

CHUNKING AS AN ABSTRACTION MECHANISM

by Murray S. Campbell

Ph. D. thesis as report CMU-CS-88-116
February 22, 1988 (115 pages)
Carnegie-Mellon University
Pittsburgh
USA 15213

Abstraction is the act of selectively ignoring detail. Hierarchical planners use abstraction to focus on the important considerations of a planning task, sketching a plan in an abstraction space and successively refining this plan to account for the details that had been set aside.

One source of abstraction is a predefined type hierarchy, with the most abstract operators and objects at the highest level, and actual domain entities at the lowest levels. In this approach a plan is created using the most general operators and objects, and refined by moving down the abstraction hierarchy, specializing as directed by domain-specific constraints.

However, in domains where there is significant pre-existing structure in the problem state, a strictly top-down approach to abstraction is not always appropriate.

This thesis presents an alternative, bottom-up, method of abstracting called *chunking* that groups together closely related domain objects and characterizes their aggregate behavior. Chunking defines a class of abstract objects, *chunks*, by ignoring the details of their component parts, concentrating instead on the behavior of the whole.

Plans can be formulated in the chunk-level abstraction space, and refined to account for the aggregate nature of the chunks. This has been illustrated by the development of a program, called CHUNKER, that solves King and Pawn endings in chess. CHUNKER has proven enough to identify numerous errors in the standard chess endgame textbooks, and solves problems beyond the capabilities of state-of-the-art brute-force chess programs.

(A limited number of single copies of this thesis are available directly from the author.)



Photo by Roy Keeley Jr.

Colossus IV versus Fidelity Excellence Mach III.
Mobile, April, 1988.