# Review of the paper titled *Game Tree Searching by Min/Max Approximation*

Kai Schoos

July 23, 2017

## 1 Summary of the goals and techniques introduced

The paper by Ronaldo L. Rivest introduces the technique of min/max approximation. He tackles the problem produced by using the minimum(.) and maximum(.) functions in the minimax search algorithm with alpha-beta-pruning, which are not continuous and can therefor not be derived. By using generalized mean functions in the form:

$$M_p(a) = (\frac{1}{n}\sum_{i=1}^{n} a_i^p)^{1/p} \tag{1}$$

and letting p grow towards infinity respectively to negative infinity, the maximum and minimum values can be approximated by a continuous function. For large values of p, the derivative with respect to variables with values closer to the maximum grows towards 1, and the derivative with respect to variables with values further away from the maximum grow towards 0 ( or the minimum, respectively ).

When applying this function to the values returned from the evaluation functions of the nodes from a game tree (much like one would apply the min and max functions to the nodes of a game tree when using basic min/max search), one can estimate how much a given node influences the root value. The bigger the influence, the higher the chance for of being expanded.

The negative logarithm of the values gained by this approach are summed along the edges up to the root of the tree and this sum is denoted the *penalty* of choosing this branch. The leaf node that has the smallest penalty (which is the one with the highest influence on the root node) will be expanded next.

## 2 Summary of the results

The results of the technique that are proposed are interesting. The min-max-approximation algorithm does expand up to 4 times less nodes than classic minimax techniques with alpha-beta pruning. It however uses a lot more computational resources than the latter and is therefor slower in terms of raw CPU cycles. When the limiting resource is the amount of expanded nodes, the newly proposed technique can be a lot faster than classic minimax with alpha-beta pruning. If the limiting resource however is time / clock cycles, minimax is still ahead.