JAIST Repository
https://dspace.jaist.ac.jp/

Title	モデル検査を用いたシステム検証の支援に関する研究
Author(s)	中野, 昌弘
Citation	
Issue Date	2003-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1680
Rights	
Description	Supervisor:二木 厚吉, 情報科学研究科, 修士

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

モデル検査を用いたシステム検証の支援に関する研究

中野 昌弘 (110093)

北陸先端科学技術大学院大学 情報科学研究科

2003 年 2 月 14 日

キーワード: 形式仕様, 証明論的手法, モデル検査, 振舞仕様, 仕様変換.

1 研究の背景と目的

形式的手法により，ハードウェア・ソフトウェア等のシステムを始め，様々なシステムの正しさを検証する取り組みが活発に行われている．ある性質を満たすシステムを設計・実装するために，仕様・性質を記述し，仕様の上で性質が成り立っているか証明することを考える．仕様の検証法は大きく分けて 2 種類ある．

証明論的手法 は，十分な仕様の記述能力と検証能力を有す検証法である．検証作業は，人が補題の発見など証明の指針を与えることで検証を進めていく．検証成功は性質の成立を意味するが，検証の失敗から直ちに性質の不成立が導かれるわけではない．検証の失敗は仕様・性質・検証法のいずれかに問題があるため，修正して再度検証を試みる必要がある．しかし検証失敗の原因が仕様 · 性質にあるか検証法にあるのか判断することは非常に困難である．

無限状態遷移機械が記述可能な仕様として，振舞仕様がある．振舞仕様では状態は隠蔽され明示的に記述することはできない．振舞仕様は観測演算により現在の状態の一部の情報を観測し，ある状態を記述した項を得る．操作演算により現在の状態を変更する．この時の遷移関係は，操作演算を適用する前後の観測結果の変化を定義することにより与える．初期値は初期状態に対する観測結果を定義することにより与える．

モデル検査的手法 は，対象を有限状態機械に限定した検証法である．検証作業は完全に自動で行うことができる．検証の成功は性質の成立を意味し，失敗は性質の不成立を意味する．失敗時には反例を挙げることができ，仕様・性質を修正するための重要な手がかりとなる．

モデル検査の一つとして SMV がある．SMV は状態を有限の値をとる変数の組で表し，状態の変更は next 演算により全ての変数の値を一斉に更新することで行われ

Copyright © 2003 by Nakano Masahiro

る．状態の遷移規則は各変数に対し next 演算を行う前後の値の変化を定義することにより与える．初期状態は init 演算を定義することにより与える．

両検証法は大きく異なり，統合することでより有用な検証法となることが予想される．その取り組みの一つとして本研究では，振舞仕様から SMV 仕様を生成する変換器を実装する．モデル検査により仕様に対するテストを行い，証明論的手法による検証によって仕様の正しさを証明する．モデル検査によって反例が自動で発見されることにより，仕様・性質を早期に修正することが可能となり，検証を支援する．

2 本研究のアプローチ

本研究では，振舞仕様から SMV 仕様へ変換を行う．変換の制約として次の条件を満たすこととする．

- 振舞仕様での状態の等価性が，各観測演算の観測結果の等価性によって与えられる．
- 振舞仕様の抽象データ型をブールと自然数に限定．

振舞仕様では，観測演算によって観測できない情報というものが存在しうる．最初の仮定により，全ての情報が観測演算で観測できることが保障される．これにより各観測演算を SMV の状態変数に変換することで，振舞仕様の隠蔽された状態を状態変数の組で表すことができるようになる．次の仮定により，変換処理を定義できるようにする．任意の抽象データ型上の項を変換することは容易ではないため，まずこれらのデータ型に対して変換を行えるようにする．これらの制約を与えても広範な問題を取り扱うことができ，モデル検査により検証を支援することができる．

変換処理は，状態・next 演算・init 演算を振舞仕様から定義することにより行われる．状態は，各観測演算に対し観測演算が返すデータ型に対応した状態変数を定義することで与える．各状態変数に対し next 演算を定義するために，まず遷移関係を記述した等式を観測演算ごとに分類する．分類ごとに各等式から等式の適用条件・適用後の値を取り出し，それらを列挙することで next 演算を定義する．遷移規則の条件式は，等式の条件式・適用する操作演算の種類などの情報から決定する．init 演算は，初期値を定義した等式があればそれにより定義し，なければ初期値として任意の値をとることができるように定義する．

変換処理において本質的な問題点は，振舞仕様が無限状態遷移機械を扱うのに対し，SMV は有限状態遷移機械を扱うことである．このため，変数の変換処理で対応する型を定義できなかったり，無限個の変数を要求されてしまう．本研究では無限状態遷移機械から有限状態遷移機械を得るため，制限による方法を用いる．制限は状態遷移機械の一部を取り出す処理である．仕様に対し容易に適用できるため，直ちにモデル検査を行うことができる．同様の技術として抽象化があるが，抽象化法を得ることは容易ではなくモデル検査を

適用することが遅れる．ただし，制限による方法は検査漏れが生じるため仕様の正しさを示すことができなくなる．

3　研究の成果

本研究では，状態遷移機械を記述する振舞仕様に対して，証明論的手法に基づく CafeOBJ とモデル検査的手法に基づく SMV を併用した検証法を提案した．振舞仕様に対するモデル検査を行うため仕様変換器 C-TRAS を実装した．

振舞仕様に対するモデル検査は次のように行われる．まずはじめに，C-TRAS より得られた SMV 仕様に対しモデル検査を行い反例を探す．反例が見つかれば，その原因を考察・修正し，再度モデル検査を行う．見つからない場合は，２通り考えられる．

(1) 制限によって切り捨てた状態遷移機械に反例が含まれている．

(2) 振舞仕様に誤りが存在しないため，モデル検査で反例が得られない．

仕様に誤りがないことを示すために，CafeOBJ により検証を行う必要がある．本研究では，SMV により得られた反例が元の振舞仕様でも反例であることを証明した．

実際に Load と Store に基づく相互排他システム，Test & Set に基づく相互排他システム，Peterson のアルゴリズムに対して実験を行った．振舞仕様をある範囲に制限し，モデル検査を行った．いくつかのモデルにおいて反例が報告されたので，その反例が元の振舞仕様においても反例である事を CafeOBJ によって確認した．証明論的手法による証明ではこれらの反例は自動的に得られないため，振舞仕様に対するモデル検査の有効性を確認した．また Load と Store に基づく相互排他システムでは，SMV によって報告される反例よりその原因を考察し，仕様を修正した．修正した仕様に対しては SMV により反例が報告されなくなり，明らかな誤りを修正することができた．本研究は，振舞仕様に対し SMV により反例を探すことで CafeOBJ による検証を支援する．誤りが存在することだけでなく反例を示すことができるため，誤りの原因を考察を助け，仕様の修正を容易にする．

制限を適用した仕様に対するモデル検査は，元の仕様の誤りを発見しない可能性があることから検証と呼ぶことはできない．仕様が正しいことを検査するためのテスト技法の一つであり，仕様変換器は仕様に対するテスト生成器ととらえることができる．通常のテスト技法と比べ，本手法は有限・無限の遷移系列に対するテストが行え，またある状態空間に対し，網羅的なテストを行えるという特徴を持つ．通常のテストは，無限の遷移系列のテストケースを記述不能であり，テストケースの網羅性も保証されない．本手法によるテストは，通常のテストと比べこれらの点でより強力であり，本研究は仕様に対するテストの信頼性向上においても貢献した．