要 旨

検証結果に影響を与えない到達可能性
検証用コード削減手法

大崎　洋平

近年の情報技術の発展により，社会や日常生活のいたるところでソフトウェアが用いられている．ソフトウェアの役割が大きくなるにつれて，ソフトウェアの信頼性をどのように確保するのかという問題が重要となる．

システムに不具合がないか数学的に証明する手法のひとつとしてモデル検査がある．モデル検査では，システムの状態遷移モデルに対して網羅的な検査を行うことで，システムの振る舞いが正しいかどうか確かめる．モデル検査をプログラムのソースコードの検証に適用することで，仕様に対するソースコードの誤りを発見できる．モデル検査をソースコードに適用するためには，ソースコードを状態遷移モデルを表す検証用コードに変換する必要がある．しかし，ソースコードの全ての情報を検証用コードで表すと状態数や遷移数が大きくなり，実用的な時間でモデル検査を行えない．

そこで本研究では，ソースコード中のプログラム文への到達可能性を検証するための情報をソースコードに追加し，静的スライシングを用いて検証したい振る舞いを保持しつつ最低限のステップ数を持った検証用コードを作成する手法を提案する．

キーワード　モデル検査，静的スライシング

Abstract

A Code Reduction Method for Reachability Verification without Altering the Verification Result

Yohei OSAKI

Software is used in everywhere of the society and daily life as a result of the development of the information technology in recent years. As the role of software becomes important, the problem how to guarantee the reliability of software becomes crucial.

Model checking is one of the techniques for mathematically proving whether a system has a defect. In model checking, it is confirmed whether the behavior of the system is correct by performing exhaustive inspection of the state transition model of the system.

A difference between source code and its specification can be discovered by applying the model checking to the verification of the source code. To do so, the source code should be converted into the code for the verification that specifies a state transition model. However,the state transition model way have a lot of numbers of states the model cannot be inspected in practical time when all information on the source code is shown by the code for the verification.

In this thesis, we propose method for generating code for verification with the minimum steps without altering the verification result. This method adds information for verification of reachability to a program statement to source code and reduces code for verification using static slicing.

key words Model Checking, Static Slicing