

atax.TX

Tri Kurniawan Wijaya & Xi Luo

Dresden, 31 March 2010

System Overview

- Java Basic Player
- Prolog Prover
- Multiplayer games:
 - Monte Carlo Tree Search (MCTS) with UCT
- Singleplayer games:
 - Start Clock: DFS & Iterative Deepening
 - Play Clock: MCTS with UCT

Multiplayer Games

■ MCTS with UCT

- Goal value for each player:
 - $(G1 - AvgOthers) * 100 + G1$.
 - Where $G1$ is the real goal value according to the game rules, $AvgOthers$ is the average of the goal value (according to the game rules) of the other players.
- Action selection:

$$a^* = \operatorname{argmax}_{a \in A(s)} \left\{ Q(s, a) + C \sqrt{\frac{\ln N(s)}{N(s, a)}} \right\}$$

- Average visit of each action is used instead of $N(s)$
- $C = 8000$

Multiplayer Games (contd.)

■ TO-MAST

- $\tau = 1000$
- Used only in playout phase for the action selection

$$\mathcal{P}(a) = \frac{e^{Q_h(a)/\tau}}{\sum_{b=1}^n e^{Q_h(b)/\tau}}$$

■ RAVE

- Modification for the action selection:

$$\beta(s) \times Q_{RAVE}(s, a) + (1 - \beta(s)) \times Q(s, a)$$

$$\beta(s) = \sqrt{\frac{k}{3n(s) + k}}$$

- Instead of $\beta(s)$, $\beta(a)$ is used by replacing $n(s)$ with $n(s, a)$
- $k = 30$
- The information gathered from the nodes in the tree only instead of the entire episode

Singleplayer Games

- Start clock
 - If “step counter” is exist:
 - Eliminate the step counter
 - DFS with hash table
 - The state is replaced if the same state with smaller depth is found
 - Iterative Deepening with hash table
- Play clock
 - MCTS with UCT (similar with the one in multiplayer games)
 - $C = 400000, \tau = 500000, k=30$
- Each time a terminal state with a better score is found:
 - The path from the current state to the terminal state is saved
- Whenever terminal states with value= 100 is encountered:
 - the search procedure is stopped
 - the path from the current state to the terminal is saved