

## 4L-06 モデル検査ツール SPIN による並行プログラムのデバッグを支援する状態遷移追跡支援ツールの設計と実装

山川真延† 川端英之‡ 弘中哲夫‡

広島市立大学情報科学部情報工学科† 広島市立大学大学院情報科学研究科‡

## 1 はじめに

モデル検査ツール SPIN を用いたモデル検査は並行プログラムのデバッグに有用である。一方、SPIN ではランダムシミュレーションも実行可能で、システムのモデル化の段階で挙動を大雑把に把握したいという場面や、開発者が予期していない箇所での仕様の不完全さやバグを発見するのに有用である。これら両者の総合により効率的なプログラム開発が期待できる。しかしながら現状では、モデル検査とランダムシミュレーションの結果を融合するしくみは SPIN にはない。

例えば、ランダムシミュレーションの検査対象である、モデルからランダムに得られる遷移列が全状態中でどこに位置するかを即座に確認できるツールがあると有用である。また、SPIN を核とする統合開発環境である jSpin にモデル検査で得られる状態遷移図を可視化する機能はあるが、ランダムシミュレーションで得られる状態遷移図の可視化機能はない。

本研究では、ランダムシミュレーションによるデバッグに着目し、ランダムに得られる遷移列の同定や、可視化をするツールの開発を試みる。本稿では以上で述べたようなツールの設計及び、実装について述べる。

## 2 モデル検査ツール SPIN と状態遷移図

## 2.1 モデル検査ツール SPIN と jSpin

SPIN はモデル検査を自動で行うためのツールの一つである。SPIN では並行動作をするシステムを、チャンネル通信を用いてモデル化する。モデルは仕様記述言語 PROMELA により記述される。

SPIN のための GUI 環境の一つに jSpin がある。jSpin において状態遷移図を出力するシステムは SPINSPIDER である。

SPINSPIDER は SPIN の実行結果から状態遷移図を抽出して描画する。SPINSPIDER によって描かれる状態遷移図は、モデル中で到達可能な全状態を網羅したものである。

Design and implementation of a Tool for State Transition Tracking for Debugging Concurrent Programs with the SPIN Model Checker

Masanobu Yamakawa† Hideyuki Kawabata‡ Tetsuo Hironaka‡

†Department of Computer and Network Engineering, Hiroshima City University

‡Graduate School of Information Sciences, Hiroshima City University

## 2.2 SPIN における状態遷移の把握

SPIN では状態を各プロセスのプログラムカウンタと内部変数の値の組み合わせにより表現する。

SPIN により抽出される状態遷移の情報は、ログファイル経由で得ることができる。例えばランダムシミュレーションを行なった結果得られる状態遷移情報は図 1 のようになっている。

タイミグスタンプ	プロセス名	状態番号	モデル中の行	プログラムカウンタ
57:	proc 0 (Source:1)	jikken.pml:9 (state 9)	-end-	
59:	proc 0 (Source:1)	jikken.pml:5 (state 4)	else	
60:	proc 2 (Destination:1)	jikken.pml:29 (state 3)	output = 0	
62:	proc 0 (Source:1)	jikken.pml:6 (state 5)	((input==0))	
63:	proc 0 (Source:1)	jikken.pml:7 (state 6)	input = 1	
64:	proc 1 (Relay:1)	jikken.pml:14 (state 1)	((input!=0))	
65:	proc 1 (Relay:1)	jikken.pml:15 (state 2)	((output==0))	
66:	proc 1 (Relay:1)	jikken.pml:17 (state 3)	(1)	

図 1: ログファイルの例 (一部切り取り)

## 3 状態遷移追跡ツールの設計

## 3.1 システム概観

本ツールはログファイルの情報を可視化し、即座に確認することができるようにする。また、jSpin から静的に得られるモデル検査の結果を用いた状態遷移図と融合させる。これにより、ランダムシミュレーションで検査した状態列が全状態中のどこに位置するかが判断できるようになる。

## 3.2 システム構成

本ツールは図 2 に示すようにログファイル解析、状態遷移図抽出、ログファイル編集、状態遷移図マージという基本単位から構成されている。

ログファイル解析では、ログファイルを図 1 に示したように切り分け、データセットを作成する。そして状態遷移図抽出で、そのデータセットを用いてシーケンス図と状態遷移図を抽出する。

ログファイルには SPIN がモデル検査時の状態定義に使用しないプログラムカウンタが存在する。そのためそのままでは二つの状態遷移図をマージできない。ログファイル編集では、ログファイルに存在するプログラムカウンタを SPIN が使用するプログラムカウンタに即したものに編集する。その後、二つの状態遷移図をマージする。

## 3.3 シーケンス図抽出

図 1 のログファイルから描かれたシーケンス図の一部を図 4 に示す。本図ではランダムシミュレーションの処

