The Essence of Parallel Algol

Stephen Brookes

Department of Computer Science, Carnegie Mellon University, Schenley Park, Pittsburgh, Pennsylvania 15213

Received March 1997; revised June 1998

We consider a parallel Algol-like language, combining procedures with shared-variable parallelism. Procedures permit encapsulation of common parallel programming idioms. Local variables provide a way to restrict interference between parallel commands. The combination of local variables, procedures, and parallelism supports a form of concurrent object-oriented programming. We provide a denotational semantics for this language, simultaneously adapting possible worlds to the parallel setting and generalizing transition traces to the procedural setting. This semantics supports reasoning about safety and liveness properties of parallel programs, and validates a number of natural laws of program equivalence based on noninterference properties of local variables. The semantics also validates familiar laws of functional programming. We also provide a relationally parallel semantics. This semantics supports standard methods of reasoning about representational independence, adapted to shared-variable programs. The clean design of the programming language and its semantics shows that procedures and shared-variable parallelism can be combined smoothly.