

Übungsblatt

Grundlagen der Künstlichen Intelligenz

03.11.2023, DHBW Lörrach

Beantworten Sie die Fragen zum Pseudo-Code in Python für die Monte Carlo Baumsuche:

```
1  # main function for the Monte Carlo Tree Search
2  def monte_carlo_tree_search(root):
3      while resources_left(time, computational power):
4          leaf = traverse(root)
5          simulation_result = rollout(leaf)
6          backpropagate(leaf, simulation_result)
7      return best_child(root)
8
9  # function for node traversal
10 def traverse(node):
11     while fully_expanded(node):
12         node = best_uct(node)
13
14     # in case no children are present / node is terminal
15     return pick_unvisited(node.children) or node
16
17 # function for the result of the simulation
18 def rollout(node):
19     while non_terminal(node):
20         node = rollout_policy(node)
21     return result(node)
22
23 # function for randomly selecting a child node
24 def rollout_policy(node):
25     return pick_random(node.children)
26
27 # function for backpropagation
28 def backpropagate(node, result):
29     if is_root(node) return
30     node.stats = update_stats(node, result)
31     backpropagate(node.parent)
32
33 # function for selecting the best child
34 # node with highest number of visits
35 def best_child(node):
36     pick child with highest number of visits
```

Pseudo-Code für Monte Carlo Baumsuche in Python. Aus: <https://www.geeksforgeeks.org/ml-monte-carlo-tree-search-mcts/>

1. Erläutern Sie die Strategie für das Durchlaufen des Suchbaumes im Monte Carlo Algorithmus, die durch die Auswertungsfunktion `best_uct(node)` (s. Programmzeile 12) zum Ausdruck gebracht wird.

2. Welchen Vorteil bietet die Aktualisierung der Knoten-Statistik (s. Programmzeile 30) basierend auf dem Ergebnis (z.B. Erfolg oder Niederlage) eines Playouts (auch: Roll-Out, s. Funktion in Programmzeile 18) für die Auswahlstrategie?

3. Nennen Sie jeweils einen Unterschied und eine Gemeinsamkeit zwischen der Monte Carlo Baumsuche und der Minimax Suche.