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４日現在

機関番号：13302
研究種目：若手研究（Ａ）
研究期間：2008～2011
課題番号：20680001
研究課題名（和文）　高度な並行・並列組込みソフトウェアの検証法に関する研究

研究課題名（英文）　Research on the verification of highly parallel and concurrent embedded software

研究代表者
青木 利晃（AOKI TOSHIAKI）
北陸先端科学技術大学院大学・情報科学研究科研究科・准教授
研究者番号：20313702

研究成果の概要（和文）：
本研究では，スケジューリングを伴う並行・並列ソフトウェアと，スケジューリングを提供するリアルタイムオペレーティングシステム(RTOS)を対象とした．成果としては，前者に関しては，実時間を含む振る舞いを検証するためのアルゴリズムおよびツールを提案し，後者に関しては，RTOS の設計と実装を検証する手法およびツールを提案し，実際に使われている RTOS の検証も行った．これにより，現実的なセッティングで，モデル検査に基づいた手法の提案に成功し，実際に，現実問題に適用できることがわかった．

研究成果の概要（英文）：
We focus on parallel/concurrent software which is controlled by real-time operating system(RTOS) and RTOS itself. We have proposed an algorithm and tool to verify the behavior of parallel/concurrent software which contains scheduling by RTOS and real-time for the former. For the latter, we have proposed a method and tools to verify the design and implementation of RTOS. In addition, we have applied those method and tools to RTOS products. We succeeded in proposing verification methods based on model checking in practical settings and conducted that they are applicable to practical software products.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
2008 年度	6,200,000	1,860,000	8,060,000
2009 年度	4,900,000	1,470,000	6,370,000
2010 年度	4,300,000	1,290,000	5,590,000
2011 年度	3,600,000	1,080,000	4,680,000
総　計	19,000,000	5,700,000	24,700,000

研究分野：形式手法，形式検証，ソフトウェア工学，ソフトウェア科学
科研費の分科・細目：情報学・ソフトウェア
キーワード：形式手法，形式検証，モデル検査

１．研究開始当初の背景

今日の機器ではソフトウェアによる柔軟な制御や多様性が必要であり，あらゆる機器にソフトウェアが組み込まれるようになってきた．このようなソフトウェアのことを組込みソフトウェアと呼ぶ．組込みソフトウェアは，デバイスなどの制御などを行うため，実行が並行化された，多重割り込みソフトウェアや，リアルタイムオペレーティングシステム(以下，RTOS と略す)上にマルチタスクソフトウェアとして実装される場合が多い．従来はシングルプロセッサ上に実装されていたが，最近では，携帯電話や家電機器に見られるように，複数の CPU やマルチコア CPU を用

いて高度に並行・並列化されるようになってきた．さらには，それらを横断的に用いて，並行性がヘテロに組み合わされたソフトウェアも出現し始めた．このようなソフトウェアの検証は非常に難しく，実際，その信頼性の低下が著しい．
現在，組込みソフトウェア開発現場では，開発ソフトウェアの信頼性を保証する検証作業として，主に，開発ドキュメントを人手により確認するレビュー手法や，ソフトウェアの実装後に動作させながらチェックを行うテスト手法を用いている．しかしながら，これらの手法では，高度に並行・並列化された振る舞いの検証は困難である．非常に多くの動作の組み合わせを調べる必要があり，また，誤りのうち再現性が無いものが多いためである．そこで，本研究では，組込みソフトウェアの検証に，モデル検査手法を用いる．モデル検査手法では，並行・並列動作を効率的に自動チェックできるため，組込みソフトウェアの検証に適していると言われている．実際，いくらかの事例が発表され始めている．

２．研究の目的
本研究課題では，複雑な並行・並列性が取り扱えるようモデル検査手法を拡張し，現実的な組込みソフトウェア検証する手法を提案する．現状のモデル検査ツールでは，複雑な並行・並列動作を検証することは困難である．RTOS 上に実装されたマルチタスクソフトウェアでは，優先度，プロセスの実行停止・再開，共有資源のロックなどの概念を用いて，並行・並列動作が複雑に制御されている．タイマなどを用いて周期的に実行される場合も多い．現在提案されているモデル検査ツールでは，このような複雑な並行処理を直接的に取り扱うことはできない．また，RTOS 自体も並行・並列化された CPU 上に実装されているため，その検証は非常に困難である．そこで，本研究課題では，以上のような複雑な並行・並列動作を含むソフトウェアを対象として，モデル検査に基づいた検証手法を提案する．

３．研究の方法
組込みソフトウェアは，一般に，基本ソフトウェアであるリアルタイムオペレーティングシステム(RTOS)と，その上で動作するアプリケーションソフトウェアにより構成される．前者は，主に，組込みソフトウェアで重要となる実時間性を保証するために，並行・並列動作するソフトウェアの実行を制御するスケジューリングの仕組みを提供している．後者は，RTOS により提供されているサービスを用いて，開発対象となる機能を実現する．そこで，本研究では，これらの両方を対象として，検証手法を提案した．
(1)RTOS の検証
本研究では，車載ソフトウェアで用いられる RTOS である OSEK/VDX を実際の事例として用いた．ソフトウェア開発では，一般に，「要求分析」,「設計」,「実装」といった工程に分けられる．それぞれの工程では，単に，ドキュメントやコードを作成するだけでなく，それらの品質を保証するのが望ましい．従来の開発手法では，ソフトウェア実装後に検証作業が集中していたが，上流工程で決定するソフトウェアの構造に関する性質は，実装前に検証すべきである．そこで，設計工程で作成する設計モデルをモデル検査により検証することにした．
検証した設計モデルに基づいて実装する際，設計検証で保証した性質は，実装後も成立していなければならない．よって，検証した結果をソフトウェア実装後も保証する仕組みが必要である．そこで，本研究では，モデル検査を用いて十分に検証された設計モデルをテストオラクルと見なし，テストケースを自動生成することにより，実装の検証を行うことにした．
(2)アプリケーションソフトウェアの検証
検証対象のソフトウェアが RTOS 上で動作するマルチタスクソフトウェアの場合，それぞれのタスクの振る舞いを記述するだけでなく，それらのタスクがどのようにスケジューリングされるか，などについても考慮しなくてはならない．さらに，組込みソフトウェアでは，ハードウェアからの割り込みによる処理を行う必要がある．通常，割り込みは通常のタスクの動作より優先的に取り扱われる．これらのことが絡み合うと，非常に複雑な振る舞いになり，検証を困難にしている．一方，現在提案されているモデル検査ツールでは，このような複雑な並行処理を直接的に取り扱うことはできない．そこで，以上のような複雑な振る舞いを検証するためのモデル検査アルゴリズムとツールを提案する．
本研究では，組込みソフトウェアで重要である実時間性を，設計工程で検証する手法に焦点を当てる．設計工程では，システムの振る舞いや時間に関する見積りを試行錯誤するため，様々なトレードオフを行いながら，それらを決定していく．そのため，振る舞いや時間見積りを変更しながら，検証をしていくことになる．そのような活動を効率的に行うため，本研究では，パラメトリック解析の手法を採用した．パラメトリック解析では，時間を定数ではなく変数(パラメータ)として設計モデルに記述して，期待する性質(例えば，デッドラインを守る)が成立するような条件を生成する．この条件を守るような時間であれば，期待する性質が成立することが保証されるので，最適な時間見積りを効率的

に行うことが可能になる．

４．研究成果
本研究では，RTOS 上で動作する並行・並列アプリケーションソフトウェアと，RTOS 自体を対象とした．それぞれに関する成果を以下に示す．

(1)RTOS の検証
①環境自動生成ツールの提案
本研究では，車載システム用の RTOS である OSEK/VDX を対象とした． RTOS は，タスクや割り込みルーチン(ISR)からのシステムサービスの呼び出しを受けて動作をするオープンシステムである．すなわち，タスクや ISR が無いと動作しないのである．設計モデルも同様で，実行可能なくらい詳細に記述はされているが，タスクや ISR からシステムサービスを呼び出さないと実際には動作しない．Spin による検証を行うためには，RTOS の設計モデルとは別に，タスクや ISR などを表現する外部の記述が必要である．このような記述は環境と呼ばれている．一方，このような環境には，タスクの数，ISR の数，優先度の割り当てなど，様々な構成が考えられる．それぞれの構成毎に手作業で環境を作成するのは非常に手間がかかり困難である．そこで，UML を拡張して構成のバリエーションをまとめて記述する手法，および，その記述から環境を自動生成するアルゴリズムとツールを提案した．

②検証結果の分析方法の提案
上記の方法で自動生成した環境毎にモデル検査を実施することになる．モデル検査自体は自動的に行うことができる．しかしながら，生成された環境の数が膨大となるため，その実行時間に時間がかかることが判明した．そこで，コンピュータクラスタを用いて，並列に実施する手法を提案した．このようにして実施したモデル検査の結果も膨大となり，手作業で結果の分析を行うことは困難である．そのため，関係データベースに検証データを格納し，SQL でクエリを発行することにより，その分析を行った．その結果，効果的，かつ，効果的に設計検証を行うことができた．

③テストケースの自動生成法の提案
検証した設計モデルに基づいて実装する際，設計検証で保証した性質は，実装後も成立していなければならない．よって，検証した結果をソフトウェア実装後も保証する仕組みが必要である．そこで， RTOS の実装が設計モデルと整合していることを確認するために，設計モデルからテストケースを自動生成し，網羅的にテストすることにした．本研究では，設計モデルの妥当性を確認するのに，大きな労力を割いている．この活動により，相対的に設計モデルの品質は高いはずである．そのため，設計モデルをテストオラクルと見なし，テストケースを抽出するのである．我々は，モデル検査のアルゴリズムに基づいて，テストケースを網羅的に生成するツールを提案した．このツールを用いて生成したテストケースは 75 万弱である．さらに，これらのテストケースから，テスト用のアプリケーションを自動生成し，RTOS の実装のテストも行った．テストケースの数が膨大なため，テストに３ヶ月を要したが，すべてについて実行することができた．これらのことから，本研究で提案した手法とツールは，現実問題に適用できることがわかった．モデル検査に基づいた手法を，このように大規模に適用した事例は見たことが無く，非常に大きな成果が得られたと考えている．

(2)アプリケーションソフトウェアの検証
実時間ソフトウェアの設計を対象としたパラメトリック分析手法を提案した．この手法では，動作の実行時間を変数として表現した設計モデルを対象として，デッドラインなどの実時間性を満たす変数に関する条件を求める．このような手法は，モデルの設定によっては決定不能問題になるが，本研究では，現実的なセッティングに基づいて，定数として扱うものと，変数として扱うものを切り分けた．例えば，タスクの周期は定数であるが，状態遷移中に実行するアクションにかかる時間は変数とした．これにより，決定可能で停止するモデル検査アルゴリズムを提案することに成功した．そして，そのアルゴリズムの停止性，健全性，完全性などの基本的な性質を証明し，ツールの実装を行った．実装したツールでは，入力となるモデル化言語を定義し，提案したアルゴリズムに基づいて自動的に分析を行う．また，このアルゴリズムでは，入力のモデルに基づいて到達可能な状態を探索するが，その到達可能な状態はモデルの重要な特徴を表現している．そこで，DOT とよばれるグラフを表現する記述を生成し，到達可能な状態を表示できるようにした．そして，ライントレーサや踏切システムなどの，いくらかの典型的な振る舞いの検証に適用して，有効性を確認した．

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.Pham Ngoc Hung, Viet Ha Nguyen, Toshiaki Aoki, Takuya Katayama, A Minimized Assumption Generation Method for Component-Based Software Verification,

IEICE Transactions, E93-D, No.8, pp.2172-2181, 2010, 査読有

2.Pham Ngoc Hung, Toshiaki Aoki, Takuya Katayama: Modular Conformance Testing and Assume-Guarantee Verification for Evolving Component-Based Software, IEICE Transactions Vol.E92-A, No.11, pp.2772-2780, Nov., 2009, 査読有

3.Yasuyuki Tahara, Nobukazu Yoshioka, Kenji Taguchi, Toshiaki Aoki, Shinichi Honiden: Evolution of a course on model checking for practical applications, ACM SIGCSE Bulletin, Volume 41 , Issue 2 (June 2009), p.38-44, 2009, 査読有

4.Hideaki Nishihara, Koichi Shinozaki, Koji Hayamizu, Toshiaki Aoki, Kenji Taguchi, Fumihiro Kumeno:Model checking education for software engineers in Japan, ACM SIGCSE Bulletin, Volume 41 , Issue 2 (June 2009), p.45-50, 2009, 査読有

〔学会発表〕(計 22 件)

1.Pham Ngoc Hung, Viet Ha Nguyen, Toshiaki Aoki, Takuya Katayama, An Improvement of Minimized Assumption Generation Method for Component-Based Software Verification, IEEE-RIVF International Conference on Computing and Communication Technologies,2012.2.28, ホーチミンシティ, ベトナム

2.Jiang Chen, Toshiaki Aoki, Conformance Testing for OSEK/VDX Operating System Using Model Checking, Asia-Pacific Software Engineering Conference, 2011.11.7, ホーチミンシティ, ベトナム

3.Warawoot Pacharoen, Toshiaki Aoki, Athasit Surarerks, Pattarasinee Bhattarakosol,Conformance Verification between Web Service Choreography and Implementation Using Learning and Model Checking, IEEE International Conference on Web Services, 2011.7.4, ワシントン DC, アメリカ

4.Hsin-Hung Lin, Toshiaki Aoki, Takuya Katayama, Automated Adaptor Generation for Services Based on Pushdown ModelChecking, IEEE International Conference and Workshops on Engineering of Computer-Based Systems, 2011.4.18, ラスベガス, アメリカ

5.Pham Ngoc Hung, Nguyen Viet Ha, Toshiaki Aoki, Takuya Katayama, Assume-Guarantee Tools for Component-Based Software Verification, International Conference on Knowledge and Systems Engineering, 2010.10.8, ハノイ,ベトナム

6.Kenro Yatake, Toshiaki Aoki, Automatic Generation of Model Checking Scripts based on Environment Modeling, International SPIN Workshop on Model Checking of Software, 2010.9.27, エンスヘーデ,オランダ

7.Chaiwat Sathawornwichit, Toshiaki Aoki, Takuya Katayama, Modeling of Real-Time System Designs for Parametric Analysis, IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, 2010.8.23, タイパ,マカオ

8.谷崎裕明, 青木利晃, 片山卓也, Alloy を用いた構成変更支援ツールと適用実験, 情報処理学会 第 169 回ソフトウェア工学研究会, 2010.7.22, 北九州

9.青木利晃, RTOS 設計検証の経験から, ソフトウェアシンポジウム, 2010.6.10, 横浜

10.青木利晃, 形式手法教育の取り組みについて, 先端ソフトウェア工学に関する Grace 国際シンポジウム 形式手法の産業応用ワークショップ 2010, 2010.3.15, 東京

11.Hsin-Hung Lin, Toshiaki Aoki, Takuya Katayama, Non-Regular Adaptation of Services Using Model Checking, IEEE International Symposium on Object-Oriented/Component/Service-oriented Real-Time Distributed Computing, 2010.5.6, カルモナ,スペイン

12.矢竹健朗, 西端浩和, 青木利晃, 環境モデリングによるモデル検査スクリプトの自動生成, 組込みシステムシンポジウム, 2009.10.21, 東京

13.西原秀明, 青木利晃, 粂野文洋, 篠崎孝一, 田口研治, 早水公二, MCBOK2008:ソフトウェア開発のためのモデル検査知識体系, 組込みシステムシンポジウム, 2009.10.21, 東京

14.青木利晃,モデル検査手法の普及活動とその応用, SPI Japan 2009, 2009.10.6, 新潟

15.Pham Ngoc Hung, Toshiaki Aoki and Takuya Katayama, An effective framework for assume-guarantee verification of evolving component-based software, Proceedings of the joint international and annual ERCIM workshops on Principles of software evolution (IWPSE) and software evolution (Evol) workshops, 2009.9.20, アムステルダム, オランダ.

16.Pham Ngoc Hung, Toshiaki Aoki and Takuya Katayama, A Minimized Assumption Generation Method for Component-Based Software Verification, In the 6th International Colloquium on Theoretical Aspect of Computing, 2009.8.16, クアランプール, マレーシア.

17.青木利晃, NGUYEN Tam Thi Minh, モデル検査による設計検証と整合テスト, 情報処理学会 第 14 回組込みシステム研究会, 2009.7.24, 愛知.

18.青木利晃, 実用的な形式手法 - モデル検査手法とその応用, 東芝ソフトウェアフォーラム 2009／第九回東芝 SEPG カンファレンス, 2009.7.10, 神奈川

19.Toshiaki Aoki, Tadashi Sekiguchi, Masayuki Hirayama, and Tomoji Kishi, Detecting and Analyzing State Inconsistencies in Multi-task Software, 12th IEEE International Symposium on Object-Oriented/Component/Service-oriented Real-Time Distributed Computing, 2009.3.20, 東京.

20.土肥雅俊, 青木利晃, C プログラムの実行に基づいたモデル検査実験, ソフトウェア工学の基礎ワークショップ, 2008.11.14, 兵庫.

21.青木利晃, 山崎真吾, モデル検査によるリアルタイムオペレーティングシステムの設計検証, 組込みシステムシンポジウム, 2008.10.31, 東京.

22.Toshiaki Aoki, Model Checking Multi-task Software on Real-time Operating Systems, International Symposium on Object-Oriented Real-Time Distributed Computing 2008, 2008.5.7, フロリダ, アメリカ.

〔図書〕(計 3 件)
1.青木利晃, CQ 出版, 組込みソフトウェア開発技術, 9 章 組込みソフトウェアの静的検証技術, 351 ページ(pp.271-307), 2011

2.青木利晃,電子情報通信学会 「知識ベース」, UML/ステートチャート, 7 群 1 編 2 章 5 節(pp.35-44), 2009.

3.吉岡信和, 青木利晃, 田原康之：SPIN による設計モデル検証, 近代科学社, 226 ページ(pp.15-113), 2008.

〔産業財産権〕
○出願状況（計 0 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 0 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
青木　利晃（AOKI TOSHIAKI）
北陸先端科学技術大学院大学・情報科学研究科・准教授
研究者番号：20313702