様式Ｃ－１９

科学研究費助成事業（科学研究費補助金）研究成果報告書

平成 24 年 6 月 11 日現在

機関番号：62615
研究種目：基盤研究(C)
研究期間：2009～2011
課題番号：21500006
研究課題名（和文）　様相論理の決定手続きとソフトウェア検証への応用
研究課題名（英文）　Decision procedures of modal logics and their application to software verification

研究代表者
田辺 良則　　（TANABE YOSHINORI　　）
国立情報学研究所・アーキテクチャ科学研究系・特任研究員
研究者番号：　60443199

研究成果の概要（和文）：様相論理を用いたポインタ操作プログラムの検証，特に停止性判定への応用を図るために必要な理論を構築した．ひとつには，クリプキ構造操作に対する最弱事前条件および最強事後条件の解明であり，もう一つは様相 μ 計算の意味論の拡張である．後者では，真偽値が min-plus 代数 N∞に値をとるような体系を構築し，この体系上でのモデル検査問題と充足可能性判定問題に解を与えた．

研究成果の概要（英文）：We built a theory for applying modal logics to verification of programs that manipulate pointers, especially for techniques to judge the termination. It includes the weakest precondition and the strongest postcondition of operations of Kripke structures, and an extension of the semantics of modal mu-calculi. In the latter, we built a semantics in which truth values are members of min-plus algebra N-infty, and gave solutions to the model-checking problem and the satisfiability judgment problem with this semantics.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
2009 年度	700,000	210,000	910,000
2010 年度	700,000	210,000	910,000
2011 年度	700,000	210,000	910,000
総　計	2,100,000	630,000	2,730,000

研究分野：総合領域
科研費の分科・細目：情報学・情報学基礎
キーワード：様相論理，決定手続き，ソフトウェア検証，モデル検査，充足可能性，様相 μ 計算，min-plus 代数

１．研究開始当初の背景

アルゴリズムやプロトコルの正しさを検証したり，誤りを発見したりする手段に，形式論理の応用である，数理的技法がある．近年では，ソースコードを直接検証する手法が広く研究されている．

ソースコード検証を可能にする技術のひとつが抽象化である．プログラムの実行状態全体は巨大な集合になり，直接解析することは困難である．適切な抽象化によって，多数の類似した実行状態（具体状態）を１つの抽象状態にまとめることができれば，解析が可能となる．抽象状態の遷移を調べるためには，自動証明器が用いられる．

特に困難な課題の一つに，リスト，木，DAG などのポインタで構成されたデータ構造を

扱うプログラム（以下，ポインタ操作プログラムと呼ぶ）に関する性質の検証がある．抽象化法を適用しようとしても，1 階述語論理など，従来の自動証明器が対象とする論理では，ポインタに関する性質がうまく表現できないからである．

開始当初，この課題に対するアプローチとして，様相論理の一種である，拡張様相μ計算と呼ばれる一群の論理を用いて，抽象化を行う枠組みを提案していた．それまでの研究で，この枠組みに基づく試作ツールを用いて，ベンチマーク的な安全性検証問題を，実用的な時間で解くことに成功した．理論的基礎となったのは，われわれはが考案した拡張様相μ計算に関する，実用的な充足可能性判定手続きである．そのような手続きがあることは従来から知られていたが，実問題に応用可能なものは存在しなかった．我々は，適用可能な論理式の範囲に，「不動点演算子の無交代性」という制限を加えることで，実装可能な手続きの構築に成功した．この手続きをベースに，検証ツールに必要な自動証明器を実現した．

２．研究の目的

上述のように，我々の手法によって，ポインタ操作プログラムの部分的正当性は検証可能となった．この研究では，これを補完する性質である停止性の検証を行うために必要な理論的な探求を行うことを目的とした．

一般に，プログラムが停止することを判定するためには，実行状態を，自然数のような整列集合や，一般には well-founded な集合への対応づけを利用する．数値を扱うプログラムは，比較的容易にそのような対応を見いだすことができるが，ポインタ操作プログラムでは，困難な場合が多い．

この課題に対する我々のアイディアは，N∞値クリプキ構造を利用するというものである．集合としての N∞は，自然数の集合に最大元 ∞ を添加したものである．ここに，2 つの演算 min と+ によって，dioid と呼ばれる代数構造が入ることから，min-plus 代数と呼ばれている．これらの演算を用いて，論理式ϕの「真偽値」[[ϕ]]を N∞の要素で与えることができる．完全な真，偽はそれぞれ 0 と ∞ であり，他の値はその中間となる．通常，拡張様相μ計算の論理式の意味論はクリプキ構造で与えられるが，真偽値を拡張することで，N∞値クリプキ構造が得られる．ポインタ操作プログラムの停止問題を，N∞値クリプキ構造における充足可能性判定問題に帰着させることができることがわかる．

以上の考察のもと，この研究課題では N∞値クリプキ構造における充足可能性判定法を解明することを目的とする．その前提として，同構造に対するモデル検査法の解明を図る．

３．研究の方法

以下の項目について研究を進めることとした．

(1) クリプキ構造に対する変形操作に関する最弱事前条件および最強事後条件の解明

(2) 様相μ計算の N∞値意味論の確定と，それと同値なゲーム意味論の決定

(3) N∞値意味論のもとでの，モデル検査問題の解法

(4) N∞値意味論のもとでの，充足可能性問題の解法

まず，(1)により前向き推論と後ろ向き推論の両方において，プログラムの検証を様相 mu 計算論理式の充足可能性問題に帰着させることが可能となる．次に(2)で，本研究課題の主対象である N∞値意味論を確定する．ここでは，後に必要となるゲーム意味論との同値性が重要となる．充足可能性問題を解く前提として，まずモデル検査問題を解くことが必要になるため，これを(3)で実施する．(4)の充足可能性判定問題では，まず，論理式の変換によって従来の 2 値論理式の充足可能性判定問題に帰着するアプローチをとり，その後に直接の判定方法を検討する．

４．研究成果

(1) クリプキ構造に対する変形操作に関する最弱事前条件および最強事後条件の解明

我々の検証手法はその基本において，クリプキ構造に対する操作に関するものである．ポインタ操作プログラムの検証についても，プログラムが扱う対象をクリプキ構造とみなすことによって，応用することが可能となる．この研究項目では，必要となるクリプキ構造に対する変形操作を確定し，それらに対する最弱事前条件および最強事後条件を確定する．

操作前のクリプキ構造に関する性質を現す論理式と，操作後の性質を現す論理式とを結びつけるのがこれらの条件となる．操作後の論理式をϕ，操作を op，最弱事前条件を wp(ϕ,op)，操作前の論理式をϕで現すとすると，$\phi \rightarrow$wp(ϕ，op) が成立することを確認することで，操作前のϕから操作後のϕが導かれることがわかる．言い換えれば，$\phi \rightarrow$wp(ϕ，op) の恒真性を判断すればよいが，これは，充足可能性の双対である．このようにして最弱事前条件を確定させることは，充足可能性判定問題を検証に応用するために必要な過程である．以上は後ろ向き推論を想定しての議論であるが，前向き推論を想定すれば，最強事後条件が同様の役割を果たすことになる．

しかしながら，様相μ計算の論理式ϕに対し，その最弱事前条件や最強事後条件が再び様相μ計算の論理式になるかどうかは，全く明らかではない．実際，様相μ計算には多数の変種があり，応用先に応じて使い分けるのであるが，変種によっては最弱事前条件がその言語では記述できないものがあることがわかった．我々がプログラム検証への応用を考えている変種に絞って検討をすすめた結果，最弱事前条件についてはその変種の論理式でちょうど記述できることが判明した．一方，最強事後条件に関してはそのような単純な事態ではないことがわかった．しかし，検証に必要な範囲で十分な結果を得ることができた．具体的には，クリプキ構造たちに適切な同値関係を導入することにより，その同値類の適当な要素上で，最強事後条件として振る舞うようなものを見つけて来られる，という結果を得た．この結果を用いれば上述のような前向き推論が可能となる．

(2) 様相μ計算のN∞値意味論の確定と，それと同値なゲーム意味論の決定

様相μ計算の各論理式ϕに対して，その真偽値なるN∞値[[ϕ]]を定義することで意味論が定義される．N∞に入っている代数構造(min-plus代数)を参照して，選言を min で，連言を plus で解釈するというのが基本的なアイディアである．(結果として，[[true]] = 0, [[false]] = ∞となる．) 不動点演算子を含むほとんどの演算子の処理は自然に定まるが，否定演算子の処理だけが自明ではない．ここでは，ハイティング代数等における処理を参考に，まず含意演算子の処理を定め，それを用いて[[¬ϕ]] = [[ϕ→false]]と定めることとした．当然の帰結として，古典論理におけるトートロジーの真偽値がすべて[[true]] (= 0) と等しくなるわけではないということになる．

意味論を確定させた後に重要となるのはゲーム表現である．2値の場合でも，様相μ計算においては，論理式の構成に関する帰納法があまりうまく働かないケースが多くある．これは，変数を束縛する不動点演算子と，その変数の出現がある意味同一視されるために，論理式の意味に循環的な部分が出てくるのが原因である．（元の意味論と同値な）ゲーム表現を使うことによってこの問題は回避され，見通しの良い議論が可能になる．また，具体的に与えられた論理式とクリプキ構造上の状態から，その論理式のその状態における成否を判断する際にも，ゲーム表現を使用する方が簡単になることが多い．これらの観点から，上で定義した意味論と同値になるゲーム表現を見いだすことは重要な事項である．

まず最初に，ゲームでは Player1 か Player2 かに必勝法があるかどうかの2値の判断しかできないのに，論理式の値としてたくさんの値を取り得るという問題がある．このため，アリーナを論理式ϕとクリプキ構造の状態 s のペア（ϕ,s)でなく，N∞の要素 n も加えた(ϕ,s,n)とした．この点が Player1 の必勝領域に属することが，[[ϕ]](s) <= n と同値となるようにゲームを設計した．

さて，2値の場合に比較して，本質的に難しいのは否定の扱いである．2値の場合には，すべての否定記号を中に押し込んでリテラルとして扱うことが可能であったが，二重否定がキャンセルできない今回の体系ではその手法は使えない．そこで，否定記号が現れるたびに Player1 と Player2 の役割を交代するという議論を行った．さらに，優先度の与えかたにも工夫が必要であった．2値の場合のように単純にμには奇数を，νには偶数を割り当てることはできず，μ/νのちがい，n の有限/無限の差，および，変数の束縛地点と変数の出現地点の間における否定記号の出現の有無の要因によって，優先度割り当てを決定することで，正しいゲームを設定することができた．

(3) N∞値意味論のもとでの，モデル検査問題の解法

我々の目標は充足可能性判定問題であるが，その解明の前提としてモデル検査問題に取り組んだ．ここでは，モデル検査問題とは，具体的に与えられた論理式ϕとクリプキ構造の状態 s に対して，[[ϕ]](s)を求める手続きを与える，というものである．状態を変数と見る標準的な変換によって，これは次のように言い換えられる：2項の min, plus, implication, および最小，最大不動点をとる演算の組み合わせで定義される N∞上の多変数関数の値を具体的に求める手順を与えること．

Implication を除外すると，問題は簡単になり，2値のμ計算の場合とほぼ同様に求めることが可能である．そこで implication を含めても計算が可能となるように拡張を図り，以下のように成功をみた．

まず，N∞を，Z∞に拡張し，min だけでなく max も演算子に加え，対称性を良くした．((+∞)+(-∞)をどう定義するかという問題があるが，+演算子も2種類用意する．上記の結果が+∞となるものと-∞となる2種類である）当然焦点となるのは不動点演算子である．対称にしたので，LFP(最小不動点)のみを考える．2値の場合と違って，繰り返しの計算をしたのでは無限に計算が繰り返される可能性があるので，加速する手段を考案しなければならない．まず LFP が簡単に計算が可能な関数のクラス “stepless” を定義した．これは直観的には「上に凸」な関数な

のだが，+∞と-∞のところも都合良く振る舞うようなクラスである．このクラスでは，単純な繰り返しによってLFPを求めることができる．一般の関数については，stepless関数で下から近似する，という方針をとった．近似の方法が微妙であり，ナイーブな近似方法だと近似自体が無限回の過程を要することになるが，うまく近似を行うことにより，全体の計算回数を有限に抑えることが可能となる．

(4) N∞値意味論のもとでの，充足可能性判定問題の解法

モデル検査の場合と同様に，N∞値の場合には，充足可能性判定問題の定義も自明ではない．ここでは，与えられた論理式φに対し，クリプキ構造とその状態sで，[[φ]](s)=0となるものが存在するかどうかを判定する手続き，という定義を採用する．

基本的なアプローチは，φを2値の論理式に変換し，2値論理式についてよく知られた充足可能性判定手続きを利用しよう，というものである．実際には，4つの変換関数trZer(・)，trInf(・)，trPos(・)，trFin(・)を考える．たとえばtrZer(φ)は，[[φ]]=0を「現す」2値論理式である．より正確には，「[[φ]](s)=0となるsが存在する」ことと「trZer(φ)が充足可能である」ことが同値となる．残りは，各々[[φ]]が無限大，正，有限であることを「現し」ている．

変換に当たっては，各演算子ごとにいろいろな工夫が必要である．たとえば□演算子について，[[□φ]](s) = ∞ となるのは，隣接する状態が無限にあってそこで[[φ]]が正の値をとるケースと，隣接するどれか1つの状態で[[φ]]が無限大になるケースとがあり，この2つは本質的に異なる．このため，2値の論理式の側で様相を2つ(1と∞)用意し，trInf(□φ) = ⟨1⟩trInf(φ) ∨ ⟨∞⟩trPos(φ) のように定義する．このほか，不動点演算子についても，無限個の有限の値が∞に収束していくケースをうまく扱えるような変換定義が必要となる．

このように定義した変換規則が正しいことの証明には，項目(2)で確立したゲーム表現が重要な役割を演じた．

(5) 今後の展開に向けて

項目(4)で得られた充足可能性判定法は，計算量的には不利であることと，判定可能な値が0と∞のみであることであるという問題点がある．これを解決するため，2値論理式を経由しない直接の判定手続きの構築にも取り組んだ．昨年度までの検討では，タブローノードに意味を与えても展開がうまくできないという課題を抱えていた．これを，論理式の値が「k以下」，「kより大きい」という意味を与えうるようにタブローノードの構成を拡張することによって，すべての論理記号に関する展開をうまく機能させることに成功している．もう一つの課題であった優先度関数の定義に関しても，ローカルに優先度を定義するのではなく，繰り返して現れるノード間に，否定に関する遷移が存在するかどうかに依存して優先度を変化させる方法を導入することで，充足可能性と整合する定義を与えることができた．本内容は発表準備中である．

また，本研究の応用先として，定理証明支援系への組み込みに取り組んだ．N∞値が減少する列になることを示すことで，プログラムが停止することを示す枠組みである．本年度は，証明支援系CoqでN∞(min-plus代数)に関するライブラリを作成し，システム記述の際に取り込めるようにした．この上でモデル検査アルゴリズムの停止性証明を行うことを目指した実装を行った．また，ソフトウェアモデル検査アルゴリズムの証明を本手法で行うための準備を開始している．

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 Kosuke Ono, Yoichi Hirai, Masami Hagiya and Yoshinori Tanabe: Using Coq in Specification and Program Extraction of Hadoop MapReduce Applications, Proceedings of 9th International Conference on Software Engineering and Formal Methods (SEFM 2011), 査読有. LNCS-Vol.7041, 2011, pp.350-365.

② Watcharin Leungwattanakit, Cyrille Artho, Masami Hagiya, Yoshinori Tanabe, and Mitsuharu Yamamoto: Model Checking Distributed Systems by Combining Caching and Process Checkpointing, Proceedings of 26th IEEE/ACM International Conference on Automated Software Engineering (ASE), 査読有. 2011, pp.103-112.

③ Alexis Goyet, Masami Hagiya, Yoshinori Tanabe: Decidability and Undecidability Results of Modal μ -calculi with N-infinity Semantics. Proceedings of 17th Internaional Workshop of Logic, Language, Information and Computation (WoLLIC2010). 査読有. 2010, pp.148-160.

④ Dai Ikarashi, Yoshinori Tanabe, Koki Nishizawa, Masami Hagiya: Modal μ -calculus on min-plus algebra N-infinity, コンピュータソフトウェア,査読有,Vol.27, No.3, 2010, pp.99-113.

⑤ Masami Hagiya, Yoshinori Tanabe:

Fixed-point Computations over Functions on Integers with Operations Min, Max and Plus, Proceedings of 6th Workshop on Fixed Points in Computer Science (FICS2009), 査読有, 2009, pp.108-115.
⑥ Yoshinori Tanabe, Toshifusa Sekizawa, Yoshifumi Yuasa, Koichi Takahashi: Pre- and Post-conditions Expressed in Variants of the Modal μ -calculus, IEICE Transactions on Information and Systems, 査読有, Vol. E92-D, 2009, pp.995-1002.

〔学会発表〕(計 5 件)
① 姜 帆(登壇), 田辺 良則, 本位田 真一: Coq を使用した MapReduce アプリケーションの検証と Scala コードの抽出,第 14 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 (PPL2012), 2012-03-09. 和歌山県白浜町旅館むさし.
② 田辺 良則(登壇), Cyriile Artho, Watcharin Leungwattanakit, 山本 光晴, 萩谷 昌己: ネットワークアプリケーションのマスター・スレーブ方式モデル検査アルゴリズムについて.日本ソフトウェア科学会第 28 回大会, 2012-09-29. 沖縄産業支援センター
③ Yoshinori Tanabe(登壇), Vinh Cuong Tran, Masami Hagiya: Toward Liveness Verification in Java Pathfinder, 第 6 回ディペンダブルシステムシンポジウム (dss2009), 2009.12.14. 大阪大学大学院
④ Masami Hagiya, Yoshinori Tanabe(登壇): Games and Natural Number-valued Semantics of the Modal μ-calculus. 日本ソフトウェア科学会第 26 回大会. 2009-09-16. 島根大学
⑤ Alexis Goyet, Masami Hagiya, Yoshinori Tanabe(登 壇): Decidability and Undecidability Results of Modal μ -calculi with N-infinity Semantics. 情報処理学会プログラミング研究会(PRO). 2009-06-08. 東京工業大学

６．研究組織
(1)研究代表者
田辺 良則 (TANABE YOSHINORI)
国立情報学研究所・アーキテクチャ科学研究系・特任研究員
研究者番号：60443199