Game Tree Searching by Min / Max Approximation

This document discusses the research publication “Game Tree Searching by Min / Max Approximation, Ronald L.Rivest“

Goals

The author aims to optimize the Min/Max Game Tree Search by first expanding into subtrees that have the highest uncertainty in terms of their estimated value. Or in other words, to first explore (expand) those subtrees that could most likely change an estimate higher up in the game tree. By doing so we can more accurately estimate the overall game situation before a resource constraint is hit (e.g.: time, or number of calls).

Techniques

Sensitivity analysis

The author introduces a way to calculate the ‘sensitivity’ of an estimated game state value to a potential better estimate of one of its child game states. If the sensitivity is low further exploring a certain child game state to get a more accurate estimate is unnecessary, as it will not change the estimated value higher up in the game tree significantly compared to other lower game state that have a higher sensitivity.

To measure the sensitivity the author replaces the Min/Max function with a differentiable approximation. By doing so the differentiable approximation can be partially derived in respect to a certain line of play. If the value of the derivative is low (e.g. 0) further exploring that line of play is not that important, as the estimated value will not change that much. The opposite is true if the value of the derivative is high. Then it’s to be expected that further exploring that line of game will more significantly influence the already known estimates.

Penalty base iterative search methods

The afore mentioned idea that game states with low ‘sensitivity’ don’t need to be further explored is implemented by a penalty based iterative search. This algorithm allows for the idea that certain edges in the game tree have high penalties which ‘discourage’ the iterative search to explore them further. The author introduces a way to relate ‘sensitivity’ to penalty. The sensitivity is inverse to the penalty (nonlinear). This means lower ‘sensitivity’ leads to higher penalty and therefore lines of play with lower ‘sensitivity’ are more reluctantly explored by the search algorithm.

Results

The author notes that the introduced technique yields better results than traditional Min/Max-Search with Alpha/Beta-pruning in case the number of calls to the “move” operator (“move” meaning calculating new game states) is the limiting factor. In case pure CPU time is the most limiting factor traditional Min/Max-Search with Alpha/Beta-pruning performs better. Its shown that Min/Max Approximation checked muss less game states and yet was able to outperform Min/Max-Search with Alpha/Beta-pruning.

It’s said the Min/Max-Approximation could be further improved in future by maybe not always evaluating the globally least-penalty tip. A probabilistic approach could maybe help here.

Jörg Jenni, The Old Curates House, Horton Street, Heckmondwike, WF16 0LL, United Kingdom
Tel: +44 (0) 7872 463615
Mail: jjenni@gmx.ch