A KNOWLEDGE-BASED APPROACH OF CONNECT-FOUR
The Game Is Solved: White Wins

Victor Allis

M.Sc. Thesis
Free University, Amsterdam
Department of Mathematics and Computer Science
The Netherlands
94 pages

[Although at first sight perhaps not of prime interest for our computer-chess community, the underlying report describes the investigation of a game, which, just like chess although much simpler, belongs to the class of *two-person zero-sum games with complete information.* Since the techniques used therefore apply equally well to computer-chess programs, we feel the present work worthy to be mentioned, - Eds.]

We quote the abstract:

"A Shannon C-type strategy program, VICTOR, is written for Connect-Four, based on nine strategic rules. Each of these rules is proven to be correct, implying that conclusions made by VICTOR are correct.

Using VICTOR, strategic rules were found which can be used by Black at least to draw the game, on each 7 x (2n) board, provided that White does not start at the middle column, as well as on any 6 x (2n) board.

In combination with conspiracy-number search, search tables and depth-first search, VICTOR was able to show that White can win on the standard 7 x 6 board. Using a database of approximate half a million positions, VICTOR can play in real time against opponents on the 7 x 6 board, always winning with White."

A HIGH-PERFORMANCE PARALLEL ALGORITHM TO SEARCH DEPTH-FIRST GAME TREES

Robert Morgan Hyatt

Ph.D. thesis
University of Alabama
Computer Science Department
Birmingham, AL, USA 35294

We quote the abstract:

"The alpha-beta tree search problem is described in mathematical terms to determine the feasibility of searching these trees in parallel. Three algorithms are developed to search depth-first game trees in parallel using a shared-memory multiprocessing computer system. Test results for uniform, non-uniform and alpha-beta depth-first game trees are provided.

The principal-variation splitting algorithm (PVS) along with an enhanced version (EPVS) are described. After discussing the performance of these algorithms, a new parallel tree-search algorithm, dynamic tree splitting (DTS) is developed and applied to the same trees. The DTS algorithm provides superior performance on the three types of trees discussed above, primarily because DTS does not require that all processors work together on descendants of the same node. This reduces synchronization overhead where a processor is out of work and has to wait on other processors to complete their work before moving to some other