Bounded Transaction Model Checking

Xiaofang Chen and Ganesh Gopalakrishnan

UUCP-06-003

School of Computing
University of Utah
Salt Lake City, UT 84112 USA

February 27, 2006

Abstract

Industrial cache coherence protocol models often have too many reachable states, preventing full reachability analysis even for small model instances (number of processors, addresses, etc.). Several partial search debugging methods are, therefore, employed, including lossy state compression using hash compaction, and bounded model checking (BMC, or equivalently, depth-bounded search). We show that instead of a BMC approach, a *bounded transaction* approach is much more effective for debugging. This is because of the fact that the basic unit of activity in a cache coherence protocol is that of a *transaction* - e.g., a complete causal cycle of actions beginning with a node making a request for a line and obtaining the line. The reduced effectiveness of BMC mainly stems from the fact that by limiting only the search depth, it cannot be guaranteed that complete transactions get selected, or that the right kind maximal number of interacting transactions.

Thus, instead of bounded model-checking, which explores all possible interleavings in BFS, we propose a bounded *transaction* model-checking approach for debugging cache coherence protocols, where the criterion is to allow a certain *number of transactions* chosen from a set of potentially interfering set of transactions, to be explored. We have built a bounded transaction version for the Murphi model checker and shown that it can find seeded bugs in protocols far more effectively, especially when full BFS runs out of memory and misses these bugs. We compare our work with similar ideas - such as debugging communicating push-down systems[1] by bounding the number of interleavings (a similar idea, but different in detail).