様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

平成　３０　年　　５　月　３１　日現在

機関番号：　１３９０３
研究種目：　基盤研究(C)（一般）
研究期間：　2015～2017
課題番号：　１５Ｋ００３０５

研究課題名（和文）無限構造を扱うプログラム推論を用いたソフトウェア検証方法の確立

研究課題名（英文）Verifying Software Systems using Reasoning about Programs Handling Infinite Structures

研究代表者

世木　博久（SEKI, Hirohisa）

名古屋工業大学・工学（系）研究科（研究院）・教授

研究者番号：９０２４２９０８

交付決定額（研究期間全体）：（直接経費）　　3,200,000円

研究成果の概要（和文）：本研究は，リアクティブ・システムなど無限に継続する状態を扱うソフトウェアの正当性検証に向けた方法論の確立を目的とする．そのために，計算論理に基づくプログラム推論技術を中核に用いる点に特徴がある．次のような3つの研究成果を得た．(1) 推論に用いる余論理プログラム(co-logic programs) とホーンμ-計算 (Horn μ-calculus)に関する諸性質やその応用を示した．(2) 検証に必要な仕様を実行ログから導出する仕様発見アルゴリズムを提案した．(3) 仕様表現に必要とされる時間等の量的制約も含むパターン発見アルゴリズムの設計と効率的な実装を行った．

研究成果の概要（英文）：The overall objective of this research is to develop a computational-logic based methodology for verifying software such as reactive systems using program analysis; we use logic programs to represent a given system and a correctness property we want to prove, and then apply to a logic program encoding the system and the property to be verified, some methods for reasoning about programs such as program transformations that preserve the validity of that property. We have obtained the following three main results: (1) our verification method uses co-logic programs, and we have shown the relationship between co-logic programs and Horn mu-calculus and some applications of it. (2) We have proposed an algorithm for mining specification formulas from a sequence database of execution logs. (3) We have proposed a new method for mining patterns with quantitative constraints such as time constraints.

研究分野：　知能情報学

キーワード：　計算論理　推論アルゴリズム　プログラム推論　システム検証

１．研究開始当初の背景

ソフトウェアはますます複雑化・高度化する様々なシステムを制御し運用するための社会基盤として広く利用されており，その安全性や信頼性を保証することは喫緊の課題となっている．ソフトウェアの高信頼性を担保するために，その正当性を保証するソフトウェア設計の方法論を確立することが望まれている．ソフトウェアの妥当性確認(バリデーション)やテストという伝統的な方法では，誤りやバグを発見することはできるが，正当性を保証することはできない．従って，安全性や信頼性の向上のためには，その正当性を形式的に検証することを可能にする形式的手法に基づいたソフトウェア設計の方法論を構築することが必要となる． また，対象として，安全性が社会的にも経済的にもとりわけ重要なミッション・クリティカルなシステムであるリアクティブ・システム (reactive systems) を扱うことができるような検証の枠組が必要とされている．

従来のシステム検証の方式としては，Clarke らによるモデル検査 (model checking) とその拡張が代表的である．この方法の課題として，(i) 主に有限状態システムが対象で，本研究が対象とする無限状態システムの扱いは未だ限定的で十分に研究されていないこと，(ii) 安全性(safety)や活性(liveness)などの振舞い仕様を時相論理式(CTL や LTL など)で表現した性質が検証の対象のため，表現が本質的に命題論理式に制限されており，システムの構造的性質などの自然な表現が難しいことがある．

また，計算論理の分野でも論理プログラムを用いた検証方法が研究されてきた．例えば，Jaffar らによる時間オートマトンに対する制約論理プログラム(CLP)による方法では，振舞い仕様と構造的性質を検証の対象とし，特別な帰納法(coinduction)スキーマを用いて検証を行う．また，論理プログラムの変換を用いた検証方式では，特別な帰納法スキーマを必要としない利点があるものの，リアクティブ・システムのような動作が無限に継続するようなシステムは扱われてこなかった．

我々は先行研究で，無限項を扱う確定余論理プログラム(co-logic programs) [Gupta et al. 07]に対するプログラム変換の新しい枠組みを提案している [Seki 11, 12]. 本研究では，この結果を基にして，より広いリアクティブ・システムを扱うことを目指し，無限構造を扱えるプログラム推論技術を用いて検証を行うアプローチを採用する．これ自体が新しい研究課題であり世界的にも事例がほとんどない．

また，この方法は特別な帰納法スキーマを用いない点で Jaffar らの方法より単純で，従って実応用への有効性を持つ．実際的なシステム検証に用いるためには，先行研究の結果を発展させ，無限構造を扱う論理プログラム推論の方法論を確立し，実応用のための課題を明らかにすることが必要となる．

２．研究の目的

本研究では、複雑化・高度化するシステムやソフトウェアを対象にして、その安全性や信頼性の向上を目指し，設計・開発の正当性・妥当性の形式的検証を可能とする計算論理に基づくシステム検証のための方法論を確立することを目的としている．ソフトウェアの正当性を検証する形式的手法として，計算論理に基づく方法が有効と考えて，その分野で蓄積されてきた研究成果を可能な限り利用する．その中でも，論理プログラムに対するプログラム推論技術を用いた検証技術を中核に用いる点に特徴がある．また，プログラム推論に関する計算論理分野の知見とともに，ソフトウェア科学における形式手法の分野における知見も利用して，それらの有機的な結合を図り，新たな研究展開を促すことを目指す．

本研究の提案者は，先行研究で無限項を扱う確定余論理プログラムに対するプログラム推論の新しい枠組みを提案している．本研究では，この枠組みを基にして更に広い範囲の対象クラスの問題が扱えるように拡張して，リアクティブ･システムを対象とする無限構造を扱うプログラム推論技術を用いる検証方法を確立する．また，本研究では，システムの諸性質を検証するために必要となる仕様についても，それを自動的に導出するような枠組みについて検討する．そして，それを組み込んだシステム検証方式の実現のための課題を明らかにすることを目的とする．

３．研究の方法

本研究で採用するプログラム推論を中核に用いた検証アプローチでは，対象とするシステムの性質の検証を次のような過程で行う．対象システムとその証明すべき性質(仕様式)が与えられると，最初に，そのシステムの動作を表現する論理プログラムを構成する．また，証明すべき性質を記述する仕様式は一階述語論理式,あるいは時間論理式で表現される．一階述語論理式の場合は,それを余論理プログラムの節の形式に等価変換する．このようにして得られた余論理プログラムに対して，必要ならばプログラムの意味を保存する変換規則を繰り返し適用して，最終的に証明すべき性質を表す論理式の真理値が容易に分かる形式の論理式を導出するように変換を行う．

このようなプログラム推論を用いた方法でシステムの検証を行うためには，その基盤

となる余論理プログラムの理論的基盤を構築すること，推論規則をシステム検証に有効に適用するための方法論を確立すること，そしてその有効性をソフトウェア・システムの検証において確認することが必要となる．また，検証に用いる仕様式をどのように準備するかという課題がある．本研究では，特に以下の四つのタスクに焦点を当てて研究を行った．

(1) 検証のための論理プログラムに対するプログラム推論の設計

検証方式としてプログラム変換等の技術を中核にしたプログラム推論を用いるので，その方式設計が中心課題になる．本研究では基礎となる論理プログラムとして余論理プログラムを採用している．その特長として，従来の論理プログラムの意味論で使われている極小モデルの意味論と同時に最大モデル意味論も扱うことができて，それを利用した帰納法と余帰納法の推論を用いた検証が可能となっている．

その一方で，余論理プログラムではプログラムの意味を定義するため構文上の制約(層状化制限)が課されており表現能力が限定される．そのために，システムの様々な性質を表現した仕様式を扱う際の制約となっている．この問題を解決するために，より一般的な記述を許すホーンμ-計算 (Horn μ-calculus) のような枠組みの利用可能性を検討する．

また，ホーンμ-計算のような拡張プログラムを対象とする推論方式の定式化を検討する．特に，仕様が一階述語論理式で与えられる場合に必要となるプログラムの節本体が一階述語論理式であるような拡張されたプログラムを対象とする推論規則の定式化を進める．また，仕様式に現れる否定表現を扱うためのプログラム推論法である否定除去のようなプログラム変換手法について検討する．更に，対象システムの仕様が時間論理式で表現される場合の検証方式についても検討する．

(2) 仕様マイニングのためのパターン発見アルゴリズムの設計

検証対象のシステムが満たすべき性質を表現する仕様をユーザが正確に与えることは難しい点を考慮し，実行ログから自動的に仕様を導出する仕様マイニング（仕様発見）のアプローチを検討する．プログラムの実行ログを格納した系列データベースからのパターンマイニングにより仕様発見を行うことを目指す．仕様マイニングの基礎となっている従来研究について詳細検討をすすめ，新たな仕様マイニング・アルゴリズムを設計し，その実装を通して動特性の解析を行う．

(3) パターン発見アルゴリズムの実装方式の検討

仕様マイニングでは，通常の系列マイニングと比較して，CPU 処理時間やメモリ消費量の点で計算量的に負荷が大きいことが知られている．そのために，アルゴリズム実装上の効率化についてさまざまな方向から検討する必要がある．

特に，仕様マイニングの処理に内在する並列性に着目して，それを可能な限り利用する並列アルゴリズムの設計について検討する．また，実行ログを格納した系列データベースに含まれる系列に対して，その射影演算等の文字列処理を頻繁に行うことが必要になることに留意して，それを効率化するためのデータ構造等の設計を行い，その有効性を実験によって検証する．系列パターンマイニングの分野の従来研究で扱われているデータベースを中心にして，提案する仕様マイニング・アルゴリズムと従来方法の実験結果を蓄積する．また，本手法をプログラミング言語(Java など)で PC 上に実装する場合の課題についても検討する．

(4) 仕様表現のための量的制約導入の検討

検証対象のシステムの諸性質を表現するためには時間制約等の量的制約が必要となる．そのために，量的制約を含むパターン発見アルゴリズムの設計を行う．量的制約表現としては，先行研究で扱っている形式概念分析(Formal Concept Analysis)の分野で使われている飽和区間パターンの採用を検討する．また，区間パターンを含んだルールを効率よく発見するアルゴリズムを設計する．その有効性を，さまざまなデータベースについて実験的に検証する．

４．研究成果

本研究の主な成果として次の三点があげられる．

(1) 余論理プログラムの枠組みの拡張

先行研究で，Gupta らによる余論理プログラム(co-logic programs) に対するプログラム推論の方法として,プログラム変換規則を提案している．

① この結果を元に，余論理プログラムの枠組みを拡張してより広いクラスのプログラムを扱えるようにするために, Charatonik らのホーンμ-計算 (Horn μ-calculus) に着目した．述語の否定を計算する双対プログラムを導出する否定除去技法を用いて，ホーンμ-計算の枠組みで双対プログラムを定式化した．その結果，より広いクラスの余論理プログラムにもこのプログラム変換技法が適用可能になった．

② ホーンμ-計算が余論理プログラムの真の拡張であり，「層状化制限」という構文上の制約を必要としない枠組みであることを示した．また，ホーンμ-計算のためのプログラム推論の規則と拡張した双対プログラ

ムを示した．
③ 論理プログラムにおける主要な意味論である有礎モデルと解集合について，ホーンμプログラムとの関係を示した．特に，有礎モデルについては，双対プログラムの不動点による特徴づけを示した．この結果，推論方式の中核となる余論理プログラムの意味論についてより一般的な理論的性質が明確になった．

(2) 仕様発見アルゴリズムの提案
ソフトウェアの正当性検証に必要な仕様をユーザがあらかじめ正確に与えることは難しい問題である．そのために，実行ログから自動的に導出する仕様発見（仕様マイニング）のアプローチを採用し，以下のような結果を示した．
① プログラムの実行トレースにおける出現パターンから仕様を発見する仕様マイニングの従来研究である Lo らの NR3 (Non-Redundant Recurrent Rule Miner) と呼ばれる仕様マイニング・アルゴリズムに着目した．プログラム実行ログを格納した系列データベースから再現(recurrent)ルールというパターンを発見する仕様発見アルゴリズムについて，NR3 の実装と実験を通して，処理時間やメモリ使用量等の動特性分析を行い検討した．
② Lo らの NR3 についての検討に基づき，再現ルールの生成を効率化した仕様発見アルゴリズム LF-NR3 (Loop-fused NR3) を提案した．このアルゴリズムでは，再現ルールの生成過程にプログラム推論技術であるプログラム変換手法を適用して効率化を図った．
③ 再現ルールマイニングの効率化のためのデータ構造を提案した．再現ルールマイニングでは，実行ログを格納した系列データベース中の系列に対し，その部分系列を生成する処理を頻繁に行う．その処理を効率よく行うために，系列におけるイベントの出現箇所を記憶しておくハッシュ構造を導入して，高速化を実現した．

(3) 仕様マイニングのためのパターン発見方式の効率的な実装
再現ルールマイニングでは，通常の系列マイニングと比較して，系列データベースの射影演算や射影データベースに対するマイニングの繰り返しという負荷の大きな処理が必要とされる．その効率化のための検討をすすめ，以下のような結果を与えた．

① 再現ルール生成過程における処理の並列性に着目し，それを利用した並列アルゴリム pNR3 (parallel NR3)を提案した．またそのアルゴリズムを Java 言語により実装して，その有効性を実験により評価した．
② 再現ルールマイニングにおいて基礎となっている飽和パターンを効率良く導出するための並列アルゴリズムを提案した．またそのアルゴリズムも実装して，従来法よりも優れた性能を示すことを確認した．
③ 仕様表現に必要な時間等の量的制約も含むパターン発見アルゴリズムの設計とその実装を行った．量的制約表現として，形式概念分析の分野で使われている飽和区間パターン(closed interval patterns, CIPs)を用いてそれをルール表現に用いる枠組みを採用し，区間パターンを含むルールを効率よくマイニングする枝刈り戦略を持つアルゴリズムを設計した．また，それを実装していくつかの実験によりその効果を検証した．

余論理プログラムやその拡張であるホーンμ-計算の枠組みは，無限項を含む無限構造を自然に扱うことができるため，リアクティブ・システムを対象としてその性質を検証することが可能となった．また，検証に必要な仕様をプログラムの実行ログから発見する仕様マイニングについても，そのアルゴリズム設計の基礎となるための知見を蓄積した． これらの結果は，実応用に向けた検証システムの実現に貢献する意義がある．

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 4 件）

① H. Seki, M. Nagao, “Parallel Algorithms for Enumerating Closed Patterns from Multi-Relational Data”, Discrete Applied Mathematics, 2018, 印刷中, 査読有.
② M. Nagao, H. Seki, “An FCA Approach to Mining Quantitative Association Rules from Multi-Relational Data”, International Journal of Computational Intelligence Studies, 2018, 印刷中, 査読有.
③ H. Seki, “On Dual Programs in Co-logic Programming and the Horn μ-calculus”, Formal Aspect of Computing, Vol.29, No.3, pp. 401-421, 2017, DOI 10.1007/s00165-016-0404-0, 査読有.
④ H. Seki, “On Dual Programs in Co-logic Programming”, Logic-Based Program Synthesis and Transformation, 25th Int'l. Symp., LOPSTR2015, Revised Selected Papers, Lecture Notes in Computer Science, Springer-Verlag, LNCS 9527, pp. 1-15, 2016, 査読有.

〔学会発表〕（計 9 件）

① S.-Y. Yoon, H. Seki, “Towards Efficient Mining of Non-Redundant Recurrent Rules from a Sequence Database”, IEEE 10th Int'l. Workshop on Computational Intelligence and Applications (IWCIA2017), 2017.

② S.-Y. Yoon, H. Seki, “Parallel Mining of Non-Redundant Recurrent Rules from a Sequence Database”, 18th Int’l Symp. on Advanced Intelligent Systems (ISIS 2017), 2017.
③ H. Seki, M. Nagao, “An Efficient Java Implementation of a GA-based Miner for Relational Association Rules with Numerical Attributes”, 2017 IEEE Int'l. Conf. on Systems, Man and Cybernetics (SMC 2017), 2017.
④ M. Nagao, H. Seki, “On Mining Quantitative Association Rules from Multi-Relational Data with FCA”, IEEE 9th Int'l. Workshop on Computational Intelligence and Applications (IWCIA2016), 2016.
⑤ M. Nagao, H. Seki, Mining Correlated Association Rules from Multi-Relational Data Using FCA”, Joint 8th Int’l. Conf. on Soft Computing and Intelligent Systems and 17th Int’l Symp. on Advanced Intelligent Systems (SCIS&ISIS 2016), 2016.
⑥ M. Nagao, H. Seki, “Towards Parallel Mining of Closed Patterns from Multi-Relational Data”, IEEE 8th Int'l. Workshop on Computational Intelligence and Applications (IWCIA2015), 2015.
⑦ M. Nagao, H. Seki, “Towards Efficient Mining of Closed Patterns from Multi-Relational Data”, 11th Int'l. Conf. on Knowledge Management (ICKM), 2015.
⑧ H. Seki, “On Dual Programs in Co-logic Programming”, 25th Int'l. Symp. on Logic-Based Program Synthesis and Transformation (LOPSTR2015), 2015.

〔図書〕（計　　件）

〔産業財産権〕

○出願状況（計　　件）

名称：
発明者：
権利者：
種類：
番号：
出願年月日：
国内外の別：

○取得状況（計　　件）

名称：
発明者：
権利者：
種類：
番号：
取得年月日：
国内外の別：

〔その他〕
ホームページ等

６．研究組織
(1)研究代表者
世木 博久（SEKI Hirohisa）
名古屋工業大学・大学院工学研究科・教授
研究者番号：90242908

(2)研究分担者
（　　）

研究者番号：

(3)連携研究者
（　　）

研究者番号：

(4)研究協力者
（　　）