氏　　名	NGUYEN, Thuy Thi
学位の種類	博士（情報科学）
学位記番号	博情第 494 号
学位授与年月日	令和 5 年 3 月 24 日
論文題目	PROGRAM VERIFICATION FOR MICROCONTROLLER-SPECIFIC CODING RULES
論文審査委員	青木利晃　北陸先端科学技術大学院大学　教授
	田中清史　同上　教授
	石井大輔　同上　准教授
	増原英彦　東京工業大学　教授
	石尾隆　奈良先端科学技術大学院大学　准教授

論文の内容の要旨

Microcontrollers are small computers specially designed for embedded systems. Since they were introduced, microcontrollers have been used in a wide range of applications, including mission-critical applications such as satellite rockets, automotive systems, and medical devices. Hence, it is essential to ensure the reliability of microcontroller-based applications. When developing these applications, hardware-dependent features must be considered in addition to standard C language features. For hardware-dependent features, a hardware manual is provided for each microcontroller. In this manual, information that requires special attention as coding rules are emphasized, along with the explanation of the features of the hardware. Currently, the process of verifying these coding rules is performed manually, as no single tool can directly handle this task. The reason is that the coding rules are non-standard while existing verification tools often support standard coding rules only. Verifying the coding rules is time-consuming and laborious as both the volume of the embedded source code (e.g., thousands of lines of code) and the number of coding rules are often large (e.g., the hardware manual of a popular microcontroller contains 2415 pages and 492 fragments that describe notes or cautions).

Several researchers tried to handle this task. In general, they tended to create new tools or extend existing tools. Implementing new tools or extensions is a heavy and inflexible solution. Additionally, new hardware models are frequently introduced. It is impractical to introduce one verification tool for each model.

This research aims to automate these coding rules' verification processes by proposing a flexible approach. Specifically, we proposed a verification framework that utilizes advanced techniques in program analysis and model-driven engineering. Firstly, the program analysis techniques (i.e., pattern matching, abstract interpretation-based static program analysis, bounded model checking, and counterexample-guided abstract refinement) are combined to analyze C programs effectively. Secondly, heuristic-based

natural language processing techniques are employed to analyze the hardware manual and extract hardware knowledge. Thirdly, model-driven engineering techniques are employed for comprehensively modeling the hardware, compiler, and source code of microcontroller-based systems. Finally, model querying techniques enable flexibly verifying the system against the target coding rules.

The approach was evaluated by applying to handle a benchmark source code and an industrial source code. The experiment with the benchmark source code showed that the approach is feasible in verifying microcontroller-based systems against the register-access coding rules. Although the benchmark was a small-size source code only, the source code represented different ways for violations to occur. The approach analyzed the source code successfully and detected all expected violations of the target register-access coding rules. We even find violations that senior developers miss. The experiment with the industrial source code showed that our verification framework was applicable to a real product. The precision in this experiment was 0.8, as two false warnings were detected. The recall was 1, as all expected violations were found.

This research contributes a practical solution for an important problem in the industry. Formal verification theory is highly established through a long history of development. Many methods proposed in academics could work well with small benchmark source code, but not many are applicable to industrial applications. The industrial setting is much more complex in comparison with laboratory environments. Among needs in the industry, verifying systems against microcontroller-specific coding rules is a heavy yet critical task. There is an emergency call for automated solutions. In response, our work is expected to be a practical and effective solution to judge the conformance of the coding rules and a tool for other tasks like source code understanding. The proposed verification framework promises to reduce the huge number of manual tasks in the current verification process in practice.

Keywords: Microcontroller-specific coding rules, C programs, Program analysis, Knowledge modeling, Program verification

論文審査の結果の要旨

本博士論文では，C 言語のプログラムがマイクロコントローラ（以下，MCU と略す）特有の制約を満たしているかどうか検証する手法を提案している．近年の自動車の高度な電子化により，多くの MCU が用いられるようになってきた．それに伴い，MCU 自体の性能も向上し高度な制御を実現できるようになった一方，そのためのソフトウェアは劇的に複雑化している．車載システムでは，高い品質が要求されるため，MISRA C や CERT C といった，コーディング標準が存在する．MCU 向けソフトウェア開発では，これらの規約に加えて，MCU 特有の制約を満たすことを確認しなければならない．コーディング標準は一般的な規約であるため，その検証を支援するプログラム解析ツールが提供されているが，MCU 特有の制約に関しては，そのようなツールが十分に提供されていないのが現状である．制約が MCU 依存であり，かつ，数が多いためである．

本博士論文では，複数の理論を組み合わせて，そのような現実的問題を解決する手法を提案している．提案手法では，問題解決のために応用可能かつ有効に働く理論を見極め，自然言語処理，文法検査器，抽象解釈，モデル検査，CEGAR(Counter Example Guided Abstraction Refinement)，モデル駆動開発技術を組み合わせている．既存研究では，新たな制約を取り扱えるよう既存ツールを拡張したり，新規ツールを開発ものが多い．一方，本研究では，既存ツールや理論を拡張するのではなく，それらを組み合わせて新たな制約を取り扱う手法を提案している．これは，MCU のバリエーション，および，制約の数が多いことに対処するためである．また，新たなプログラム解析手法や，より効率的な手法を柔軟に取り込むことができるといった特徴もある．これが本研究の新規な点であり，かつ，独創的な点である．本分野の理論の現状を踏まえ，現実的な問題を工学的に解決する実践的な手法であると言える．提案手法の評価では，ベンチマークプログラムだけでなく，実際の車載システムのプログラムに適用することに成功している．プログラム解析を含む形式手法は，一筋縄では実践が行えないため，その実践のための理論や技術も学術的な研究の対象となっている．本博士論文では，実践的な手法の提案と実際の実践事例の両方を示すことができており，この成果は，形式手法の分野において，高く評価できる．

以上，本博士論文は，計算機科学における理論を組み合わせて実践応用するための手法を提案しており，学術的にも工業的にも貢献するところが大きい．よって，博士(情報科学)の学位論文として十分に価値があるものと認めた．