

Research Review

June 27, 2017

1 Technical Summary - Game Tree Searching by Min / Max Approximation* by Ronald L. Rivest

1.1 Abstract

The paper describes an iterative penalty based method for searching minimax game trees based on approximating the min and max operators by generalized mean-valued operators. For minimax game trees, a player is always trying to expand the next node which has largest effect on the game outcome i.e. the move that has highest probability of winning the game. Proposed approach is to approximate the min and max operators with generalized mean-value operators. Author argues that generalized mean value operators are good approximation to the min/max operators with higher sensitivity due to its continuous derivative. Author also argues that under certain conditions, penalty based iterative search gives superior performance as compared to alpha-beta search which is a greedy algorithm. In the proposed approach, each parent and child node share a penalty across the edges which indicates the influence of child node on the parent node. It assigns a nonnegative penalty to every edge in the game tree such that edges representing bad moves are penalized more than edges representing good moves. To choose the next move, it expands the tip node t which has the least penalty. Penalty across the edges are back propagated up to root of the tree, across the edges of child and parent nodes. A child node with least penalty is selected to move the game forward.

1.2 Results Summary

Experimental results indicate that proposed scheme outplays alpha-beta with iterative deepening, when both schemes are restricted to the same number of calls to the move operator. However, on time constrained search, alpha-beta outperforms. Results also outline that large amount of memory is required for the penalty based search as compared to greedy algorithms. Also, penalty based method optimizes the weights across the nodes, rather than selecting the best move to make from the root. For example, if there is only one move to make from the root, then a penalty-based scheme may search the subtree below that move extensively, even though such exploration can't affect the decision to be made at the root. The penalty on an edge was computed to be 0.05 plus the absolute value of the difference between the natural logarithm of the value of the node and the natural logarithm of the value of his "best" sibling. Author argues that results are sensitive to this constant but there seems no explanation on the chosen value and how it affects the results by changing this value.