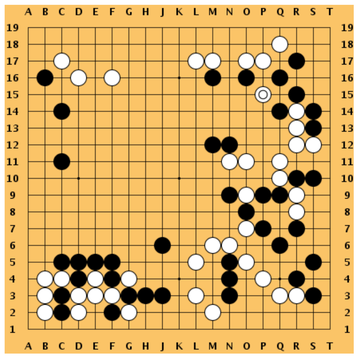
LEARN TO PLAY GO

ALBERT LIU (ALBERTPL@STANFORD.EDU)

MOTIVATION & CHALLENGES



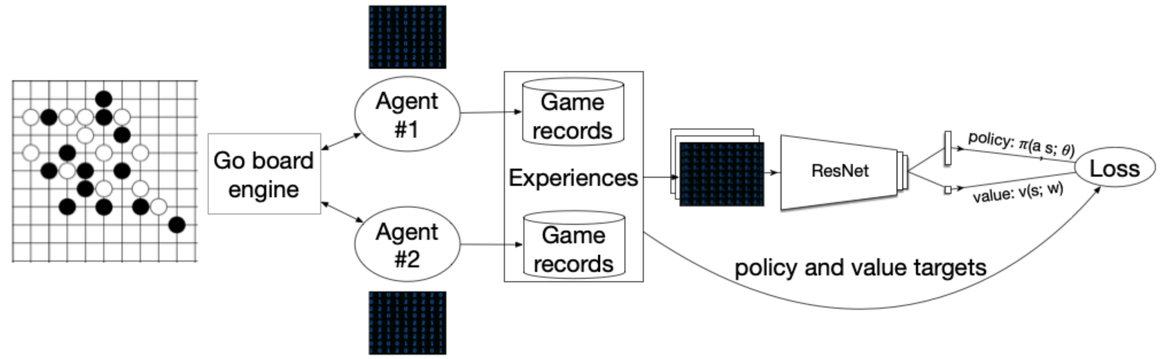
- simple rule but mastering requires many years of study by human
- state based search is intractable due enormous search space ($\sim 10^{170}$), large number of legal move per state(~ 250)
- the difficulty to handcraft a heuristic evaluation function
- AlphaGo and AlphaGoZero have rocked the Go and AI world

PROBLEM DEFINITION

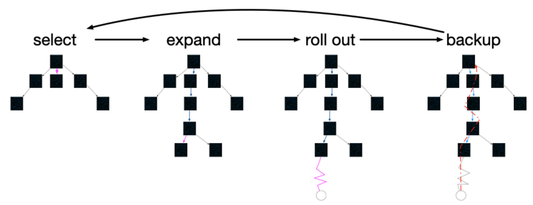
Turn-taking, two player, fully observable zero-sum game.

- The state space is all possible placements of the stones on the board and the player of that turn.
- The action space is any legal position of a stone and a pass action.
- The goal is to maximize territory, by observing the board only.

SYSTEM ARCHITECTURE



MONTE CARLO TREE SEARCH



Input: root node s_0: current game state
Input: $c > 0$: parameter to control the degree of exploration

while *within computation bound* **do**
  $s \leftarrow s_0$
  $\Delta \leftarrow \emptyset$
  // selection based on tree policy: Upper Bound Confidence
  while s *is in the tree* **do**
    $a \leftarrow \arg\max\{q(s,a) + c\sqrt{\frac{\log \sum_{a'} n(s,a')}{n(s,a)}}\}$
    $\Delta \leftarrow \Delta \cup \{(s,a)\}$
    $s \leftarrow \text{Succ}(s,a)$
  expand tree with the new node s
  continue the game from s with random policy and let r be the reward
  // update each node on the path $s_0 \rightarrow ... \rightarrow s$
  for $s, a \in \Delta$ **do**
    $n(s,a) \leftarrow n(s,a) + 1$
    $q(s,a) \leftarrow q(s,a) + \frac{1}{n(s,a)}(r - q(s,a))$
return $\arg\max_a q(s_0, a)$

REINFORCEMENT LEARNING

PRELIMINARY RESULTS

ANALYSIS