Game Tree Searching by Min / Max Approximation

This paper uses the idea of approximating the “min” and “max” operators by generalised mean-valued operators to find out which node to expand in min/max game tree. The authors explore two different constraints: Time constraint and Player movement constraints in their experiment

Min/max approximation strategy uses the penalty-based iterative search method:

- The algorithm assigns a weight value to each edge between parent and child nodes.
- A low weight score is assigned to an edge when it is selected,
- Unselected edges get a higher scored compared with selected ones
- For choosing the node to expand, the algorithm selects the branch with the smallest sum from bottom node to root.

The experiment compares and contrasts min/max with alpha-beta pruning strategy with the penalty based approach which min/max approximation strategy uses. In the experiment, it takes Connect-Four game as well as resource bounds (time constraints and move constraints) as rules in the game.

The result is that when using time constraints, this approximation method performs worse than alpha-beta pruning, but in case of a limited number of moves, it outperforms alpha-beta pruning. The reason is when using min/max approximation; computing generalised mean value consumes more time. When using move constraint, each move uses more information, so it performs better than alpha-beta pruning.

To conclude, Min/max approximation needs more computation resources compared with tradition min/max calculation. Game tree searching by min/max approximation outperforms alpha-beta pruning strategies when basing the constraint on movements rather than time.