Hayes, Ian J.; Meinicke, Larissa A.; Winter, Kirsten; Colvin, Robert J.
A synchronous program algebra: a basis for reasoning about shared-memory and event-based concurrency. (English) Zbl 1425.68293

Summary: In this paper we introduce an abstract algebra for reasoning about concurrent programs, that includes an abstract algebra of atomic steps, with sub-algebras of program and environment steps, and an abstract synchronisation operator. We show how the abstract synchronisation operator can be instantiated as a synchronous parallel operator with interpretations in rely-guarantee concurrency for shared-memory systems, and in process algebras CCS and CSP. It is also instantiated as a weak conjunction operator, an operator that is useful for the specification of rely and guarantee conditions in rely/guarantee concurrency. The main differences between the parallel and weak conjunction instantiations of the synchronisation operator are how they combine individual atomic steps. Lemmas common to these different instantiations are proved once using the axiomatisation of the abstract synchronous operator. Lifting these concepts from rely/guarantee concurrency to a higher level of abstraction makes them more widely applicable. We demonstrate the practicality of the algebra by showing how a core law from rely-guarantee theory, the parallel introduction law, can be abstracted and verified easily in the algebra. In addition to proving fundamental properties for reasoning about concurrent shared-variable programs, the algebra is instantiated to prove abstract process synchronisation properties familiar from the process algebras CCS and CSP. The algebra has been encoded in Isabelle/HOL to provide a basis for tool support for concurrent program verification based on the rely/guarantee technique. It facilitates simpler, more general, proofs that allow a higher level of automation than what is possible in low-level, model-specific interpretations.

MSC:
68Q85 Models and methods for concurrent and distributed computing (process algebras, bisimulation, transition nets, etc.)
68N30 Mathematical aspects of software engineering (specification, verification, metrics, requirements, etc.)
68T15 Theorem proving (deduction, resolution, etc.) (MSC2010)

Keywords:
refinement calculus; program algebra; rely/guarantee; concurrency; process algebra; synchronous parallel

Software:
Isabelle/HOL; Circus

Full Text: DOI arXiv

References:


This reference list is based on information provided by the publisher or from digital mathematics libraries. Its items are heuristically matched to zbMATH identifiers and may contain data conversion errors. It attempts to reflect the references listed in the original paper as accurately as possible without claiming the completeness or perfect precision of the matching.