5A-2

モデル検査自動化ツールの開発
～検査自動化と反例解析効率化～

高田　沙都子†　　森　奈実子†　　村田　由香里†
株式会社　東芝　ソフトウェア技術センター†

1. はじめに

近年，ソフトウェアの複雑化や大規模化とともに，開発の上流工程で混入した誤りが下流工程で発見されることによる修正コストの増加が問題視されている．また，ソフトウェアに対する信頼性・安全性の保証はますます重要になってきており，上流工程での品質向上は必須の課題となっている．

モデル検査は，システムの振舞いについて全ての状態の網羅的な自動検査が可能な技術であり，再現の難しい発見困難な誤りの検出や，上流工程で誤りを早期発見することによるコスト削減の効果が期待されている．しかし，実際の開発現場でモデル検査を適用するには，課題も存在する．課題としては，検査実施に作業コストがかかる点や仕様をモデル化する際に専門的知識や経験を要する点が挙げられる．そこで，システムの振舞いを表す状態遷移表とシステムが満たすべき仕様（制約条件）から自動的にモデル検査を実施する機能と，検査結果の解析を補助する機能を備えることで専門性や作業コストを削減するツールを開発した．本稿では開発ツールの機能および評価結果について述べる．

2. モデル検査の概要

モデル検査適用時の作業フローを図 1 に示す．今回は，モデル検査器として SPIN[1]を採用した．

モデル検査の入力となるのはシステムの振舞いを表す動作仕様と制約条件の二つである．開発者は検査を行うために，動作仕様から，SPIN 独自の記述言語 Promela を用いて検査用モデルを作成し，制約条件から線形時相論理(LTL)を用いて検査式を記述する必要がある． SPIN は，検査用モデルと検査式から全ての実行パスを自動的に網羅し，検査式に違反する実行系列（反例）が存在しないかを検査する．反例が検出された場合，開発者は反例を解析し仕様を修正する．

Development of Automated Model Checking Tool : Automated Model Checking & Efficiency of The Counterexample Analysis
Satoko Takada†, Namiko Mori†, Yukari Murata†
†TOSHIBA Corporation Software Engineering Center

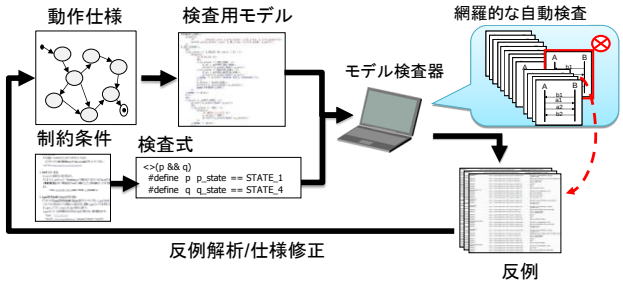

図 1　モデル検査の作業フロー

3. モデル検査自動化ツール

これまで検査用モデルや検査式の記述は手作業で行ってきたが，スキルや時間を要するうえ，反例が可読性の低いテキストログで表示されるため，解析作業も容易ではなかった．これらの課題を解決するため，検査用モデルおよび検査式の自動生成機能，モデル検査実行機能，反例解析を補助する機能を持つことで，モデル検査の適用効率を向上させるモデル検査自動化ツール(図 2)を開発した．

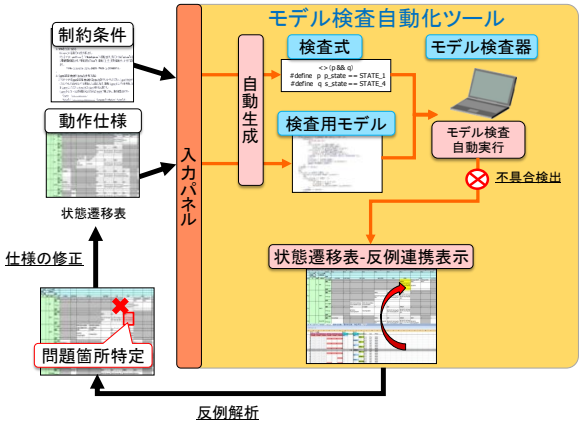

図 2　モデル検査自動化ツール

3.1. 検査用モデル生成機能

検査用モデル生成機能ではシステムの動作仕様から SPIN 独自の記述言語 Promela で記述された検査用モデルを自動生成する．入力とする仕様は，状態遷移表を採用した．

本ツールは変数，アクション，ガード条件を定義する表を備えている．これらの定義表と状態遷移表を元に図 3 のように Promela の検査用モデルが生成される．

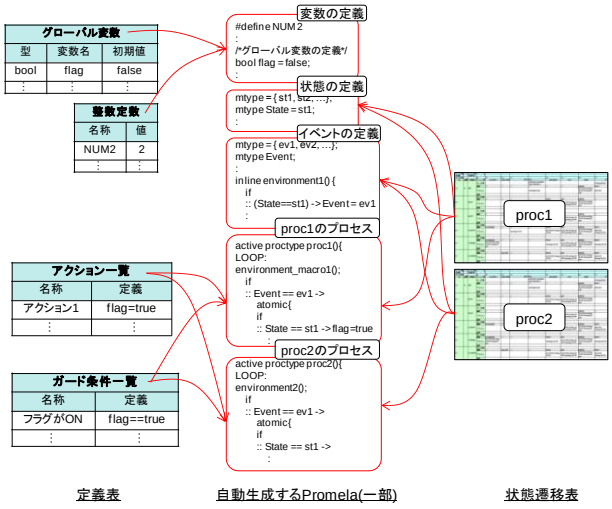

図 3 Promela の生成

3.2. 検査式生成とモデル検査実行機能

検査式生成機能は制約条件を図 4 のようにルール化された日本語で入力することにより，LTL で記述された検査式を自動生成する．

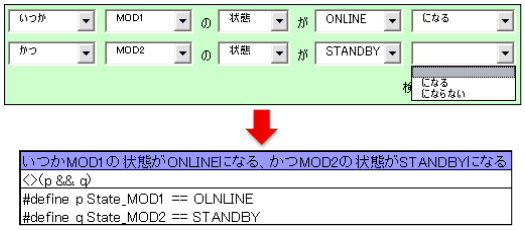

図 4 制約条件入力と検査式生成

モデル検査実施機能はバックグラウンドで SPIN を実行する機能である．ユーザが操作パネル上の検査実行ボタンを押下すると自動化ツールが SPIN を起動し，検査を実行する．

3.3. 状態遷移表-反例連携表示機能

状態遷移表-反例連携表示機能は可読性を向上させた反例と状態遷移表を連携して表示させる機能である．

SPIN が出力したテキスト形式の反例から変数，状態，イベントに代入された値や，実行ステップに対応する状態遷移表のマスの ID といった解析に必要な情報のみを抽出する．抽出した情報を図 5 のようにチャート形式に変換して表示することで可読性を向上させる．

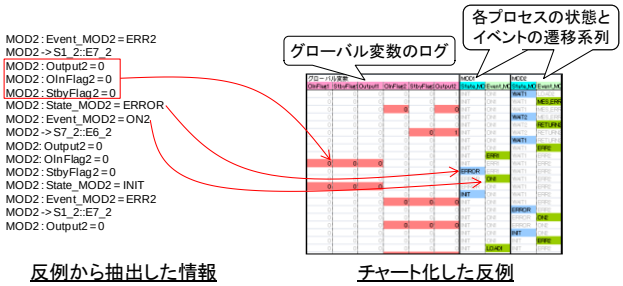

図 5 反例のチャート化

さらに，反例上のあるステップを選択すると，図 6 のように，選択したステップに対応する状態遷移表上のマスを強調表示する．この機能を用いることでユーザは反例と動作仕様を比較しながら反例の解析を進めることができる．

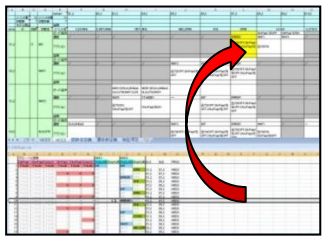

図 6 状態遷移表-反例連携表示

4. 評価方法および結果

モデル検査適用未経験者２名に，手作業によるモデル検査の実施とモデル検査自動化ツールを用いたモデル検査の実施を行いモデル検査の各工程で要した工数を計測させた．計測した工数を比較したところ，表 1 のようになった．

表 1 工数比較結果

モデル検査工程	ツール未使用	ツール使用
検査用モデル作成	4.5h	1.5h
検査式作成	3.2h	1.5h
モデル検査実行	0h	0h
反例解析	3.8h	1.5h
計	12.5h	4.5h

5. 考察

結果から検査用モデル作成，検査式作成では，これまで手作業で行っていたものを自動化したことで工数が削減できていることがわかる．反例解析についても，可読性向上の効果により原因を突き止めることが容易になった．適用工程全体で約 60%削減できたことから，モデル検査自動化ツールにより適用効率は大きく向上したと考えられる．

6. おわりに

本稿では，モデル検査適用の効率向上および簡易化を目標にモデル検査の適用作業を一部自動化，サポートを行うツールを開発した．モデル検査未経験者によりツール有無それぞれの適用工数を比較したところ，適用効率が向上していることを確認した．

参考文献

[1] SPIN Model Checker,
http://SPINroot.com/SPIN/whatiSPIN.html