Zakowski, Yannick; Cachera, David; Demange, Delphine; Petri, Gustavo; Pichardie, David; Jagannathan, Suresh; Vitek, Jan

Verifying a concurrent garbage collector with a rely-guarantee methodology. (English)

Zbl 1468.68068

Summary: Concurrent garbage collection algorithms are a challenge for program verification. In this paper, we address this problem by proposing a mechanized proof methodology based on the Rely-Guarantee proof technique. We design a compiler intermediate representation with strong type guarantees, dedicated support for abstract concurrent data structures, and high-level iterators on runtime internals. In addition, we define an Rely-Guarantee program logic supporting an incremental proof methodology where annotations and invariants can be progressively enriched. We formalize the intermediate representation, the proof system, and prove the soundness of the methodology in the Coq proof assistant. Equipped with this, we prove a fully concurrent garbage collector where mutators never have to wait for the collector.

MSC:

68N20 Theory of compilers and interpreters
68N30 Mathematical aspects of software engineering (specification, verification, metrics, requirements, etc.)
68Q60 Specification and verification (program logics, model checking, etc.)
68V15 Theorem proving (automated and interactive theorem provers, deduction, resolution, etc.)

Keywords:
verified garbage collection; concurrency; Coq; rely-guarantee

Software:
z3; Jitawa; Boogie; GCminor; Coq; Milawa; CompCertTSO

Full Text: DOI HAL

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.