要 旨

メモリモデルを考慮したメモリアクセスを提供するSPIN用ライブラリ

松元　稿如

プログラムを運用する際は、事前にプログラムの動作が正しいことを検証することが必須である。現代の計算機上で動作するプログラムでは，複数のスレッドを並行に動作させて処理を行う場合がある．そのような場合においても，プログラムが正しく動作することを検証しなければならない．しかし，複数のスレッドを並行に動作させて処理を行う場合は，単一のスレッドで処理を行う場合よりも検証するのが難しい．複数のスレッドを並行に動作させて処理を行うプログラムの検証には，例えば各スレッドの命令が入り混じった順序で実行されるという難しさがある．そこで，モデル検査法を使用した自動検証ツール SPIN を用いた検証が行われている．モデル検査法とは，検査対象から状態の変化に関する動作，例えばプログラムならば変数の操作などを抜き出したモデルを作り，それが期待する性質を満たすかどうかを検査する手法である．各スレッドの命令が入り混じって実行される以外にも，あるスレッドの実行したメモリにアクセスする命令 (メモリアクセス命令) の列の実行順序が，他のスレッドからは入れ替わって見える可能性があるという難しさもある．他のスレッドからの見え方はプロセッサ毎に異なっており，そのパターンをメモリコンシステンシモデル（メモリモデル）と呼ぶ．複数のスレッドを並行に動作させて処理を行うプログラムのモデル検査では，メモリアクセス命令の実行順序が入れ替わって見えることを考慮する，つまり，メモリモデルを考慮することも必須となる．しかし，SPIN はメモリモデルを考慮していない．

そこで本研究では，メモリモデルを考慮したメモリアクセス命令を提供するライブラリを開発した．提案するライブラリは，SPIN の入力となるモデルを記述するための Promela

という言語で書かれている．また，メモリモデル毎に別のファイルに記述されており，モデル中に読み込むことによって，それぞれのメモリモデルを考慮したメモリアクセス命令を使用できるようになる．本研究では，シーケンシャルコンシステンシ，トータルストアオーダリング，パーシャルストアオーダリングの三つのメモリモデルのライブラリを開発した．どのメモリモデルでも共通のインターフェースで利用できるように設計したため，検査対象のPromela コードを書き換えることなく異なるメモリモデルの下での検査を行える．なお，一般的にモデル検査には，モデルが大きくなるにつれて状態数が爆発的に大きくなるという欠点があるが，本研究では，検査対象のモデルから情報を受け取り，ライブラリのサイズを検査対象にとって必要最低限に抑えられるように設計することで対処した．

提案するライブラリを実装し実験を行ったところ，ライブラリがメモリモデルを考慮した正しい動作をするメモリアクセス命令を提供できることを確認した．

キーワード　モデル検査，メモリコンシステンシモデル，SPIN

Abstract

A Memory Access Library under Relaxed Memory Models for SPIN

Kosuke MATSUMOTO

It is desirable to verify correctness of programs when we use the programs. It is difficult, however, to verify multi-thread programs. One of the reason is that executions of threads are interleaved with each other. Model checking is used for multi-thread programs. Model checking checks whether a model, which is an abstract program that reflect behavior of the verification target program that we concern about, satisfies an expected property. Another reason is that the order of memory access instructions executed by a thread may appear different from the order written on the program to other threads on modern multiprocessors. The pattern of how to be seen the order of memory access instructions executed from other threads is called a memory consistency model (memory model). It is required to consider the memory model for verifying programs that work on a modern multiprocessor. However, the SPIN model checker does not consider the memory model.

This research developed a library that provide memory access instructions that enable memory model conscious model checking. This library is written in Promela that is a language to describe a model for SPIN. Also, this library is described in a different file for each memory model. By including the file of a memory model into the user's model, the user can use memory access instructions that can be reordered according to the memory model. This research developed a library of three memory models,

sequential consistency, total store ordering and partial store ordering. All libraries were designed to have a common interface, so that we can switch memory models without modifying the model of the program. Generally, model checking has a difficulty that the amount of computation becomes larger explosively as the model becomes larger. Therefore, we designed a library, so that we can adjust the size of models of memory access instructions by using information of the user's model.

We experimented with a library. As a result, we found that the library can provide the correct memory access instructions according to the memory model.

key words model checking, memory consistency model, SPIN