平成 19 年度
実績報告

「実用化を目指した組込みシステム用ディペンダブル・オペレーティングシステム」
平成 18 年度採択研究代表者

前田　俊行

東京大学　大学院情報理工学系研究科　教授

ディペンダブルシステムソフトウェア構築技術に関する研究

1. 研究実施の概要

計算機（携帯電話、PDA、PC 等)、そして計算機を相互に接続するネットワークは、既に広く一般に普及し、もはや生活に欠かすことのできない社会基盤となった。このため、これらの計算機上で動作するソフトウェアの安全性・信頼性を確保することが非常に重要な課題である、ということが広く認識されるようになった。しかしながら、計算機システムで最も基礎的で重要なソフトウェアであるシステムソフトウェア（オペレーティングシステムなど）は、依然として 30 年以上前に考案された C 言語や、アセンブリ言語等の安全でない言語を用いて構築されており、その安全性・信頼性には大きな疑いがあると言わざるを得ない。また現実に多くの問題、例えばシステムの異常停止等の問題が発生している。

これに対し本研究では、近年目覚しい発展を遂げた静的プログラム解析技術（プログラムを数学的理論に基づいて解析することで、プログラムを実行することなく、その性質を知る技術)、特に型理論とモデル検査理論に基づいた、システムソフトウェアの構築・検証技術を実現することで、システムソフトウェアの高信頼化に寄与することを目指す。より具体的には、(1) システムソフトウェアを記述可能な型付きアセンブリ言語の設計・実装 (2) C 言語から型付きアセンブリ言語への変換器の設計・実装 (3) モデル検査技法に基づくシステムソフトウェアの解析、の 3 つについて研究を行う。

本年度は前年度より引き続き、割込みやマルチコア・SMP に対応した型付きアセンブリ言語の設計や、ポインタ演算やキャスト・関数ポインタなどの C 言語の機能を実現可能な型付きアセンブリ言語の設計・プロトタイプ実装、C 言語から型付きアセンブリ言語への変換器の設計・プロトタイプ実装、クラスタなどの並列計算機上でモデル検査を行うための調査・検討・試験実装などを行った。次年度は、本年度までの調査・検討結果や理論設計、試験実装をもとにして、実際にシステムソフトウェアの一部を検証可能な程度に実用的な検証器を実装することを目指す。

2. 研究実施内容

(1) システムソフトウェアを記述可能な型付きアセンブリ言語の設計・実装

本年度は、ハードウェア割込みやマルチコア・SMPに対応した型付きアセンブリ言語の理論設計を行った。ハードウェア割込みに対しては、対応する割込みハンドラが、プログラムの任意の実行ポイントで実行され得ると仮定して型検査を行う。また、マルチコア・SMP に関しては、複数コア(CPU) 間で共有され得るメモリ領域に対して、型を変更するようなメモリ操作を基本的に禁止して、CPUハードウェアが提供するatomicityを保った命令のみ、命令の開始から終了までの間、一時的に型を変更することを許すことで、共有され得るメモリ領域の型安全性を保つ。また、マルチコア・SMP 特有のメモリコンシステンシの問題に対しては、存在型 (existential type) の理論を応用することで、幾つかの既存の同期プリミティブを実現できることを確かめた。

また、型付きアセンブリ言語の型検査器を、異なる CPU アーキテクチャごとに一から作成するのは非効率であるため、CPU アーキテクチャごとのアセンブリ言語から、ある共通の中間表現へ変換し、その中間表現上で型検査を行う仕組みを検討した。さらに、中間表現上で行った型検査の結果が、CPU アーキテクチャごとに行った型検査の結果と同等になるための条件についての考察を行った。

また、柔軟なメモリ管理の記述をプログラマに許しつつもメモリ安全性を損なわず、かつ、C 言語からのコンパイルも可能な型付きアセンブリ言語の実現を目指して、本年度は、ポインタ間のエイリアス情報を追跡するエイリアス型システムを、分離論理 (separation logic) で用いられている分離含意 (separating implication) で拡張することで、従来のエイリアス型システムでは型付けできなかったループ構造の記述が可能になることを示した。

(2) C 言語から型付きアセンブリ言語への変換器の設計・実装

本年度は、前年度から継続して行った C 言語からの変換が可能な型付きアセンブリ言語の設計を完成させ、また型検査器のプロトタイプ実装を行った。具体的には、従来の型付きアセンブリ言語を依存型の理論に基づいて改良することで、ポインタ演算やキャスト、構造体、共用体、可変長引数関数、関数ポインタ等の C 言語特有の複雑な言語機構を型付きアセンブリ言語上で実現した。また、実際に C 言語で記述されたプログラムを型付きアセンブリ言語へ変換する変換器のプロトタイプ実装を行った。具体的には、前年度までに検討した安全化コンパイラ (CCured や Fail-Safe C など) の手法を応用・改良して、型安全性が保証された動的検査コードを挿入する手法を考案した。また、実際に幾つかのプログラムをこの変換器を用いて型付きアセンブリ言語へ変換し、有用性・問題点の検討を行った。

また本年度は、C 言語プログラムのコンパイルに広く用いられているコンパイラ GCC の中間言語 RTL に対する型システムの検討も行った。具体的には、C 言語のコンパイル上本質的な部分を RTL から抜き出して簡略化した言語である Tiny RTL を定義し、これに対して型システムを与えた。

(3) モデル検査技法に基づくシステムソフトウェアの解析

本年度は、前年度の調査結果に基づき、モデル検査器のプロトタイプ実装を行った。モデル検査器は、主なプログラム要素として、C 言語構文解析、意味解析、充足性検査器、抽象実行処理の四つの部分から構成される。本年度は抽象実行以外の基本要素である構文解析、意味解析、充足性検査器を作成した。抽象実行部分は現在実装中である。抽象実行は検査器の中心であり、検証精度や検証時間といった検証能力に関わるため、様々な手法について調査・検討を行い、その結果、C 言語で記述された大規模プログラムの検査に対して現状で実用可能性が高いと考えられるブール値プログラムへの変換に基づく手法を、今回のプロトタイプ実装では採用した。また、モデル検査器は処理量が大きく大規模プログラムを扱うためには並列処理が必須であり、並列化が可能な実装を行っている。なお、プロトタイプ実装に際して、既存の検査ツールを流用することも検討したが、既存ツールはモデル検査の並列化や型検査との組み合わせなどを考慮せずに実装されており、また継続的に実用するには品質が低いと考えられる等、実用的な検査器を作成するという我々の目的にそぐわないことが判明したため、自ら実装することとした。

来年度は、抽象実行について実装を行うとともに、様々な抽象実行方式について引き続き研究を継続する。また、実際のプログラムに対してモデル検査を試行しツール群の改良を進める。

3. 研究実施体制

(1)東京大学グループ

①研究分担グループ長:前田　俊行(東京大学、教授)

②研究項目

(1) システムソフトウェアを記述可能な型付きアセンブリ言語の設計・実装

(2) C 言語から型付きアセンブリ言語への変換器の設計・実装

(3) モデル検査技法に基づくシステムソフトウェアの解析