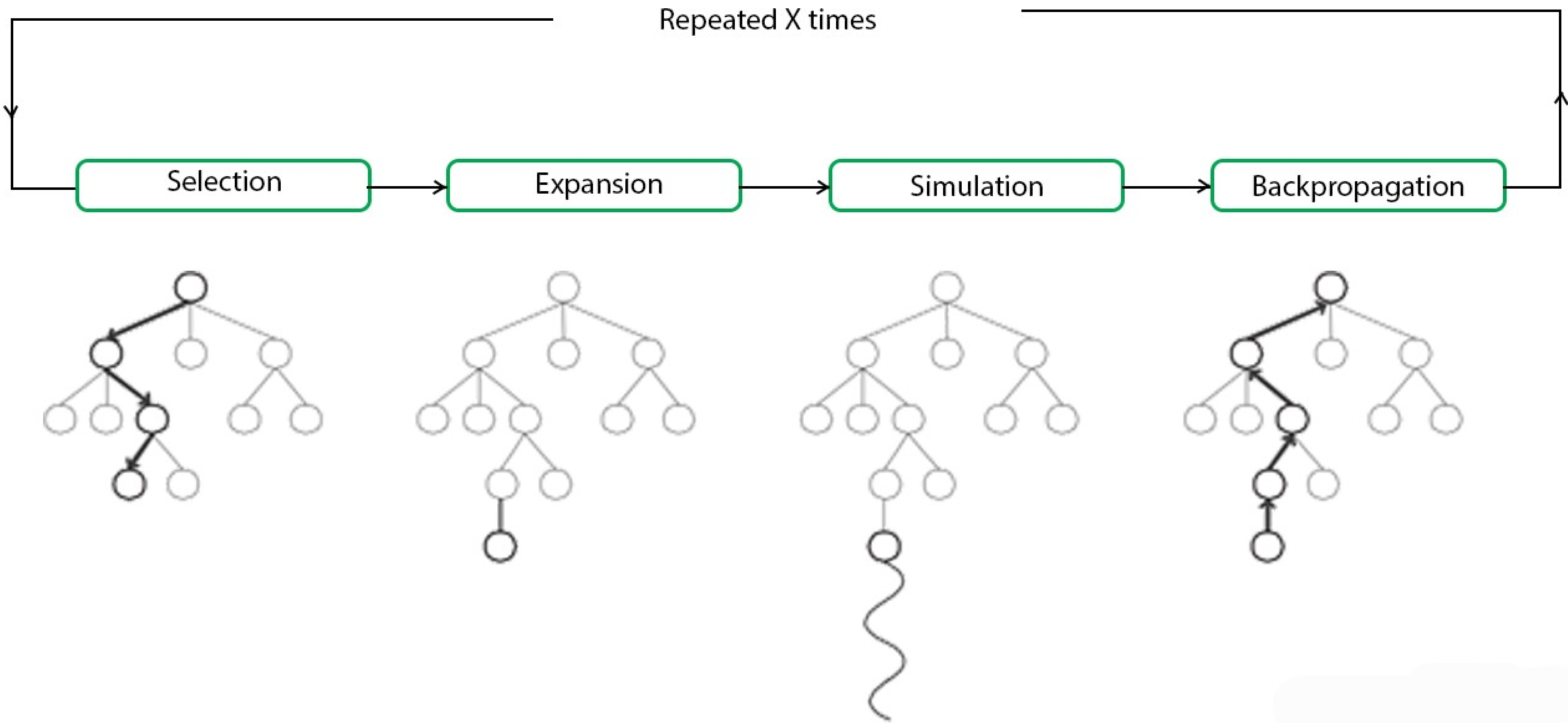
  It balances the exploration-exploitation trade-off. During tree traversal, a node is selected based on some parameters that return the maximum value.

  When traversing a tree during the selection process, the child node that returns the greatest value from the above equation will be one that will get selected. During traversal, once a child node is found which is also a leaf node, the MCTS jumps into the expansion step.

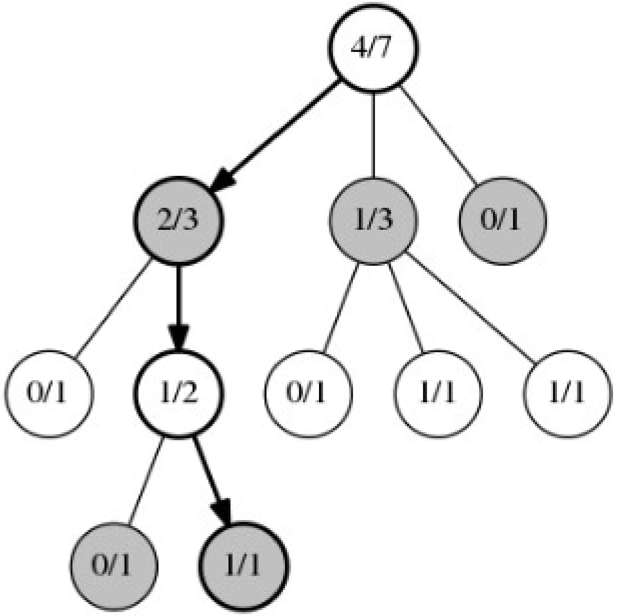
Monte Carlo Tree Search (MCTS)

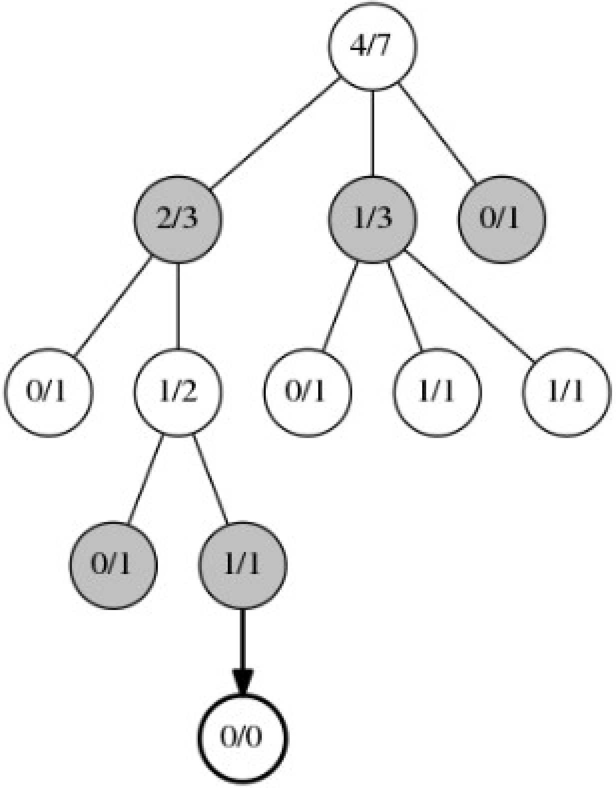


- **Expansion**: In this process, a new child node is added to the tree to that node which was optimally reached during the selection process.
- **Simulation**: In this process, a simulation is performed by choosing moves or strategies until a result or predefined state is achieved.
- **Backpropagation**: After determining the value of the newly added node, the remaining tree must be updated. So, the backpropagation process is performed, where it backpropagates from the new node to the root node. During the process, the number of simulation stored in each node is incremented. Also, if the new node's simulation results in a win, then the number of wins is also incremented..

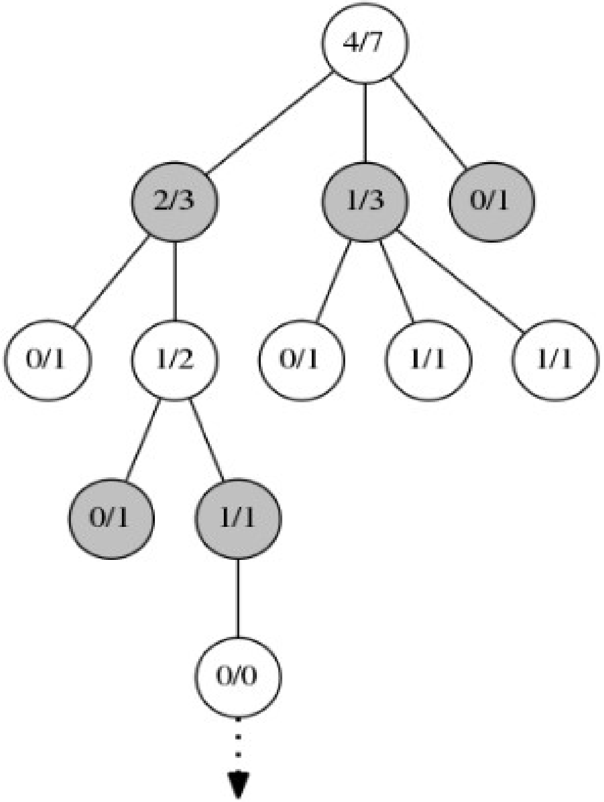

Repeated X times
Selection
Expansion
Simulation
Backpropagation

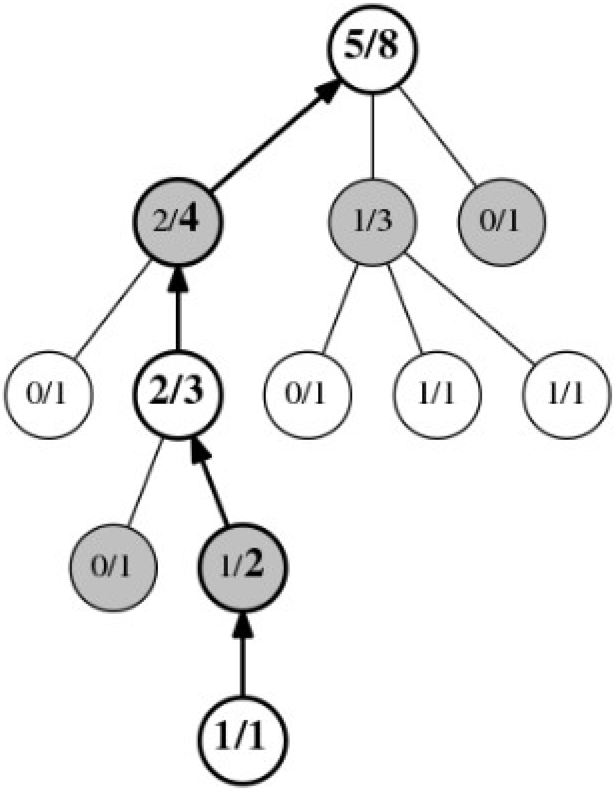
Monte Carlo Tree Search (MCTS)

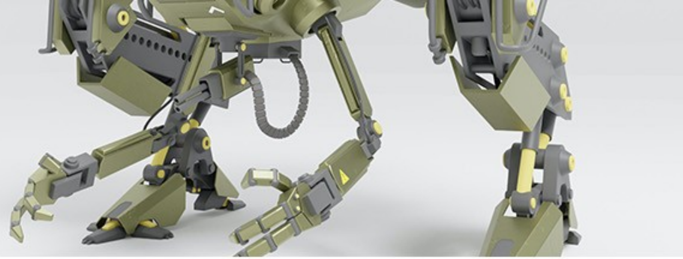




Monte Carlo Tree Search (MCTS)



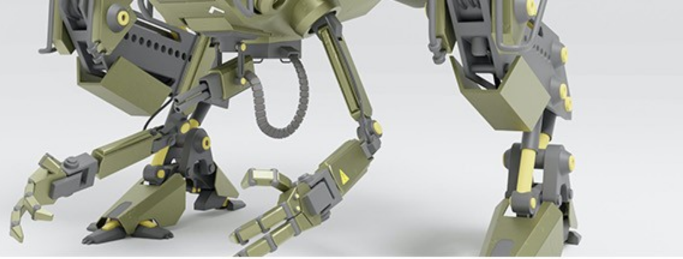




Monte Carlo Tree Search (MCTS)

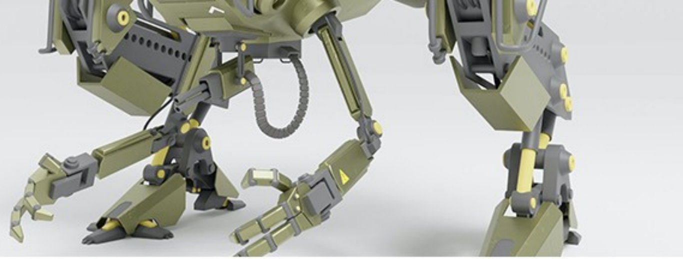
- Advantages of Monte Carlo Tree Search:
- MCTS is a simple algorithm to implement.
- Monte Carlo Tree Search is a heuristic algorithm. MCTS can operate effectively without any knowledge in the particular domain, apart from the rules and end conditions, and can find its own moves and learn from them by playing random playouts.
- The MCTS can be saved in any intermediate state and that state can be used in future use cases whenever required.
- MCTS supports asymmetric expansion of the search tree based on the circumstances in which it is operating.

Monte Carlo Tree Search (MCTS)



- Disadvantages of Monte Carlo Tree Search:
- ➔ As the tree growth becomes rapid after a few iterations, it requires a huge amount of memory.
- ➔ There is a bit of a reliability issue with Monte Carlo Tree Search. In certain scenarios, there might be a single branch or path, that might lead to loss against the opposition when implemented for those turn-based games. This is mainly due to the vast amount of combinations and each of the nodes might not be visited enough number of times to understand its result or outcome in the long run.
- ➔ MCTS algorithm needs a huge number of iterations to be able to effectively decide the most efficient path. So, there is a bit of a speed issue there.

Monte Carlo Tree Search (MCTS)



- Issues in Monte Carlo Tree Search:
- ➔ Exploration-Exploitation Trade-off
- ➔ Sample Efficiency
- ➔ High Variance
- ➔ Heuristic Design
- ➔ Computation and Memory Requirements
- ➔ Overfitting
- ➔ Domain-specific Challenges