段階的検査法を用いたモデル検査の反例分析手法

大森　祐貴　小飼 敬　上田 賀一　山形 知行　武澤 隆之

モデル検査は検査対象の振舞いモデルを網羅的に探索することで，仕様の誤りを検査する手法である．しかし，実用上のモデル検査では，状態爆発が問題となる．状態爆発を回避するために，先行研究でモデルを分割して検証を行う段階的検査法を考案した．一方，モデル検査で誤り (反例) が見つかった場合，原因を追求するため，反例を分析することが必要である．段階的検査法では検査対象を分割するので，従来の反例分析とは異なる手法が必要となる．本研究では，分割されたモデル特有の反例から，検査対象全体のモデルの反例を判別し，分析する手法を提案する．

Model checking is a method for inspecting an error of specifications,by exhaustively explore the behavior-model of the target.But in practical model checking,state explosion is a problem.We devised previous study "A Stepwise Model Checking Approach" to avoid the state explosion.On the other hand,if any error (counterexample) is found in model checking,it is necessary to analyze to pursue the cause.In a stepwise model checking approach,since splitting the inspected it requires a different approach from traditional counterexample analysis.This study propose an analysis method to determine the counterexample of the entire model from the split model specific counterexamples.

1　はじめに

社会インフラに関わる大規模システムで障害が起こると，社会に大きな悪影響を与えてしまう．そのため大規模システムの品質を高めることは極めて重要である．大規模システムの品質を保証する技術の一つにモデル検査がある．モデル検査は検査対象システムの設計書からモデルを作成し，モデルの取りうる状態を網羅的に探査することでシステム内の不具合 (反例) を発見する．反例をもとに設計書の修正を行い，反例が出力されなくなったとき，システムは正常に動作することが保証される．しかし，実務で利用すると検査対象の規模の大きさから，状態爆発が起きてしまい検証を終了することができないことがある [8]．本研究の先行研究では状態爆発を回避する手段として段階的検査法を提案した [3]．この手法は検査対象システムを分割して検証を行うことで，一度に扱う状態数を少なくする手法である．一般的なモデル検査の修正作業とは異なった操作が必要である．そこで本研究では段階的検査手法を利用したときの反例分析手法の提案を行う．

An Analysis Support of Counterexamples in A Stepwise Model Checking Approach for Verification
Yuki Omori, Yoshikazu Ueda, 茨城大学, Ibaraki University.
Kei Kogai, 茨城工業高等専門学校, Ibaraki National College of Technology.
Tomoyuki Yamagata,Takayuki Takezawa, 株式会社 日立製作所, Hitachi Co.,Ltd..

2　関連知識

2.1　情報制御システムの概要

本研究が対象とする情報制御システムは，作業員が手作業で行っていた設備の制御を自動で行うため，作業員のノウハウをルールとして体系化し，そのルールをもとに設備を自動で制御するシステムである．情報制御システムの構成を図 1 に示す．制御プログラムは制御ルール，仮想設備，仮想センサから構成される．また，物理環境は設備，センサ，制御対象から構成されている．制御プログラムはセンサから制御対象の情報を取得し．取得した値をもとに仮想設備の状態

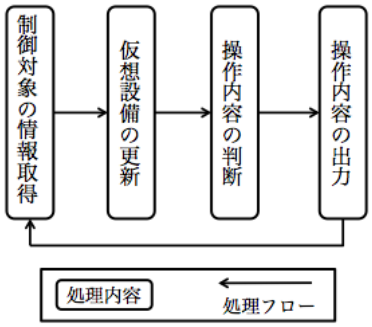

図 2　制御プログラムの動作

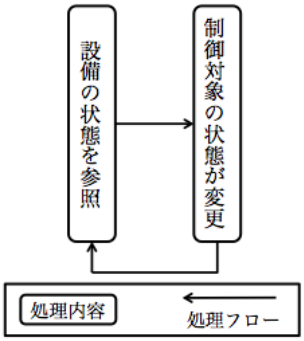

図 3　物理環境（制御対象）の動作

を更新して，制御ルールをもとに操作内容を判断し，物理環境に対して操作指示を出力する．物理環境は操作指示によって変化した状態をセンサで取得する．情報制御システムは，非同期的に連携し任意の周期で，仮想環境の値を更新し制御値の出力を繰り返すシステムといえる．

2.2　情報制御システムの振舞いモデル

情報制御システムに対してモデル検査を適用するために，振舞いモデルを作成する．制御プログラムを制御ルールモデル，物理環境を物理環境モデルとしてモデル化する．これら 2 つを合わせたモデルを振舞いモデルとする．また，本研究では制御対象の情報取得から操作内容を出力するまでを 1 実行サイクルとする．振舞いモデルはシステムの性質を表す属性を持っていて，その属性の組合せを状態と呼ぶ．制御ルールによって状態が変わることを状態遷移と呼ぶ．また，制御ルール適用前の状態を遷移前状態，制御ルール適用後の状態を遷移後状態と呼ぶ．

2.3　モデル検査ツール SPIN

モデル検査ツールは SPIN [4]，NuSMV [5]，UPPAAL [6] など様々なものが存在しており，それぞれが異なる特徴を持っている．本研究では SPIN を利用する．SPIN はオートマトンを基にしたモデル検査ツールである．SPIN を利用したモデル検査では，検査対象となるシステムの振舞いモデルと検査項目を状態遷移モデルに変換する [9]．その後，状態遷移モデル内に検査項目を満たさないパスが存在するかを網羅的に探索する．検査項目が満たされない場合，初期状態から検査項目に違反するまでのパスが反例として出力される．モデル検査者はこの反例をもとにモデルの修正を行う．

2.4　段階的検査手法

段階的検査手法はモデル検査での状態爆発を防ぐための手法である [7]．この手法は一次処理と二次処理からなる．SPIN を利用したモデル検査では検査中に実行された振舞いは全て同等に扱われる．しかし，反例分析の際は制御ルールの適用に関する情報が必要である．そこで，反例分析の際に検査中に実行された制御ルールのみを容易に取得するため，あらかじめ振舞いモデルの制御ルールにマーキングを行い，制御ルールの適用に関する行が実行されたときに情報を出力する記述を，設計書から振舞いモデルに変換する際に加える．また，設計書の制御ルールと振舞いモデルに記述された制御ルールの対応に関する情報を記憶する．

2.4.1　一次処理

はじめに，設計モデルを物理分割性，構造対称性，振舞い分割性によって分割して振舞いモデルへの変換を行う．例えば図 4 のように対象となる情報制御システムを情報のやりとりを行うサブシステム A，B と分割する．これにより状態空間を分割し，探索する範囲を限定することで，一度に扱う状態空間を小さくし状態爆発を防ぐ．作成された振舞いモデルそれぞれに対して状態マネージャを利用してモデル検査を行う．状態マネージャは属性値の組合せを 1 つ選び，モデル

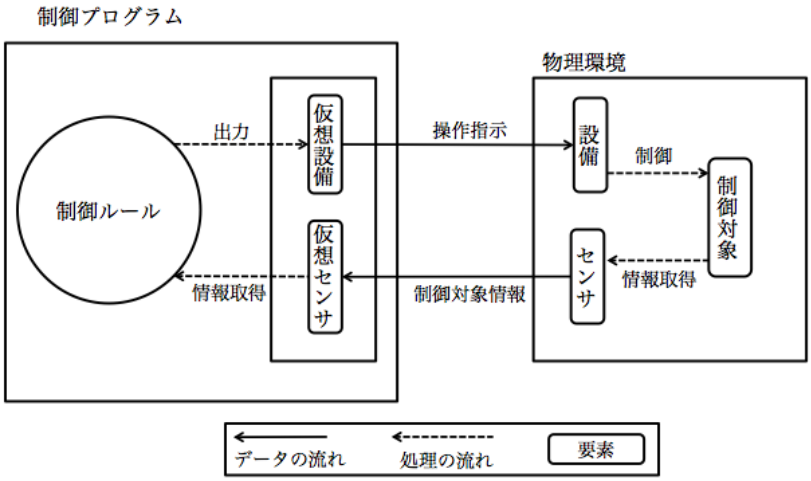

図 1　情報制御システムの構成

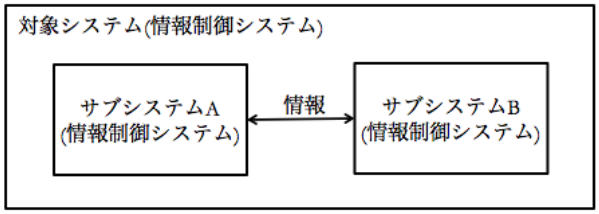

図 4　システムの分割

検査の初期状態として入力を行い，1 実行サイクル後の状態を部分遷移集合として出力する．部分遷移とは 1 実行サイクルにおいて，制御ルール適用前の仮想環境の状態と，制御ルール適用後の仮想環境の状態の組である．また，出力する際には二次処理で扱う状態遷移モデルを小さくするために，検査項目に関係ない属性を除外する処理 (着目属性アプローチ) を行う．図 4 のように分割を行った場合，図 5 のような状態遷移モデルを得る．1 つの振舞いモデルから得られた状態遷移の集合を部分遷移集合と呼ぶ．一次処理が終了した段階でシステムのデッドロックを発見することができる．1 実行サイクルの中でいずれの制御ルールも適用されなかった場合は，次の状態に進むことができないのでデッドロックに陥ったことになる．

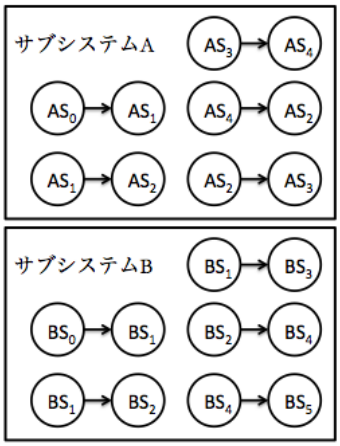

図 5　部分遷移集合の例

2.4.2　二次処理

一次処理によって得た部分遷移集合では，分割された領域間を横断する性質の検査ができない．そこで，それぞれの領域の部分遷移集合の中で同一の状態を表す状態の合成を行い，合成遷移モデルを作成する．作成した合成遷移集合に対し，検査項目と同等の検索条件を与え探索することで検証を行う．検査項目を満たさないパスが見つかった場合は反例として出力する．

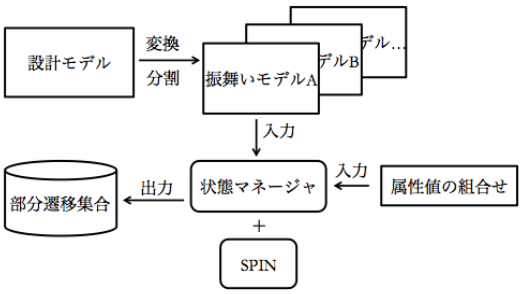

図 6　一次処理の概要図

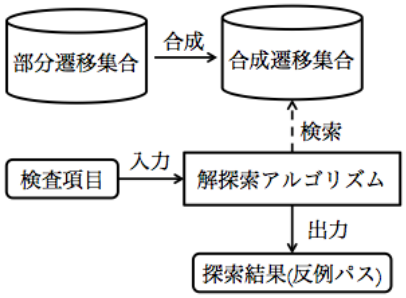

図 7　二次処理の概要図

3　段階的検査法を利用した反例分析

一次処理では状態マネージャを利用して遷移前の属性値と遷移後の属性値の組をもって部分遷移集合とした．その状態マネージャに，どの制御ルールの適用によって遷移が起こったかを同時に出力する記述を加える．これによって，合成遷移集合を検索することで，適用されたルールと状態遷移の関係を容易に調べることができるようになる．

3.1　反例パスで適用されたルールの抽出

二次処理で図 8 左の反例パスが見つかったとする．このとき，それぞれの状態間で適用されたルールは一意に決定していないことがある．図 8 右のように，反例パス上の状態が反例とならないような遷移を持っているとき，状態 ABCD のいずれかを通り反例とならないパスは，ABE，ABF，ABCG，ABCH の 4 つ存在する．反例となる遷移とルールの組を調べるために状態 A，B，C，D それぞれを初期状態としてモデル検査を行い部分遷移集合を作る．ここでは着目属性アプローチを利用せず，全ての属性値を出力させる．全ての属性値を見ることで，部分遷移集合を基に，初期状態から終端までのパスと適用されたルールの対応を見ることができる．このときに適用されるルールの例を表 1 に示す．このような表を作ることで反例パスは $R_1R_2R_3$ の順序でルールが適用され反例となったことがわかる．

3.2　反例分析

反例を導いた制御ルールが $R_1R_2R_3$ とわかったので，それらに対して分析を行う．

3.2.1　単体ルールが原因の場合

始めに，R_1, R_2, R_3 のそれぞれが反例のパス上以外でも適用されているかを調べる．反例のパス上のみで適用されている場合，ルール単体で反例の原因となっている可能性がある．

3.2.2　ルールの組合せが原因の場合

反例のパス上以外でも適用されている場合，組合せに原因がある可能性がある．組合せに原因があると考える場合は R_2R_3 や $R_1R_2R_3$ のように続けて適用されているかを調べる．反例のパス上のみでそれらの組合せで適用されている場合は，それらの組が原因の可能性がある．

3.2.3　ルール不足や実行優先度が原因の場合

反例の原因にはルールの誤りだけではなく，振舞いモデルの記述に原因がある場合もある．例えば，図 9 のように他の状態への遷移を持たずに反例となる場合はルールが不足している可能性がある．また，図 10 のように反例にならない遷移を持つにもかかわらず反例となってしまう場合はルールの実行優先度が原因の可能性がある．

4　関連研究

熊澤らは，満たすべき性質を満たすように修正した反例の軌跡を修正候補とし，有向グラフ上の最短経路問題に帰着させる．求めた軌跡と反例の差分から修正箇所を提示する方法を提案している [1]．この手法では最短経路問題として軌跡を求めるため，システム全

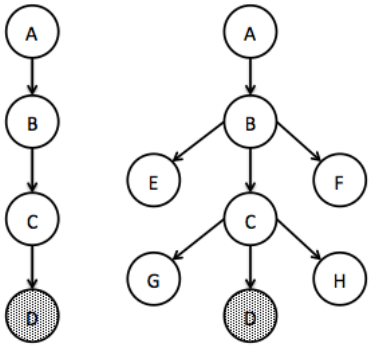

図 8　反例パスと反例パス上の状態を持つ遷移集合

表 1　各遷移で適用されたルール

遷移	適用ルール	反例になるパスか
AB	$R1$	○
	R_1	×
	R_1	×
	R_1	×
	R_1	×
BC	R_2	○
BE	R_3	×
BF	R_4	×
CD	R_3	○
CG	R_5	×
CH	R_2	×

体の状態モデルが必要となるので段階的検査法で利用することはできない．中野らは，学習理論を用いて反例集合の全体を獲得し，それらをオートマトンで表すことで修正方法の決定を容易にする方法を提案している [2]．本研究では設計書のルール記述から原因候補を探しだしている点で異なっている．

5　考察

一般的なモデル検査の反例分析では，基となるモデルに合わせた大きな状態空間で分析を行う必要があった．特に，非常に長いパスの中から組合せや順序など，一致する箇所を探しだすのは困難である．しかし，段階的検査手法では，検証の際に作成した合成遷移集合を利用することで，大きな状態空間や長いパスを使うことなく，問い合わせで組合せや順序の一致を見ることができる．たくさんある情報の中から必要な情報だけを抽出できることは有用であると考える．

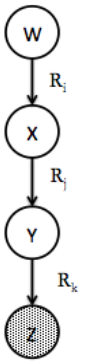

図 9　ルール不足がありうる遷移モデル

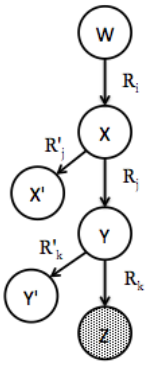

図 10　ルールの実行優先度に問題がありうる遷移モデル

6　まとめ

本論文では状態爆発問題を回避するために考案された段階的検査手法を適用したときの反例分析手法について述べた．段階的検査を進める過程で作成される合成遷移集合を利用することで，問題の原因となりそうな箇所を検索によって探しだすことができる．今後は，反例の原因をさらに限定できるような検索手順を考えていく必要がある．また，自動化を行い，実際に段階的検査手法を適用した際の有効性を調査する必要がある．

謝辞 本研究は JSPS 科研費 25330075 の助成を受けた．

参 考 文 献

[1] 熊澤努ら:モデルに基づく誤り特定と反例修正候補の提示, ソフトウェアエンジニアリングシンポジウム *2009* 論文集, *pp.55-62*, 2009.
[2] 中野昌弘ら:モデル検査における反例集合のオートマトンによる表現とその獲得方法, 第 *6* 回ディペンダブルシステムワークショップ *(DSW2008), pp.3-12*, 2008.
[3] 宮島卓巳ら:モジュラ化手法によるモデル検査の検討とモジュラ検証の実用化, 信学技報 *114(501), 25-30*,2015
[4] SPIN,*http://spinroot.com/spin/what.html*,2015
[5] NuSMV home page,*http://nusmv.fbk.eu/*,2015

[6] UPPAAL,*http://www.uppaal.org/*,2015
[7] 小山恭平ら:情報制御システムのモデル検査における状態空間分割による探索手法の提案, ソフトウェア工学の基礎 *XIX,pp.39-44*,2012
[8] 吉岡信和ら:SPIN による設計モデル検証, 近代科学社,2008
[9] Mordechai Ben-Ari 著, 中島震 監訳:SPIN モデル検査入門, オーム社,2010