Deep Blue Review

June 16, 2017

1 Deep Blue research review

The paper describes the computer chess system that defeated Garry Kasparov in 1997. Deep Blue was the result of a multi-year effort to build a world-class chess machine starting from earlier efforts in the 1980s. During these years of system improvements the evaluation function had been strongly redesigned going from around 6400 features to over 8000 and the chip search speed was increased to 2–2.5 million positions per second.

But what is Deep Blue? It's a massively parallel system designed for carrying out chess game tree searches. It's organized in three layers: The master layer searches the top levels of the chess game tree, then distributes 'leaf' positions to the workers for further examination. Then workers carry out a few levels of additional search, and finally distribute their leaf positions to the chess chips, which search the last few levels of the tree.

The paper describes how the fundamental evaluation function was implemented in Deep Blue showing that it was composed of a "fast evaluation" and a "slow evaluation" in order to skip computing an extensive full evaluation of the optimal solution. In particular fast evaluation computes a score for a chess position in a single clock cycle, while slow evaluation scans the chess board one column at a time. The search operation of the system was designed to follow an implementation of the null-window alpha-beta search in order to eliminate the need for a value stack, while the alpha-beta algorithm normally maintains two values, alpha and beta, on a stack.

Then the paper explores the technique of selective search, called "dual credit with delayed extensions", and explains how the hardware search is fast and the fact that it uses a null-window approach requires special handling of some parameters such as depth of search, depth of offset searches, etc

One of the main topic of the paper is a deep description of the Deep Blue evaluation function behaviour that is a sum of feature values with more than 8000 different patterns. A feature value can be either static or dynamic. Static values are set once at the beginning of a search. Dynamic values are also initialized at the beginning of a search, but during the search they are scaled, via table lookup, based on the value and type of pieces on the board at evaluation time. Then paper describes a detailed example of one table that gives a feel for the nature of the evaluation function.

In conclusion, the success of Deep Blue was the result of multiple factors. It's a concrete example of how difficult can be the design and the development of a world-class game agent. The paper lists some areas of improvements that could have been made and that there were many alternatives that weren't covered that have a strong pull for further exploration of this space.