様式 C-19

科学研究費補助金研究成果報告書

平成２３年６月１０日現在

機関番号：８２６２６
研究種目：基盤研究（Ｃ）
研究期間：2008～2010
課題番号：２０５０００２３
研究課題名（和文）並行システムの高信頼自動検証ツールに関する研究

研究課題名(英文)A study on a reliable automatic verification tool for concurrent systems

研究代表者
磯部　祥尚（ISOBE YOSHINAO）
独立行政法人産業技術総合研究所・情報技術研究部門・主任研究員
研究者番号：５０３５６４５８

研究成果の概要（和文）：

本研究では、並行システムの設計を支援するため、モデル検査器のように自動的に、かつ定理証明器のように記号的にそのシステムの動作を解析するツールの開発を目標として、(1)定理証明器の証明自動化と(2)モデル検査器の記号処理化の二つの側面から研究を行った。前者の(1)については、並行動作の理論 CSP に基づく定理証明器 CSP-Prover にモデル検査の自動検査アルゴリズムを実装し、証明自動化の可能性を示した。後者の(2)については、並行システムの構造や各プロセスの動作から、そのシステム全体の動作を記号処理によって自動解析する方法を提案し、その方法を解析ツール CONPASU として実装した。例えば CONPASU は、無限状態の並行システムから、その動作を理解するために有益な記号ラベル付き状態遷移図を自動生成することができる。

研究成果の概要（英文）：

This work aims at developing tools which analyze concurrent systems automatically like model-checkers and symbolically like theorem-provers, in order to support designs of the systems. Then, we have studied on such analysis-tools from two points of views: (1) automatizing proofs in theorem-provers and (2) introducing symbolic computation to model-checkers. In the former (1), we showed that proofs in CSP-Prover, which is a theorem-prover for a concurrency theory called CSP, can be automatized by implementing a model-checking algorithm to CSP-Prover. In the latter (2), we presented a method for automatically analyzing the whole behaviors from structures of concurrent systems and behaviors of component-processes by symbolic computation, and we implemented the analysis-method in a prototype-tool called CONPASU. For example, CONPASU can automatically generate symbolic-labeled transition graphs, which are useful for understanding the behaviors, from concurrent systems with infinite-states.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
２００８年度	1,100,000	330,000	1,430,000
２００９年度	900,000	270,000	1,170,000
２０１０年度	700,000	210,000	910,000
年度			
年度			
総　計	2,700,000	810,000	3,510,000

研究分野：情報科学

科研費の分科・細目：情報学・情報学基礎
キーワード：並行システム，プロセス代数，自動解析ツール，記号処理，状態数削減，無限状態システム，定理証明器，モデル検査器

１．研究開始当初の背景

処理の並列化によってシステムの設計はさらに困難になってきており、そのような並行システムの正しさを実装前に検証するためのツールとしてモデル検査器や定理証明器が研究・開発されている。モデル検査器ではその検証が完全に自動化されているため、産業界での適用事例も報告されるようになってきた。しかし、全ての状態を探索するため、状態数の多いシステムでは検証できなくなる“状態爆発”とよばれる問題がある。一方、定理証明器では記号処理（変数を値で具体化せずに記号として処理する）によって無限状態システムも検証可能であるが、人手による証明支援が必要であり、証明の完全自動化は難しいという問題がある。

２．研究の目的

本研究では、並行システムの設計支援を目的として、モデル検査器の自動解析能力と定理証明器の記号処理能力を併せもつ解析ツールの開発を目標とした。本研究開始当時（2008 年)、既に本研究代表者（磯部）はスウォンジー大学の Roggenbach 博士と共に汎用定理証明器（Isabelle）上に並行プロセスの理論 CSP を実装し、CSP のための定理証明器 CSP-Prover を開発していた。この定理証明器 CSP-Prover には約 70 個の CSP 規則が用意されており、それらを半自動的に適用することによって、無限状態をもつ並行システムに対しても仕様通りに動作することを証明することができた。しかし、CSP-Prover による証明過程では、適用する規則を人が適切に指示する必要があり、その利便性には課題が残されていた。そこで本研究では、CSP-Prover にモデル検査的な解析方法を導入し、証明を自動化することによって、利便性の向上を目標とした。

３．研究の方法

モデル検査器と定理証明器の特長を融合させるため、本研究では次の二つの側面から解析ツールの研究開発を行った。

(1) 定理証明器 CSP-Prover の利便性改善

汎用定理証明器（Isabelle）には、状況に応じて規則を選択して適用する戦略（tactics）を定義し、その戦略を実行するための証明コマンドを追加する機能が用意されている。並行動作を解析するための CSP 規則には様々な適用パターンがあるため、それを戦略とする証明コマンドを CSP-Prover に追加し、規則の自動適用を可能にする。また、モデル検査のアルゴリズムを定理証明器 CSP-Prover 上に実装することによって、証明の自動化を試みる。

(2) モデル検査的な方法への記号処理導入

自動解析できる範囲でモデル検査的な方法に記号処理を導入し、無限状態システムも解析可能なツールを開発する。そのために、並行動作の記述（CSP モデル）から記号ラベル付状態遷移図を作成する記号的な操作的意味論と、その状態数を削減する方法を考案し、その方法を解析ツールとして実装する。

４．研究成果

定理証明器 CSP-Prover の利便性改善に関する成果と、モデル検査的な解析方法に記号処理を導入して開発した解析ツール CONPASU の成果について報告する。

(1) 定理証明器 CSP-Prover の利便性改善

CSP-Prover の利便性を向上させるため、証明戦略（tactics）の強化、モデル検査の方法の導入、CSP-Prover の表現力の拡張等を行った。以下、各成果について報告する。

① 証明コマンドの強化：証明の自動化に向けて補助的な証明規則を証明し、それらを証明戦略に組み込んで、証明の自動化を進めた。その証明コマンドをスケーラブル（プロセス数可変）な自己組織化分散システムの検証に適用してその効果を確認した。この成果を日本ソフトウェア科学会の論文誌にて発表した（JSSST2008)。

② 定理証明器へのモデル検査方法の実装と適用実験：CSP-Prover 上にモデル検査方法（CSP の操作的意味論、到達可能状態導出アルゴリズム、弱双模倣等価性判定アルゴリズム）を実装し、テスト用の例題に適用した結果について研究会で発表した（TPP2008, PPL2009)。この適用実験によって、検証自動化の可能性を確認することができたが、その検証時間が予想以上にかかることも判明した。例えば、実験では数十状態の等価性判定

に数分間を必要としたため（通常のモデル検査器では数十万状態でも数秒以下）、この方法による証明自動化は性能的に難しいという結果となった。

③ CSP-Prover の拡張：CSP-Prover の表現力を改善するため、主にスウォンジー大学（ウェールズ、UK）がデータ表現の強化と動作表現の拡張を行い、国際研究会でその成果を発表した（AVoCS2008 2 件）。

(2) モデル検査的な方法への記号処理導入

上記の成果(1)②の実験結果から、定理証明器単体で自動化を進めることは困難であると判断し、モデル検査器の自動解析方法に記号処理を導入することによって、状態爆発を回避する方法の検討を行った。その成果として、並行プロセスのモデルから記号ラベル付状態遷移システムを生成するための方法（記号的意味論と状態数削減法）を考案し、その方法を並行プロセス解析支援ツール CONPASU として実装した。以下、CONPASU の研究で検討した二つの状態数削減法、CONPASU による解析例、関連研究について述べる。

① 弱双模倣等価性による状態数削減：弱双模倣等価性と呼ばれる振舞いの等しさを保存するように、観測できない内部動作をバイパスする状態数削減法を考案し、その方法を CONPASU のプロトタイプに実装して、その有効性を示した（例：変数を具体化すると変域 [0,24] で状態数 10,944 になる動作を、CONPASU では無限の変域でも状態数 8 で表現可能）。また、CONPASU では確認したい一部の動作に着目した部分特性を自動生成できるなど、既存ツールにはない結果を得ることもできた。その成果をソフトウェア工学の基礎ワークショップ（FOSE2010）で発表した。

② 失敗等価性による状態数削減：上記の①の成果によって記号処理による解析の有効性を示すことができたが、弱双模倣等価性では条件付の内部動作を適切にバイパスすることができなかった。そこで、弱双模倣等価性の代わりに失敗等価性を採用することによってその問題を解決した。また、一度得られたバイパス可能な遷移の探索結果を、繰り返し再利用して状態数を削減できるようにしたことも本方法で工夫した点の一つである。この失敗等価性を保存する状態数削減法を CONPASU に実装し（Java 言語, 6,000 行程度）、その有効性を示した。その成果を論文にまとめて国際会議（CPA2011）に投稿し、採択された（2011 年 6 月 21 日発表予定）。

③ 解析例：図 1 のデータ列を転送する並行システム TransferSys の例を用いて CONPASU による解析の概要を説明する。

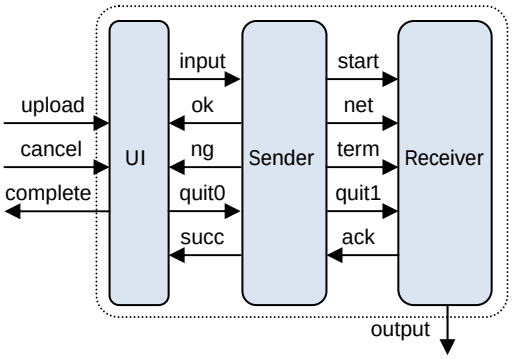

図 1. データ列転送システムの構造

これは 3 つのプロセス UI, Sender, Receiver から構成される並行システムである。各プロセスは図中のチャネル（矢印）で同期通信をしながら並行に動作する。図 2 に各プロセスの動作を記号ラベル付状態遷移図で示す。

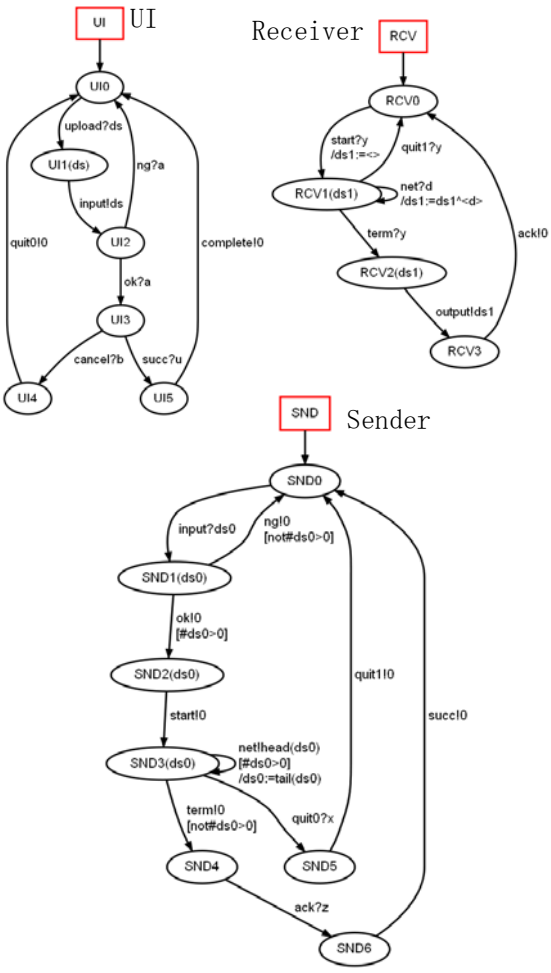

図 2. UI、Sender、Receiverの状態遷移図

このシステムでは転送するデータの種類

や数を制限していないため、変数を値で具体化すると状態数が無限になる（記号ラベルでは変数のまま表現されている）。このとき、プロセス間の送信イベントと受信イベントの同期を考慮しつつ、全体の動作を予想することは容易ではない。CONPASU は、図 1 の構造情報と図 2 の動作情報から、記号的な操作的意味論と状態数削減法によって、図 3 に示す記号ラベル付状態遷移図を自動生成することができる。この図 3 は TransferSys を外部から観測したときの動作を表している。

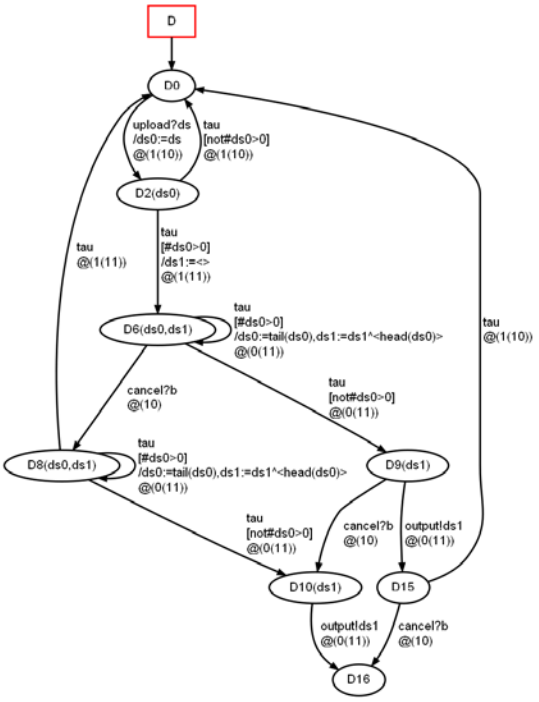

図 3. TransferSysの状態遷移図

図 3 から TransferSys が観測的にどのように動作するかを把握でき、右下にデッドロック状態の存在も確認できる。このデッドロックは、Sender がデータ転送完了直後に、ユーザがキャンセルボタンを押したときに発生する。そのような可能性は低いが 0 ではない。この不具合を解消するため、転送完了後もキャンセル処理できるように Sender を修正することが考えられる。実際にそのように Sender を修正して、再び CONPASU で解析した結果を図 4 に示す。図 4 ではデッドロック状態がなくなっていることが確認できる。

④関連研究：並行システムの状態遷移図を表示する機能をもつモデル検査器もあるが、それらは基礎的な意味論（記号的意味論ではない）を用いている。そのため、入力されるデータ数を有限に制限しても、変数が各データによって具体化されるため、状態数はデータ数と共に増加する。現在、図 3 や図 4 のような状態遷移図を自動生成ツールは他にない。

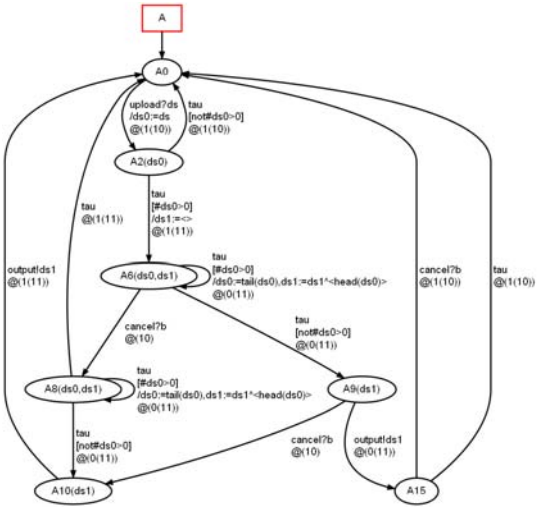

図 4. 修正版TransferSysの状態遷移図

以上、CSP-Prover と CONPASU の成果について説明してきた。上記成果(1)②が示すように、モデル検査の方法による CSP-Prover の証明自動化は性能的に難しいが、CONPASU の解析結果を CSP-Prover の証明自動化に利用できると考えている。これは今後の課題である。

５．主な発表論文等
（研究代表者、研究分担者及び連携研究者には下線）

〔雑誌論文〕（計６件）

① 磯部祥尚, CONPASU-tool: A Concurrent Process Analysis Support Tool based on Symbolic Computation, Communicating, Process Architectures 2011, WoTUG-33, IOS Press, 22 頁, 2011, 査読有. （掲載確定）

② 磯部祥尚, CONPASU-tool: 記号処理に基づく並行プロセス解析支援ツールの試作, レクチャーノート/ソフトウェア学 ソフトウェア工学の基礎XVII, 36 巻, 近代科学社, pp. 65-74, 2010, 査読有.

③ 磯部祥尚, 並行システムを解析するための逐次化と状態削減機能の実装 ～仕様の自動生成を目指して～, 電子情報通信学会技術研究報告CST（コンカレント工学）、110 巻 89 号, pp. 139-144, 2010, 査読無.

④ Gift Samuel, Markus Roggenbach, and 磯部祥尚, The Stable Revivals Model in CSP-Prover, Electronic Notes in Theoretical Computer Science (ENTCS),

Vol.250, No.2, pp.119-134, 2009, 査読有.

⑤ Liam O'reilly, Markus Roggenbach, and 磯部祥尚, CSP-CASL-Prover -- A generic tool for process and data refinement, Electronic Notes in Theoretical Computer Science (ENTCS), Vol.250, No.2, pp.69-84, 2009, 査読有.

⑥ 磯部祥尚, CSP-Prover -- a Proof Tool for the Verification of Scalable Concurrent Systems, 日本ソフトウェア科学会コンピュータソフトウェア (JSSST), 25 巻 4 号, pp.85-92, 2008, 査読有.

〔学会発表〕(計１７件)

① 磯部祥尚, CONPASU-tool: A Concurrent Process Analysis Support Tool based on Symbolic Computation, Communicating Process Architectures 2011, (CPA 2011, the 33rd WoTUG conference), 2011 年 06 月 21 日発表予定 (採択済), リムリック大学 (アイルランド).

② 磯部祥尚, CONPASU: CSPモデルの解析支援ツール, 第 7 回CSP研究会, 2011 年 05 月 21 日, 東洋大学 (東京都).

③ 磯部祥尚, CONPASU: 記号処理に基づく並行プロセスの状態数削減ツール, 第 13 回プログラミングおよびプログラミング言語ワークショップ (PPL2011), 2011 年 03 月 10 日, 定山渓ビューホテル (北海道).

④ 磯部祥尚, CONPASU-tool: 記号処理に基づく並行プロセス解析支援ツールの試作, 第 17 回ソフトウェア工学の基礎ワークショップ (FOSE2010), 2010 年 11 月 19 日, いなもと旅館 (新潟県).

⑤ 磯部祥尚, 並行システムを解析するための逐次化と状態削減機能の実装 ～仕様の自動生成を目指して～, 電子情報通信学会コンカレント工学研究会 (CST) 2010 年 06 月 22 日 北見工業大学 (北海道).

⑥ 磯部祥尚, A Prototype Tool for Analyzing Concurrent Processes -- Towards Automatic Generation of Specifications, Workshop on Symbolic Computation and Software Verification, 2010 年 04 月 06 日, 筑波大学 (茨城県).

⑦ 磯部祥尚, CSPプロセスの解析支援ツールの試作 ～仕様の自動生成を目指して～, 第 4 回CSP研究会, 2010 年 03 月 10 日, 東洋大学 (東京都).

⑧ 磯部祥尚, プロセス代数に基づく並行システムの動作解析のための逐次化ツール, 第 12 回プログラミングおよびプログラミング言語ワークショップ (PPL2010), 2010 年 03 月 04 日, 琴参閣 (香川県).

⑨ 磯部祥尚, 定理証明器による双模倣等価性の自動証明, 第 11 回プログラミングおよびプログラミング言語ワークショップ (PPL2009), 2009 年 03 月 10 日, 高山グリーンホテル (岐阜).

⑩ 磯部祥尚, The First Step for Implementing a Model Checker in a Theorem Prover -- Toward Automatic Verification in CSP-Prover, Theorem Proving and Provers (TPP2008) Meeting, 2008 年 11 月 26 日, 東北大学電気通信研究所 (宮城県).

⑪ Liam O'reilly, Markus Roggenbach, and 磯部祥尚, CSP-CASL-Prover -- A generic tool for process and data refinement, 8th International Workshop on Automated Verification of Critical Systems (AVoCS 2008), 2008 年 09 月 30 日, グラスゴー大学 (スコットランド).

⑫ Gift Samuel, Markus Roggenbach, and 磯部祥尚, The Stable Revivals Model in CSP-Prover, 8th International Workshop on Automated Verification of Critical Systems (AVoCS2008), 2008 年 10 月 01 日, グラスゴー大学 (スコットランド).

〔その他〕
CONPASU のホームページ:
http://staff.aist.go.jp/y-isobe/conpasu

CSP-Prover のホームページ;
http://staff.aist.go.jp/y-isobe/CSP-Prover/CSP-Prover.html

６. 研究組織
(1)研究代表者
磯部　祥尚 (ISOBE YOSHINAO)
独立行政法人産業技術総合研究所・情報技術研究部門・主任研究員
研究者番号：50356458