Basics of AI and Machine Learning

Monte-Carlo Tree Search

Jendrik Seipp

Linköping University

Slides modified from Basel AI group, with permission

Monte-Carlo Tree Search

Monte-Carlo Tree Search: Idea

Monte-Carlo Tree Search (MCTS) ideas:

- perform iterations as long as resources (deliberation time, memory) allow:
- build a partial game tree, where nodes n are annotated with
  - utility estimate $\hat{u}(n)$
  - visit counter $N(n)$
- initially, the tree contains only the root node
- each iteration adds one node to the tree

After constructing the tree, play the move that leads to the child of the root with highest utility estimate (as in minimax).

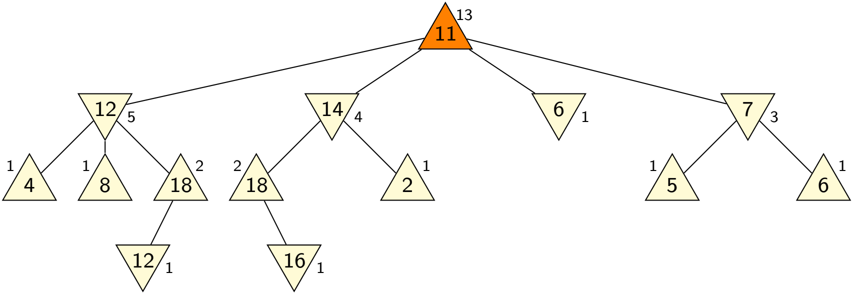
Monte-Carlo Tree Search: Iterations

Each iteration consists of four phases:

- selection: traverse the tree by applying tree policy
  - Stop when reaching terminal node (in this case, set n_{child} to that node and $p_\star$ to its position and skip next two phases)...
  - ...or when reaching a node n_{parent} for which not all successors are part of the tree.
- expansion: add a missing successor n_{child} of n_{parent} to the tree
- simulation: apply default policy from n_{child} until a terminal position $p_\star$ is reached
- backpropagation: for all nodes n on path from root to n_{child}:
  - increase $N(n)$ by 1
  - update current average $\hat{u}(n)$ based on $u(p_\star)$

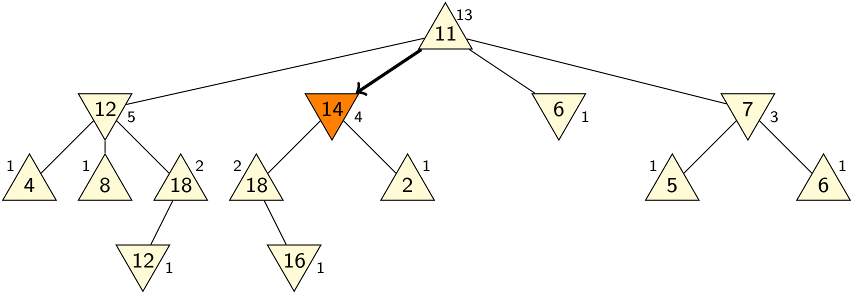
Monte-Carlo Tree Search

Selection: apply tree policy to traverse tree



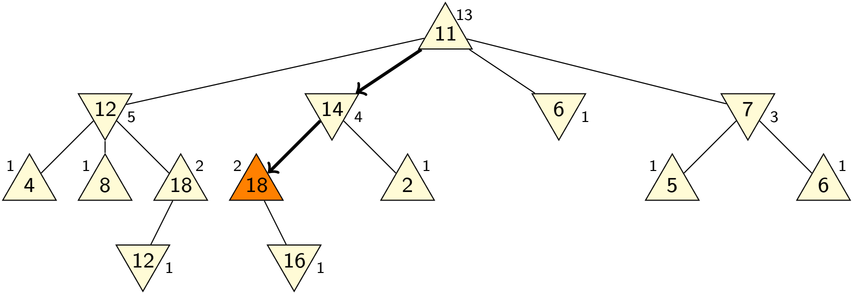
Monte-Carlo Tree Search

Selection: apply tree policy to traverse tree



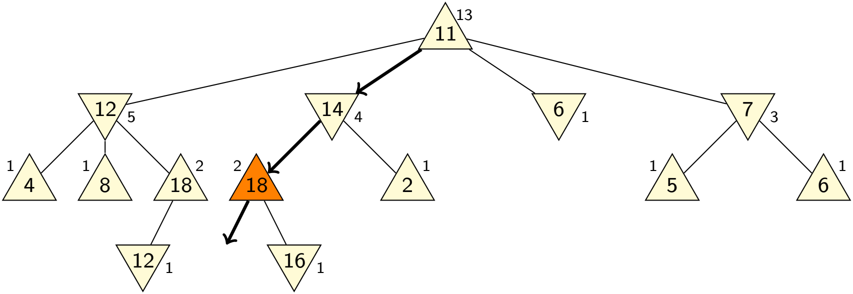
Monte-Carlo Tree Search

Selection: apply tree policy to traverse tree



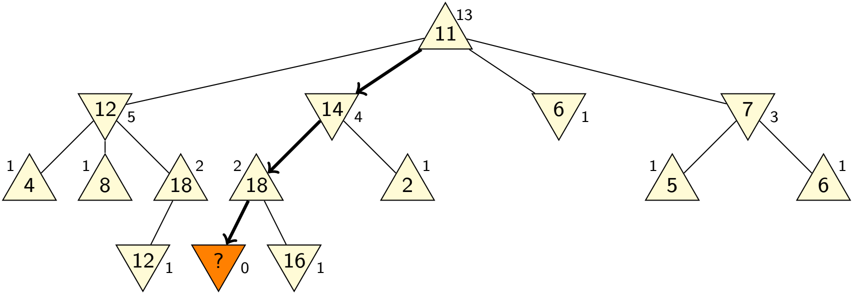
Monte-Carlo Tree Search

Selection: apply tree policy to traverse tree



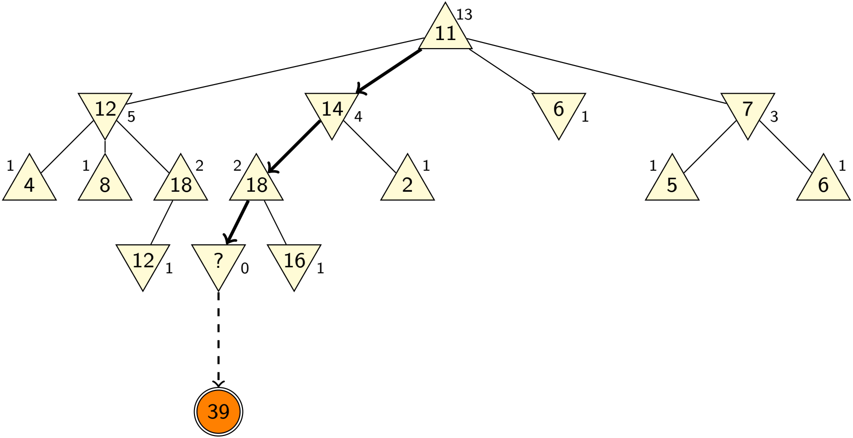
Monte-Carlo Tree Search

Expansion: create a node for first position beyond the tree



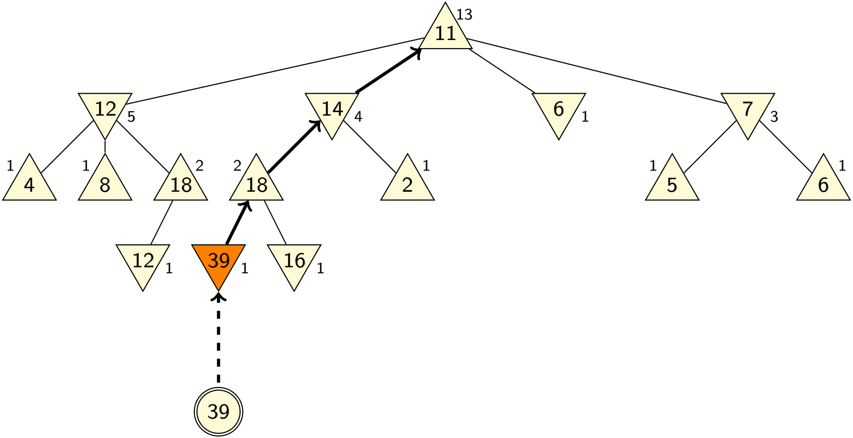
Monte-Carlo Tree Search

Simulation: apply default policy until terminal position is reached



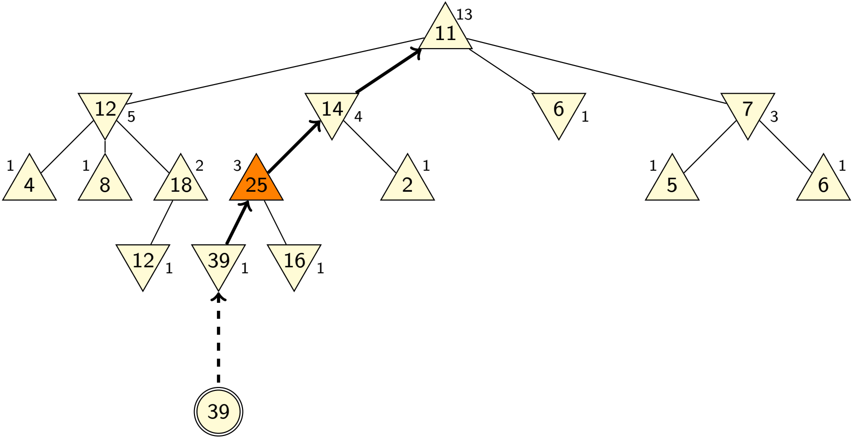
Monte-Carlo Tree Search

Backpropagation: update utility estimates of visited nodes



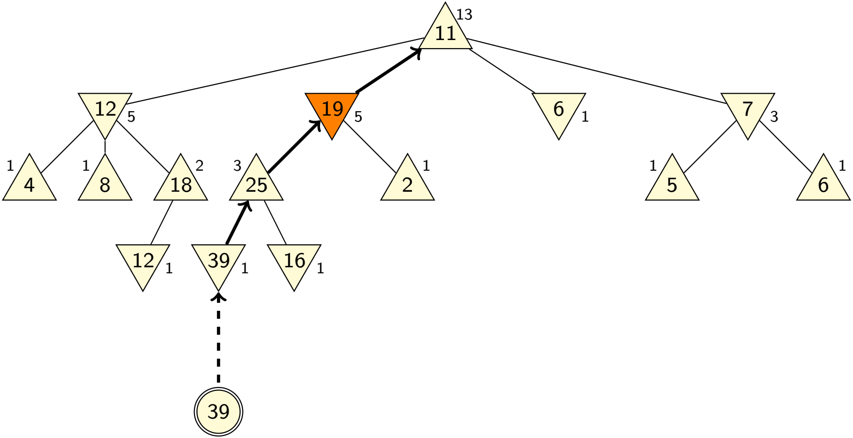
Monte-Carlo Tree Search

Backpropagation: update utility estimates of visited nodes



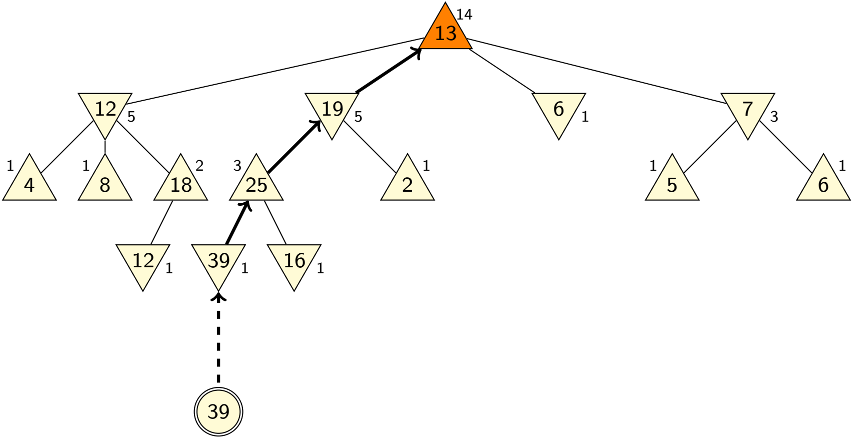
Monte-Carlo Tree Search

Backpropagation: update utility estimates of visited nodes



Monte-Carlo Tree Search

Backpropagation: update utility estimates of visited nodes



MCTS in AlphaGo

AlphaGo computes four neural networks:

- supervised learning (SL) policy network
  ⇝ for prior probabilities
- rollout policy network
  ⇝ for default policy in simulation phase
- reinforcement learning (RL) policy network
  (intermediate step only)
- value network
  ⇝ for heuristic in simulation phase

Summary

Summary

- Monte-Carlo Tree Search (MCTS) algorithms iteratively build a search tree, adding one node in each iteration.
- MCTS is parameterized by a tree policy and a default policy.