

# Research Review

---

I have read the three papers and in this summary will provide a brief review of all of them, but with emphasis on AlphaGo as it presents the state of the art in gameplay AI.

## Game Tree Searching by Min/Max Approximation

This paper uses a different search algorithm from the techniques presented during the course. In Minimax all the options are explored to find the best action, Alphabeta optimizes Minimax by ignoring nodes that will not impact the final decision. Instead this paper uses iterative search, which iteratively tries to select a single best course of action, explores it then backpropagate the result of exploration and see if that course of action is still the best or another path should be explored. This paper uses a Minimax approximation with generalized mean-valued operator to guide the selection of next leaf node to expand. Generalized mean values is an approximation to min/max values, but is derivable, which is more suitable to sensitivity analysis. It is used to calculate weights to determine which node, the value in the parent node depends the most on, and selects it for expansion. After trying this technique with 1000 games of Connect-Four the Min/Max approximation with iterative search proved to be superior to Alphabeta for the same number of calls to the basic move subroutine, but Alphabeta wins if the bound resource is time, because it is faster and could explore more nodes for the same allotted time.

## Deep Blue

This paper describes Deep Blue system, which defeated the chess world champion Garry Kasparov in 1997. The system combines specialized hardware with software to massively search a huge number of combinations for each move. The hardware consists of 30 CPUs, with 1GB of RAM and 4GB of disk per processor, 480 single chip chess search engines with 16 chip per processor, a move generator implemented as 8x8 array of combinatorial logic, which is effectively a silicon chess board. Deep Blue search is based on alpha-beta with ideas like quiescence search, iterative deepening, and transposition tables. The search algorithm is implemented as a hybrid software search with compiled C code running on CPU with hardware search implemented in the chess chip. Software search offers flexibility for modifications, hardware search is parametrized, but suffers from limitations of modifying hardware. For example hardware search lacks access to transposition tables, which is implemented in the software search. The algorithm is massively parallel with a single CPU running as the master distributing work to other processors. Deep Blue evaluation function recognizes roughly 8000 features, but implemented as a combination of fast, and slow evaluation. The fast evaluation computes only the easily evaluated major terms with high values when an approximation is good enough. The evaluation function is implemented in hardware on the chess chip. Deep Blue uses an opening book database created by hand primarily by Grandmaster Joel Benjamin and others. The book consisted of about 4000 positions. An extended open book with 700,000 was used to influence decisions in absence of opening book information by summarizing the information for a move in the extended book to nudge Deep Blue in the consensus direction of chess opening theory. Endgame database with all chess positions with five or fewer pieces was used off-line to design the chess chips, and the software search used the database on-line to guide the search algorithm.