

令和 5 年 6 月 1 日現在

機関番号：13901

研究種目：基盤研究(C)（一般）

研究期間：2018～2022

課題番号：18K11160

研究課題名（和文）書換え帰納法を利用したプログラム等価性検証技術の開発

研究課題名（英文）Development of Verification Techniques for Equivalence of Programs via Rewriting Induction

研究代表者

西田 直樹（Nishida, Naoki）

名古屋大学・情報学研究科・准教授

研究者番号：00397449

交付決定額（研究期間全体）：（直接経費） 3,400,000円

研究成果の概要（和文）：本研究では、ポインタを扱わない命令型プログラム、特に、C言語プログラムやLLVM中間表現を論理制約付き項書換えシステムに変換し、書換え帰納法および全パス到達可能性証明系に基づくプログラム検証の枠組みを構築し、検証ツールを実装した。プログラムの等価性証明は帰納的定理に帰着して書換え帰納法により証明し、指定した実行時エラーの非発生は全パス到達可能性問題に帰着して余帰納法に基づく証明系により証明する。さらに書換え帰納法に基づく証明系と循環証明系の証明力を比較するために、特定の条件を満たす帰納的述語に関するシークエントの恒真性を証明する循環証明と書換え帰納法証明が互いに変換可能であることを示した。

研究成果の学術的意義や社会的意義

本研究では、書換え理論を実用プログラムに応用することをめざし、車載組み込みシステムの検証に提案手法を応用することを試みた。この応用はこれまでにない試みであり、本研究の成果は書換え理論、特に制約付き書換えの実用性・有用性を示したと言える。さらに、これまでに研究されてきた多くの書換え理論の研究成果が提案手法を通じて応用できる可能性も示した。その観点から学術的意義だけでなく、社会的意義がある研究課題であることも示したと言える。

一方、全パス到達可能性を実行時エラーの非発生に証明に応用することも新しい試みであり、その有用性・実用性の観点から今後、さらに研究する価値がある課題であることを示した。

研究成果の概要（英文）：In this research project, we established a framework for verification of pointer-less imperative programs, especially C programs and LLVM intermediate representations: We first transform a program into an equivalent logically constrained term rewrite system, and then verify the rewrite system by means of rewriting induction or a proof system for all-path reachability. Then, we implemented a verification tool based on the framework. In the framework, equivalence of programs is reduced to inductive theorems, and is verified by rewriting induction. Non-occurrence of a specified runtime error is reduced to an all-path reachability problem, and is verified by a proof system based on co-induction. In addition, to compare a proof system based on rewriting induction with a cyclic proof system, for a sequent w.r.t. inductive definitions satisfying a certain condition, we showed transformations between a cyclic proof and a rewriting-induction proof for validity of the sequent.

研究分野：書換え理論

キーワード：プログラム変換 制約付き書換え プログラム検証 書換え帰納法 等価性 実行時エラー検証 全パス到達可能性 定理自動証明

1. 研究開始当初の背景

項書換えシステム (TRS) は関数型プログラミング言語の計算モデルとして盛んに研究され、関数型プログラムの停止性を検証する技術など理論的に興味深い成果が多いだけでなく、様々な性質を自動証明する技術の開発も盛んに行われてきた。近年、計算機性能の向上により既存の理論に基づく検証技術を実際の問題解決に応用できるようになってきている。項書換えシステムはオブジェクトを項と呼ばれる木構造のデータで表現するため、リストや木など再帰的なデータ構造を扱う事象については非常に相性が良い。また、自然数は項書換えシステムで扱いやすいデータである。一方、ほとんどのプログラミング言語で利用できる整数を項として扱うには符号の区別など自然数では必要としなかった工夫が必要となる。

プログラミング言語に組み込まれている整数とその演算子などは意味および定義が固定されている。よって、プログラムを検証する際には、算術演算や大小比較をユーザ定義関数と切り離し、本来検証したいユーザが定義する振る舞いのみを検証すればよい。TRS の分野でそれを可能とする枠組みとして 2000 年代に制約付き項書換えシステムの枠組みの研究 [1] [2] [3] [4] が再び盛んになり、SMT ソルバがサポートする背景理論を柔軟にサポートできる枠組みとして論理制約付き項書換えシステム (LCTRS) [2] が提案された。LCTRS は TRS の書換え規則 (関数定義) に適用条件に相当する制約を付随できるように拡張したもので、あらかじめ指定した記号の意味を計算時や検証時に利用できる。これにより C 言語プログラムのような命令型プログラムを LCTRS で自然に表現することが可能となったため、LCTRS は関数型プログラミング言語だけでなく命令型プログラミング言語の計算モデルとして利用できる [1] [5] [6] (図 1)。さらに、TRS のために開発された理論や技術を LCTRS に移植することで、命令型プログラムの性質検証に TRS の理論を応用できる [7] [8] [5] [9]。特に注目される性質は「等式が帰納的定理であること」(すなわち、等式が書換えシステムの計算の下で恒真であること) である。例えば、1 引数関数 f, g の等価性を等式 $f(x)=g(x)$ が帰納的定理である ($\forall x. f(x)=g(x)$ が真である) ことに帰着できる。

等式が帰納的定理であることの証明手法の一つである書換え帰納法 [10] は人間が行う証明に沿った推論規則で構成され、TRS から LCTRS に移植された [7] [5] [6] [3]。書換え帰納法の推論規則は定理証明支援システム Coq, Isabelle/HOL などに組み込まれた証明ステップに対応する。一方、Coq や Isabelle/HOL では構造に関して明示的に減少する関数定義しか扱えないのに対し、書換え帰納法は書換えシステムの停止性証明技術により停止性を証明できるシステムすべてを対象とし、推論規則の適用に戦略を与えることで自動化しやすい。また、書換え帰納法では証明の過程で必要に応じて補題等式を自動生成する技術も開発されている。それらの技術の組み合わせにより、ホア論理を用いた検証ではループ不変式を与える必要がある検証例に対しても関数定義だけを与えるだけで正しさを自動証明できる例も存在する [11] [5] [6]。

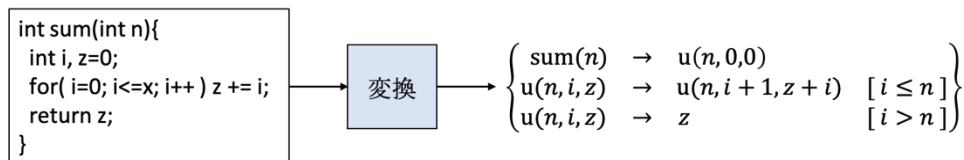


図 1 : C 言語プログラムから制約付き項書換えシステムの変換例

2. 研究の目的

本研究では、関数型プログラミング言語だけでなく命令型プログラミング言語の計算モデルとしての利用を期待できる LCTRS に対する書換え帰納法に基づく定理自動証明ツールを開発し、命令型プログラミング言語のプログラム等価性検証の新たな手法の確立をめざす。具体的には、LCTRS に対する書換え帰納法に基づく帰納的定理証明法を命令型プログラミング言語から得られた書換えシステム向けに自動化することで、命令型プログラム向けの定理自動証明ツールを開発することを目標とする。特に、関数の等価性など等式で表現できる性質の検証への応用を試みる。命令型プログラム向けの自動化を実現するために、書換え帰納法の推論規則の適用戦略の改良、補題等式の生成法の開発に取り組む。また、不等式も同時に扱い、等式付き書換えに対応するなど理論の拡張にも取り組む。

3. 研究の方法

1 年目にはまず LCTRS の書換え帰納法に基づく定理自動証明ツールの開発に取り組んだ。具体的には、様々な例題に対して証明を試みることで書換え帰納法の推論規則の適用戦略に関するヒューリスティクスを収集し、ツールとして実装した。なお、この開発は最終年まで常時実施した。さらに、書換え帰納法の正当性証明の抽象的な枠組みの構築に向けて、Isabelle/HOL 上

での書換え帰納法の形式化にも取り組んだ。

2年目には、書換え帰納法の枠組みを等式と不等式の両方を同時に扱える枠組みの構築をめざし、書換え帰納法の推論規則の適用戦略の開発、等式と不等式が混同した枠組みへの書換え帰納法の拡張に取り組んだ。また、1年目の成果に基づき、ビットベクトルを扱う LCTRS (BV-LCTRS) の停止性証明法の開発、拡張性の高い検証ツールの再実装に取り組んだ。

3年目には推論規則の適用戦略の開発過程で有用であった補題等式を生成するヒューリスティックを開発ツールに導入することをめざし、書換え帰納法の推論規則の適用戦略の開発、補題等式の自動生成法の開発、LCTRS の書換え帰納法の等式付き書換えへの拡張に取り組んだ。また、2年目の並行プログラムの排他制御の検証に関する研究成果を踏まえ、LCTRS による並行プログラムのモデル化、ビットベクトルをデータとして扱う LCTRS の停止性証明法の開発にも取り組んだ。

4年目には書換え帰納法の推論規則の適用戦略の開発だけでなく、3年目までの進捗状況から設定した課題として、LCTRS による並行プログラムのモデル化、ビットベクトルをデータとして扱う LCTRS の停止性証明法の開発と実装に取り組んだ。

最終年には指定した実行時エラーの非発生を全パス到達可能性 (all-path reachability) に帰着して検証する枠組みの開発に取り組む、本研究で開発している書換え帰納法の対話的証明ツールに全パス到達可能性問題の自動証明機能を実装した。

4. 研究成果

本研究では、ポインタを扱わない命令型プログラム、特に、C 言語プログラムや LLVM 中間表現を論理制約付き項書換えシステムに変換し、書換え帰納法および全パス到達可能性証明系に基づくプログラム検証の枠組みを構築し、検証ツールを実装した (図 2)。

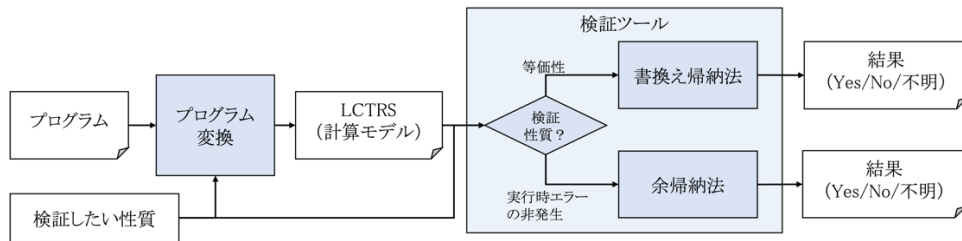


図 2: 提案手法の概要 (LCTRS に変換して性質を検証する枠組み)

(1) C 言語プログラムを対象に、ポインタ以外の基本データ型やユーザ定義データ型 (構造体や共用体を含む) を扱うプログラムを固定長の 0, 1 の列であるビットベクトル上で定義された LCTRS に変換する枠組みを構築し、実装した。その際に、大域変数とそれにアクセスする関数を扱うプログラムの変換を改良した。この枠組みでは従来はデータとして整数とその 1 次元配列のみを扱うプログラムしか変換対象でなかった点を、より多くの種類のプログラムを変換して解析できる枠組みに拡張した。そして、一重の標準的な for ループを持つプログラムから変換して得られる BV-LCTRS の停止性を証明する定理を導き、その正しさを証明した。さらに、ビットベクトル上の多項式解釈を提案し、それらに基づく停止性証明法を検証ツールに実装した。そして、提案手法が対象とするプログラミング言語を拡大することを目的に、LLVM 中間表現を対象とするように拡張を行なった。従来の対象としていた C 言語プログラムをコンパイルして得られる LLVM 中間表現 (主にポインタを扱わないもの) に対する変換を提案した。

(2) LCTRS を対象とした定理自動証明ツールの開発として、これまでに研究代表者が開発してきたツール Ctrl がビットベクトルを扱うための理論定義ファイルを作成した。さらに、自動証明機能の改良を目的として試行錯誤を容易にするラッパーの開発に取り組んだ。このラッパーは Ctrl の対話型証明機能には実装されていなかった補題候補の等式の追加を可能にした。さらに、対話証明の際に利用したコマンドの履歴に対して、改めて Ctrl を実行する際に、ファイルに記載された履歴を自動実行したのちに対話証明を開始できる機能を追加した。これにより推論規則の適用戦略の開発のための試行錯誤が容易となり、実際に試行錯誤を行った結果、新たな補題候補の自動生成法を開発するに至った。新たな補題生成法を組み込んだ自動証明では、従来の Ctrl が自動証明に成功していなかった例題の自動証明に成功した。

(3) 高信頼な検証法の開発に向けて、LCTRS、整数上の単純な命令型言語 IMP から LCTRS への変換、IMP から LCTRS への変換とその正当性の証明を Isabelle/HOL 上で形式化した。これにより、書換え帰納法によるプログラム検証に対する証明承認ツールの基盤となる理論の一部の形式化を達成した。

(4) 書換え帰納法における補題生成では、単純な一般化では変数間の関係性を失い、成り立た

ない補題候補の等式へと一般化してしまうことがある。その問題を解消するため、ラグランジュ補間を用いて変数間に成り立つ性質を発散等式の列から求めて制約に追加するアプローチを提案した。そして、これまでに自動証明に成功していないプログラムの例に対して、適切な補題を自動で生成し、等価性証明に成功する補題生成法と適用戦略を提案した。具体的には証明を発散させる等式の列の特定の場所に注目して点列を抽出し、ラグランジュ補間を用いて等式を一般化する。その手法では成り立たない補題等式を生成してしまうが、途中の証明の失敗を利用して適切な補題に修正する証明戦略を開発した。

(5) 書換え帰納法の対象を等式から不等式へ拡張する課題については、不等式を直接扱うのではなく、不等式と等価な存在限量子付き等式に変換し、書換え帰納法を存在限量子付き等式へ拡張することで、間接的に不等式を証明する枠組みを提案した。不等式をそのまま扱う場合には等式とは異なる扱い方となり、推論規則が複雑になる。そこで等式に存在限量子を導入することで、不等式と等式の両方を存在限量子付き等式として扱うこととし、枠組みが複雑にならないようにした。

(6) 検証ツールの再実装では、標準でないライブラリを用いない方針のもとで実施した。入力ファイルの構文への拡張性については `Ocaml` の `lex/yacc` を用いて実装し、`LCTRS` の基本データ型として整数、ブール、ビットベクトル、それらの配列を扱えるようにした。

(7) `LCTRS` の書換え帰納法を等式付き書換えへ拡張する必要性を検討することを目的に、停止性を持ち、特定の条件を満たす帰納的述語とそのシーケントについて、循環証明系と書換え帰納法の証明力が等価であることを、循環証明と書換え帰納法証明を互いに変換して得られることを示すこと証明した。

(8) 提案手法を車載組み込みシステムの検証に応用することをめざして、排他制御を含むプログラムから `LCTRS` への変換を提案した。変換の正しさの証明のために並行実行の意味論を定義する簡易プログラミング言語 `SIMP` の並行計算版を推論規則で定式化し、その意味論に対して変換の正当性を証明した。そして、セマフォによる排他制御を含むプログラムから `LCTRS` への変換を提案した。

(9) 実行時エラーの非発生を等価性ではなく全パス到達可能性に帰着させて検証する枠組みを提案した。全パス到達可能性の証明システムは書換え帰納法の推論規則の使用を限定して実現できることを示し、開発している検証ツールの証明機能を制限することで全パス到達可能性を証明する機能を追加実装した。具体例として、非同期整数状態遷移システム、居眠り床屋問題、並行バグを含む例を `LCTRS` に変換し、排他制御について競合状態・デッドロック・スタベーションが発生しないことを証明する実験を実施した。この実験により、規模が小さいプログラムについては実装により自動証明できることを示した。書換え帰納法の推論規則に基づいて実現した全パス到達可能性問題の自動証明機能の実現のために、幅優先探索に基づく推論規則の適用戦略を開発し、実装した。証明に失敗する例に対して、検証したい `LCTRS` が表現するシステムの初期状態を一般化する補題生成の有効性を実験により明らかにした。さらに、反証方法、公平でないパスを除外した証明法を提案し、実装した。

<引用文献>

- [1] 古市, 西田, 酒井, 草刈, 坂部, “制約付き項書換え系の潜在帰納法を利用した手続き型プログラム検証の試み,” 情報処理学会論文誌プログラミング **1**(2):100–121, 2008.
- [2] C. Kop and N. Nishida, “Term Rewriting with Logical Constraints,” in *Proc. FroCoS 2013*, LNCS 8152, pp. 343–358, Springer, 2013.
- [3] A. Bouhoula and F. Jacquemard, “Automated induction with constrained tree automata,” in *Proc. IJCAR 2008*, LNCS 5159, pp. 539–554, Springer, 2008.
- [4] C. Fuhs, J. Giesl, M. Plücker, P. Schneider-Kamp and S. Falke, “Proving Termination of Integer Term Rewriting,” in *Proc. RTA 2009*, LNCS 5595, pp. 32–47, Springer, 2009.
- [5] C. Kop and N. Nishida, “Automatic Constrained Rewriting Induction towards Verifying Procedural Programs,” in *Proc. APLAS 2014*, LNCS 8858, pp. 334–353, Springer, 2014.
- [6] C. Fuhs, C. Kop and N. Nishida, “Verifying Procedural Programs via Constrained Rewriting Induction,” *ACM Transactions on Computational Logic* **18**(2):14:1–14:50, 2017.
- [7] 坂田, 西田, 坂部, 酒井, 草刈, “制約付き項書換え系における書換え帰納法,” 情報処理学会論文誌プログラミング **2**(2):80–96, 2009.
- [8] T. Sakata, N. Nishida and T. Sakabe, “On Proving Termination of Constrained Term Rewrite Systems by Eliminating Edges from Dependency Graphs,” in *Proc. WFLP 2011*, LNCS 6816, pp.

138–155, Springer, 2011.

- [9] C. Otto, M. Brockschmidt, C. von Essen and J. Giesl, “Automated termination analysis of Java bytecode by term rewriting,” in *Proc. RTA 2010*, LIPIcs 6, pp. 259–276, 2010.
- [10] U. S. Reddy, “Term rewriting induction,” in *Proc. CADE 1990*, LNCS 449, pp. 162–177, Springer, 1990.
- [11] 中林, 西田, 草刈, 坂部, 酒井, “制約付き項書換え系の書換え帰納法における補題等式の自動生成法,” *コンピュータソフトウェア* **28**(1): 173–189, 2011.

5. 主な発表論文等

〔雑誌論文〕 計2件（うち査読付論文 2件 / うち国際共著 0件 / うちオープンアクセス 0件）

1. 著者名 Shujun Zhang and Naoki Nishida	4. 巻 127
2. 論文標題 Transforming orthogonal inductive definition sets into confluent term rewrite systems	5. 発行年 2022年
3. 雑誌名 Journal of Logical and Algebraic Methods in Programming	6. 最初と最後の頁 1-17
掲載論文のDOI（デジタルオブジェクト識別子） 10.1016/j.jlamp.2022.100779	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

1. 著者名 Takahiro Nagao and Naoki Nishida	4. 巻 155
2. 論文標題 Rewriting induction for constrained inequalities	5. 発行年 2018年
3. 雑誌名 Science of Computer Programming	6. 最初と最後の頁 76-102
掲載論文のDOI（デジタルオブジェクト識別子） 10.1016/j.scico.2017.10.012	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

〔学会発表〕 計29件（うち招待講演 0件 / うち国際学会 13件）

1. 発表者名 Misaki Kojima and Naoki Nishida
2. 発表標題 From Starvation Freedom to All-Path Reachability Problems in Constrained Rewriting
3. 学会等名 the 25th International Symposium on Practical Aspects of Declarative Languages (国際学会)
4. 発表年 2023年

1. 発表者名 Shujun Zhang and Naoki Nishida
2. 発表標題 On Transforming Rewriting-Induction Proofs for Logical-Connective-Free Sequents into Cyclic Proofs
3. 学会等名 the 9th International Workshop on Rewriting Techniques for Program Transformations and Evaluation (国際学会)
4. 発表年 2022年

1. 発表者名 Misaki Kojima and Naoki Nishida
2. 発表標題 On Reducing Non-Occurrence of Specified Runtime Errors to All-Path Reachability Problems of Constrained Rewriting
3. 学会等名 the 9th International Workshop on Rewriting Techniques for Program Transformations and Evaluation (国際学会)
4. 発表年 2022年

1. 発表者名 Naoki Nishida, Misaki Kojima, and Takumi Kato
2. 発表標題 On Transforming Imperative Programs into Logically Constrained Term Rewrite Systems via Injective Functions from Configurations to Terms
3. 学会等名 the 9th International Workshop on Rewriting Techniques for Program Transformations and Evaluation (国際学会)
4. 発表年 2022年

1. 発表者名 Shujun Zhang and Naoki Nishida
2. 発表標題 On Transforming Cut- and Quantifier-Free Cyclic Proofs into Rewriting-Induction Proofs
3. 学会等名 the 16th International Symposium on Functional and Logic Programming (国際学会)
4. 発表年 2022年

1. 発表者名 松見歩佳, 西田直樹, 小嶋美咲, 申東訓
2. 発表標題 ビットベクトル制約付き項書換え系の停止性証明のための多項式解釈プロセッサについて
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会
4. 発表年 2023年

1. 発表者名 Naoki Nishida
2. 発表標題 On Transforming Imperative Programs into LCTRSSs via Injective Functions from Configurations to Terms
3. 学会等名 the 57th TRS meeting
4. 発表年 2022年

1. 発表者名 Shujun Zhang and Naoki Nishida
2. 発表標題 On Transforming Inductive Definition Sets into Term Rewrite Systems
3. 学会等名 8th International Workshop on Rewriting Techniques for Program Transformations and Evaluation (WPTE 2021) (国際学会)
4. 発表年 2021年

1. 発表者名 Shujun Zhang and Naoki Nishida
2. 発表標題 On Transforming Cut-free Cyclic Proofs into Rewriting Induction Proofs
3. 学会等名 5th Workshop on "Women in Logic" (WiL 2021) (国際学会)
4. 発表年 2021年

1. 発表者名 Shujun Zhang
2. 発表標題 On Transforming Inductive Definition Sets into Term Rewrite Systems
3. 学会等名 the 56th TRS meeting
4. 発表年 2022年

1. 発表者名 加藤拓洋, 西田直樹, 酒井正彦
2. 発表標題 LLVM中間表現の意味論規則を表現する制約付き書換え規則について
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会
4. 発表年 2022年

1. 発表者名 比嘉慎哉, 西田直樹, 酒井正彦
2. 発表標題 制約付き書換え帰納法におけるラグランジュ補間を用いた補題生成
3. 学会等名 日本ソフトウェア科学会第38回大会
4. 発表年 2021年

1. 発表者名 Misaki Kojima, Naoki Nishida, and Yutaka Matsubara
2. 発表標題 Transforming Concurrent Programs with Semaphores into Logically Constrained Term Rewrite Systems
3. 学会等名 the 7th International Workshop on Rewriting Techniques for Program Transformations and Evaluation (国際学会)
4. 発表年 2020年

1. 発表者名 Naoki Nishida
2. 発表標題 Transformation of Concurrent Programs with Semaphores into LCTRSSs
3. 学会等名 the 54th TRS meeting
4. 発表年 2021年

1. 発表者名 Shujun Zhang
2. 発表標題 On Transformation between Cyclic Proofs and Rewriting Induction Proof
3. 学会等名 the 54th TRS meeting
4. 発表年 2021年

1. 発表者名 小嶋美咲, 西田直樹, 酒井正彦
2. 発表標題 計数セマフォを含むプログラムから論理制約付き項書換え系への変換
3. 学会等名 情報処理学会第83回全国大会
4. 発表年 2021年

1. 発表者名 Ryota Nakayama and Naoki Nishida
2. 発表標題 On Formalizing a Transformation of IMP Programs into Logically Constrained Term Rewriting Systems in Isabelle/HOL
3. 学会等名 6th International Workshop on Rewriting Techniques for Program Transformations and Evaluation (国際学会)
4. 発表年 2019年

1. 発表者名 Carsten Fuhs, Cynthia Kop, and Naoki Nishida
2. 発表標題 Proving Program Equivalence with Constrained Rewriting Induction and Ctrl
3. 学会等名 3rd Workshop on Program Equivalence and Relational Reasoning (国際学会)
4. 発表年 2019年

1. 発表者名 西江一志, 西田直樹, 酒井正彦
2. 発表標題 存在限量子付き等式を証明するための書換え帰納法の拡張
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会
4. 発表年 2019年

1. 発表者名 Naoki Nishida
2. 発表標題 Logically Constrained Rewriting over Bit Vectors
3. 学会等名 Dagstuhl Seminar 19371: Deduction Beyond Satisfiability (国際学会)
4. 発表年 2019年

1. 発表者名 小嶋美咲, 西田直樹, 松原豊, 酒井正彦
2. 発表標題 排他制御を含むプログラムから論理制約付き項書換え系への変換
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会
4. 発表年 2020年

1. 発表者名 Donghoon Shin
2. 発表標題 On Proving Termination of Bit-Vector LCTRSS
3. 学会等名 51st TRS meeting
4. 発表年 2019年

1. 発表者名 Ryota Nakayama
2. 発表標題 On Formalizing Logically Constrained TRSs in Isabelle/HOL
3. 学会等名 51st TRS meeting
4. 発表年 2019年

1. 発表者名 Yoshiaki Kanazawa and Naoki Nishida
2. 発表標題 On Transforming Functions Accessing Global Variables into Logically Constrained Term Rewriting Systems
3. 学会等名 the 5th International Workshop on Rewriting Techniques for Program Transformations and Evaluation (WPTE 2018) (国際学会)
4. 発表年 2018年

1. 発表者名 Naoki Nishida and Sarah Winkler
2. 発表標題 Loop Detection by Logically Constrained Term Rewriting
3. 学会等名 the 10th Working Conference on Verified Software: Theories, Tools, and Experiments (VSTTE 2018) (国際学会)
4. 発表年 2018年

1. 発表者名 金澤慶明, 西田直樹, 酒井正彦
2. 発表標題 論理制約付き書換えにおける構造体および共用体の表現について
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会
4. 発表年 2019年

1. 発表者名 西江一志, 西田直樹, 酒井正彦
2. 発表標題 書換え解析ツールを利用した漸近的計算量解析のためのC言語プログラムの簡易化について
3. 学会等名 平成30年度電気・電子・情報関係学会東海支部連合大会
4. 発表年 2018年

1. 発表者名 SHIN Donghoon
2. 発表標題 A Proof Assistant for Constrained Rewriting Induction with Lemma Generation Based on Equality Derivation
3. 学会等名 the 50th TRS meeting
4. 発表年 2019年

1. 発表者名 Ryota Nakayama
2. 発表標題 Formalizing Logically Constrained Term Rewriting Systems in Isabelle/HOL
3. 学会等名 the 50th TRS meeting
4. 発表年 2019年

〔図書〕 計0件

〔産業財産権〕

〔その他〕

<p>Crisys2apr (PADL 2023 version): https://www.trs.css.i.nagoya-u.ac.jp/~nishida/padl2023/ Crisys2cdcc (WPTe 2022 version): https://www.trs.css.i.nagoya-u.ac.jp/~nishida/wpte2022/</p>
--

6. 研究組織

	氏名 (ローマ字氏名) (研究者番号)	所属研究機関・部局・職 (機関番号)	備考
--	---------------------------	-----------------------	----

7. 科研費を使用して開催した国際研究集会

〔国際研究集会〕 計0件

8. 本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関			
オランダ	ラドバウド大学ナイメーヘン	アムステルダム自由大学		
オーストリア	インスブルック大学			