Summary: Game Tree Searching by Min / Max Approximation

Goal

In order to obtain expert-level play, the goal of this paper is to suggest a method which always expand the node that is expected to have the largest effect on the value. It focuses the computer's attention on the important lines of the play. The key idea is to approximate the "min" and "max" operators with generalized mean-value operators.

Techniques

1. Generalized mean values: The generalized mean values to approximate the min and max functions, we can identify in an interesting way that leaf in a game tree upon whose value the value at the root depends most strongly. This is done by taking derivatives of the generalized mean value functions at each node and using the chain rule. This leaf will be the one to expand next.
2. Game Tree Searching: When game is small, the tree can be explored completely, so optimal play is possible. On slightly larger trees minimax search with alpha-beta pruning may produce optimal play even though only a small fraction of the game tree is explored. For most interesting games the game tree is so large that heuristic approximations are needed. Penalty based iterative search method applies weight to every edge in the game. Bad moves are penalized. To identify the weight, reverse approximation was implemented. The "min/max approximation" heuristic is special case of the penalty-based search method, where the penalties are defined in terms of the derivatives of the approximating functions. Each leaf of the partial tree to be expanded is going to have the sum of the edges from the root to the leaf. The less value leaf is the best candidate to be expanded.

Results

Author has chosen Connect-Four game for experimentation. For each experiment, 49 different starting positions were considered. For each starting position, two games were played--one with alpha-beta (AB) moving first, and one with min/max approximation (MM) moving first. Thus, a complete experiment consists of 98 games. For each experiment, it was recorded how many times each strategy won, and how many ties occured. One experiment was run for each of five posible time bounds (1 second to 5 seconds, in one-second intervals), and for five possible move bounds (1000 moves to 5000 moves, in 1000-move increments). Thus, 490 games were played for each resource bound, and 980 games played altogether.

1. Based on time usage alone, alpha-beta seems to be superior to implementation of the min/max approximation approach.
2. But if we base our comparison on move-based resource limits, min/max approximation is superior.
3. Implementation of minimax search with alpha-beta pruning called the move operator approximately 3500 times per second, implementation of the min/max heuristic called the move operator approximately 800 times per second.
4. Penalty-based schemes may not perform well unless they are given a large amount of memory to work with.
5. "Min/max approximation" heuristic will allocate resources in a sensible manner, searching shallowly in unpromising parts of the tree, and deeper in promising sections.

Conclusion

Min/max approximation outplays alpha-beta with iterative deepening, when both schemes are restricted to the same number of calls to the move operator.