

審査の結果の要旨

氏名 フレデリクセン・カール・クリスチャン

本研究は、「サイズ変化停止性」を、2つのプログラム解析の枠組みである、多項式実行時間分析と活性検証に適用することにより、従来より正確でかつ効率的な分析手法を実現している。サイズ変化停止性とは、プログラムの実行において、パラメータの大きさが減少するかどうかの情報を、グラフ(サイズ変化グラフと呼ぶ)として抽出し、そのグラフを分析することにより、プログラムの停止性を判定する枠組みである。多項式実行時間分析のために、サイズ変化グラフに対し、実行時間に影響する3つの性質の判定を行うことにより、従来より多くのプログラムに対し、多項式時間で停止することを判定可能としている。また、本研究は、既存の活性検証手法を拡張することにより、より効率的な活性検証を可能としている。

論文は、以下の4つの章から構成されている。

第1章では、本論文の研究背景として、プログラムの停止性判定手法の1つであるサイズ変化停止性の概要と多項式実行時間分析と活性検証への適用方法の概要を説明し、さらに本論文の研究の意義について次に説明している。まず多項式時間計算可能性については、従来、Bellantoni と Cook が確立した、プログラムの形式的パターンによる判定手法が、最も有効なものであったが、本論文では、独自の手法により、さらに多くのプログラムに対し正しく判定することを可能にした。その手法では、サイズ変化グラフの分析により、プログラムの実行時間の上限に影響する3つの性質を判定し、それらの判定結果を用いて多項式時間で停止性を判定する。活性検証については、既存の手法である、遷移述語抽象化による公平停止性の検証手法を拡張し、サイズ変化グラフを分割し、部分毎に検証を行うことにより、より効率的な検証を可能とした。

第2章では、本論文の手法が判定可能なプログラムの構文と意味論を提示し、プログラムの実行時間の上限に影響する3つの性質、およびそれらの性質の判定を通してプログラムの多項式時間計算可能性の判定アルゴリズムの方針を説明している。3つの性質は、関数呼出しの深さ、非線形呼び出し(関数定義における同一の関数の複数回の呼出し)のパラメータの性質、および入れ子呼出し(関数定義における被定義関数自身の呼出し)のパラメータの性質である。次に、本論文の手法の基盤となるサイズ変化停止性分析手法の詳細を説明し、サイズ変化グラフの分析を利用して、その手法をどのように拡張して、前述の3つの性質を判定するかを述べている。そして、それらの判定を組み合わせたアルゴリズムを紹介し、プログラムが Bellantoni-Cook の手法によって多項式時間で停止することが判定できれば、本論文の手法でも判定できることの証明を行い、逆に本論文の手法で判定できても、Bellantoni-Cook の手法ではできない例を示すことにより、本論文の手法の方が優れていることを示している。次に、98件のプログラムに対し本論文の手法を適用した実験結果を示し、正しく判定できなかった例に対して、その原因として、条件分岐を考慮していないことや、分割統治法の不十分さなどがあることを挙げ、それぞれについて分析している。最後に、本章のまとめ、失敗例の原因の改善を含めた将来課題、および関連研究を述べている。関連研究としては、Bellantoni-Cook の方法を拡張する研究(いずれも本論文の手法より適用範囲が狭い)、および他のサイズ変化停止性に関する研究(いずれも多項式時間停止性は分析できない)を取り上げている。

第3章は、既存の手法である、遷移述語抽象化による公平停止性、すなわち、ある種の公平性が成り立つ実行トレースがそのうち停止する、という性質の検証手法を拡張した、より効率的な検証手法を提案している章である。本章では、まず対象とする非決定的逐次プログラムのための言語の構文と意味論を紹介し、Podelski と Rybalchenko による既存の手法を説明している。次に、本論文における、その手法の拡張部分を詳述している。その拡張では、まず元のプログラムに対し、無限に続く振舞いを表すサイズ変化グラフを構築し、強連結コンポーネントと呼ばれる部分に分割して、各コンポーネントに対し、遷移述語抽象化を行いながら公平停止性を検証する。なお、各コンポーネントは bottom-up order と呼ばれる順序がつけられている。抽象化と検証は、その順序で行い、既に行った分で得られた、変数値の制約から、次の検証に必要な制約を、線形計画法

で求める。その後本論文で拡張した検証アルゴリズムの正しさの証明と、1つの例に適用した実験結果を示すことにより、アルゴリズムの有効性を主張している。次に、提案した手法を並列プログラムに拡張するために、まず並列プログラムのための言語の構文と意味論を示している。そして、並列プログラムへの適用に必要な、検証手法の拡張部分を説明し、拡張した検証アルゴリズムを提示し、その正しさを証明している。最後に、本章のまとめと、提案した手法の問題点として、検証不可能なプログラムの存在と計算量について述べ、それらの問題点の解決方針を将来課題として示している。

第4章では、本論文の結論として、提案した手法をまとめ、サイズ変化停止性の2つの応用を示し、特に、部分的計算に基づく手法の特徴により、分析の正確さと効率を向上することができたことを主張している。また将来の課題として、部分的計算に基づく手法を、他のプログラム解析手法に適用すること、第2章の手法の問題点の解決、および既存の遷移述語抽象化による手法を用いた検証ツール“Terminator”に、第3章の手法をこのツールに取り入れて改善することを挙げている。

以上のように本論文は、サイズ変化停止性を、多項式実行時間分析と活性検証に適用するにあたり、前者においては、サイズ変化停止性分析において、プログラムの実行時間に影響する3つの性質の判定を行うという独創的な手法、また後者においては、サイズ変化グラフを分割し、部分毎に検証を行うという独創的な手法を提案するとともに、既存の手法より、分析や検証の効率や正確さが向上していることを、理論的分析と実験で実証したものであり、審査委員会は、その独創性、有効性は、博士号に十分値するものと判断した。

よって本論文は博士（情報理工学）の学位請求論文として合格と認められる。