



Introduction to Artificial Intelligence (CS-487)

Assignment #2

Due on Nov 13, 2022

Professor I. Tsamardinos

University of Crete
Department of Computer Science

Nikolaos Kougioulis (ID 1285)

7 November 2022

Exercise 1

Give the name of the algorithm that results from each of the following special cases:

1. Local beam search with $k = 1$.
2. Local beam search with one initial state and no limit on the number of states retained.
3. Simulated annealing with $T = 0$ at all times (and omitting the termination test).
4. Genetic algorithm with population size $N = 1$.

Solution:

1. Local Beam Search with $k = 1$ keeps track of only one state, so it represents *Hill-Climbing*.
2. Local Beam Search with one initial state and no limit on the number of states retained results in *Breadth-First Search*, as the algorithm expands the frontier.
3. Simulated annealing with $T = 0$ selects the first next state with the optimal Δx so it resembles *Best-First Hill-Climbing*.
4. A Genetic Algorithm with population size $N = 1$ performs a (trivial) selection on the single individual of the population, the two parents are the same, which then results in the same crossover and a random mutation, so it results in a *random walk*¹ (or a *Brownian motion*² in the case of continuous time and space) with probability equal to the mutation rate.

¹<https://mathworld.wolfram.com/RandomWalk1-Dimensional.html>

²<https://mathworld.wolfram.com/BrownianMotion.html>

Exercise 2

Which of the following are true and which are false? Give brief explanations.

1. In a fully observable, turn-taking, zero-sum game between two perfectly rational players, it does not help the first player to know what strategy the second player is using—that is, what move the second player will make, given the first player's move.
2. A perfectly rational backgammon agent never loses.

Solution:

1. **False.** In a fully observable, turn-taking, zero-sum game is it optimal to know the strategy of the opponent. This statement would be true in the case of a *partially* observable, turn-taking, zero-sum game.
2. **False.** Backgammon is a *stochastic* game, so even a perfectly rational agent may lose, due to the factor of stochasticity involved in rolling a die (or dice). Success in the game does not only depend on the agent's skill, but also on pure luck.

Exercise 3

Discuss how well the standard approach to game playing would apply to games such as tennis, pool, and croquet, which take place in a continuous physical state space.

Solution:

One way to approach these problems, discussed in Chapter 4, is to discretize the continuous state space, based on a spacing of size ϵ .

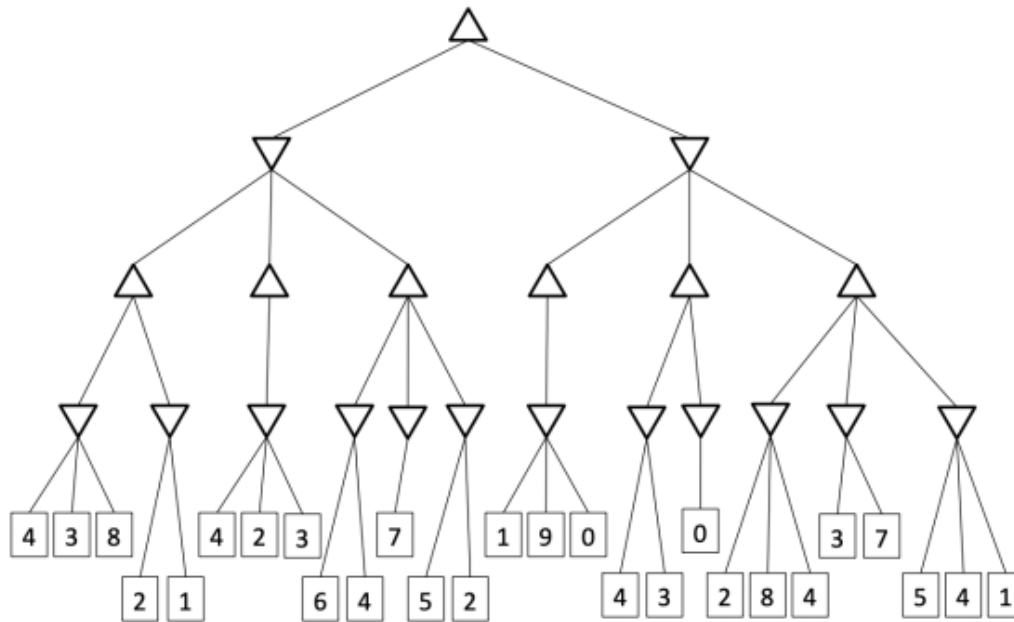
In the case of tennis, by discretizing states such as the speed of the racket hit and the positioning of the ball one may yield average results. One may perform Minimax for alternating turns of ball direction (left opposite corner, middle, right opposite corner) and ball speed (low, medium, fast), or even ExpectMiniMax after performing a Monte Carlo simulation to deduce the necessary probability distributions.

But in the case of pool and croquet, such approach must be mediocre at best. Both pool (8-ball pool or snooker) and croquet are sports that require great amounts of accuracy, with very small differences between a winning and a losing state; the positioning of the mallet, the balls and the hoops (not even taking into consideration the anomalies on the grass surface), as well as the positioning of the cue and the balls, are highly complex environments that will suffer from discretization techniques.

However one can consider individual tasks using local search in continuous spaces (*Chapter 4, Section 2*). For example, in 8-ball pool the goal of winning consists of potting all single color (or striped) balls and the 8-ball last. A local-search algorithm (like the empirical gradient method or Newton-Raphson's method) can indeed find the optimal way to achieve every local goal like "run the hoop", "play the correct ball" for croquet³ or "break the rack", "pot ball numbered k", "pot the 8-ball"⁴, which are definitely more approachable methods than trying to learn an agent how to play human-like tennis in a continuous environment.

³<https://www.croquet.org.uk/?p=games/garden/rules/GardenGolfRules>

⁴<https://upatour.com/8-ball-rules/>

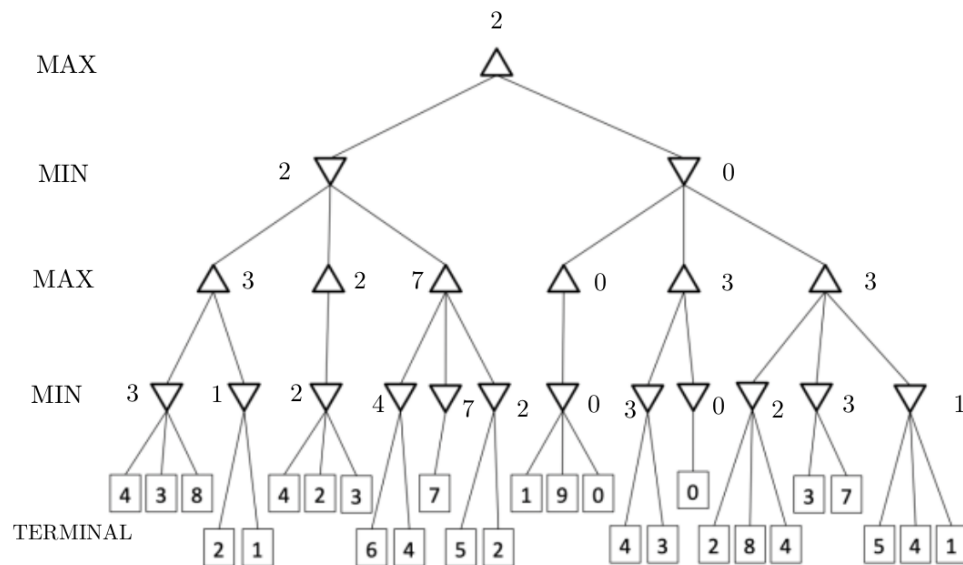


Exercise 4

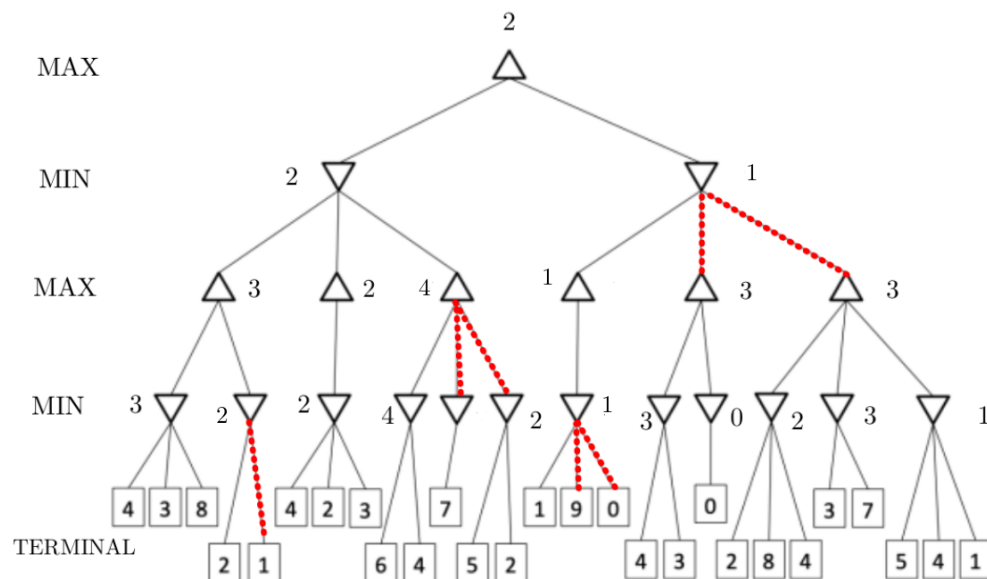
1. Give the values calculated by Minimax for all states in the tree. Do not use alpha-beta pruning.
2. Indicate which branches of the tree will be pruned by alpha-beta pruning.

Solution:

Continued on next page...



Solution to Question 1: Application of Minimax for all states of the tree, calculated iteratively from the terminal states to the root state. Minimax performs a complete depth-first exploration of the game tree. If there are b legal moves at each point and the depth of the tree is m , then the time complexity of Minimax is $\mathcal{O}(b^m)$ and the space complexity $\mathcal{O}(bm)$ respectively.



Solution to Question 2: Due to the exponential complexity of the Minimax algorithm, it is viable to find the actual Minimax decision without even looking at every node of the game tree, using alpha-beta pruning. By applying alpha-beta pruning to the Minimax algorithm, the same moves get returned but nodes that are possibly not affecting the final decision are removed (pruned) from the search tree.