

論文の内容の要旨

論文題目	進化型組込みソフトウェアにおける 継続的なメンテナンス性向上手法に関する研究
学位 申請者	佐々木 隆益

第1章：序論

近年、技術革新スピードが加速し、ソフトウェア開発コストが増加する一方、市場ニーズは短期間で変化し、技術の陳腐化、急速な製品価値低下が起きている。このような継続的な変化に対し、迅速に対応しなければならないようなソフトウェアを進化型のソフトウェアと呼び、特に進化型の組込みソフトウェアでは、長期的なメンテナンス性の向上が不可欠である。

そこで、本研究では、進化型組込みソフトウェアにおいて、継続してメンテナンス性を向上させることができる手法の開発を目的とする。

第2章：ソフトウェア開発プロセス

本章では、まず、ウォーターフォールモデルに代表される計画的開発手法とアジャイル開発に代表される適応的开发手法を中心としたソフトウェアの開発モデルを説明する。

続いて、ソフトウェアプロダクトラインの概念、ソフトウェアメンテナンスに関する一般的な定義、およびソフトウェア進化について述べる。

第3章：従来のソフトウェア分析方法論

本章では、ソフトウェアのメンテナンス性に関するソフトウェアの品質特性についての定義およびソフトウェアの分析手法とリファクタリングについて説明する。

第4章：継続的なメンテナンス性向上手法

本章では、本研究で提案する手法の全体像について述べる。

まず、本研究の対象である進化型組込みソフトウェアについて定義する。

続いて、進化型組込みソフトウェアに対して、従来の開発手法を適用した場合の課題を示し、進化型組込みソフトウェアに必要とされるメンテナンス性について定義する。

最後に、進化型組込みソフトウェアが満たすべき要件について示し、各要件に対応した拡張性可視化手法、拡張性強化手法、拡張性評価手法を提案する。

第 5 章：拡張性可視化手法

本章では、拡張性可視化手法について説明する。拡張性可視化手法は、ソースコードのみを入力とするアプローチをとり、オブジェクト指向言語構造と組込みソフトウェアのレイヤ構造を利用したアルゴリズムを特徴とする手法である。また、2種類の可視化モデルを作成し、デザインパターンや継承構造にもとづいて拡張性構造を抽出して可視化モデル内で表示する。

さらに、実際の進化型組込みソフトウェア製品を用いた評価実験を実施し、利用者に対し有用な可視化情報を提供することができ、またコード規模増大に対しても優位性のある手法であることを確認した。

第 6 章：拡張性強化手法

本章では、拡張性強化手法について説明する。拡張性強化手法は、メンテナンス性に関する問題箇所の定義と問題箇所の検出アルゴリズムを特徴とした手法である。また、メンテナンス性に関する問題構造のパターンを定義したルールと、可視化されたメンテナンス性の構造を比較することにより、問題箇所を自動的に特定し、各ルールに対応した拡張性の強化ガイドを出力する。

さらに、前章と同様に評価実験を実施し、将来の進化に対してメンテナンス性強化が有効であることを確認した。

第 7 章：拡張性評価手法

本章では、拡張性評価手法について説明する。拡張性評価手法は、進化の大きさの表現とメンテナンス性の定量化指標を特徴とした手法である。また、メンテナンス性の問題の有無に関わらず、拡張性の構造について相対的に定量化し、変更許容性と変更容易性という指標により、進化に対する機能毎の拡張性を評価することで、客観的なトレードオフ判断を提供する。

さらに、前章と同様に評価実験を実施し、進化型組込みソフトウェアの各機能に対し、2種類の評価指標を用いることで、必要とするメンテナンス性と性能や強化コストのトレードオフの検討が可能であることを確認した。

第 8 章：関連研究

本章では、本研究で用いた各要素技術に対する関連研究について述べる。

まず、ソースコードから特定の関心事を抽出する研究、特定の関心事を定量化する研究、抽出した情報をビジュアル化する研究についてとりあげる。

続いて、ソースコードの強化すべき箇所を特定する研究と、強化の教示方法に関する研究についてとりあげる。最後に、メンテナンス性を算出する研究と、ソフトウェアの評価方法に関する研究についてとりあげる。

第 9 章：結論

本研究では、ソースコードから拡張性に関する構造を可視化し、拡張性の強化および拡張性の評価を行うことで、進化型組込みソフトウェアが必要とするメンテナンス性を向上させるための要件を満たす手法の提案を行った。

ソースコードのメンテナンスに不安を抱えながらも、機能追加、拡張を繰り返している多くの開発現場において、ソフトウェアの資産価値を高める有効な手段となることが期待できる。今後は、拡張性強化ルールの拡充、異なる組込みソフトウェアへの適用や非オブジェクト指向言語へ対応が望まれる。

論文審査の結果の要旨

学位申請者氏名 佐々木 隆益

審査委員主査 大須賀 昭彦

委員 田中 健次

委員 広田 光一

委員 田原 康之

委員 石川 冬樹

近年、ネットワークの高速化、センサの小型化、コンピュータチップの高性能化等の技術革新に伴い、モバイル、クラウド、AR等、新しい概念の製品が登場してきている。これにより、ソフトウェア開発コストが増加する一方、市場ニーズは短期間で変化し、技術の陳腐化、急速な製品価値低下が起きている。このような現在のソフトウェア開発における企業の重要な問題を解決しようとしている。本論文では、特に組込みソフトウェアにフォーカスをした、進化型組込みソフトウェアを定義し、進化型組込みソフトウェアにおいて、継続してメンテナンス性を向上させることができる手法の開発を目的としている。

第2章では、本論文の背景を理解するにあたり必要となる開発プロセスについて、網羅的に説明されている。同様に、本論文で提案されている手法の理解に必要な一般的知識であるソフトウェアプロダクトラインの概念、ソフトウェアメンテナンスに関する一般的な定義、およびソフトウェア進化についても説明されている。

第3章では、本論文の提案手法と比較する一般的なソフトウェア分析方法について、設計情報をもとにした分析手法とソフトウェアの実装情報をもとにした分析手法にわけて説明されている。さらに、提案手法にて改善を図るメンテナンス性に関するソフトウェアの品質特性についての定義およびリファクタリング手法について説明されている。

第4章では、本研究で提案する手法の全体像について説明している。本研究の対象である進化型組込みソフトウェアについて定義し、従来の開発手法である計画的開発手法や適応的開発手法を適用した場合の課題を示している。さらに、進化型組込みソフトウェアに必要とされるメンテナンス性について論じ、メンテナンス性が悪化していく原因を具体的に述べている。

最後に、進化型組込みソフトウェアが満たすべき要件について示し、各要件に対応した拡張性可視化手法、拡張性強化手法、拡張性評価手法を提案している。

第5章では、拡張性可視化手法について説明している。本手法では、ソースコードのみを入力とし、2種類の可視化モデルを作成し、デザインパターンや継承構造にもとづいて拡張性構造を抽出して可視化モデル内で表示している。この手法の特徴は、オブジェクト指向言語構造と組込みソフトウェアのレイヤ構造を利用したアルゴリズムとなっていることである。

さらに、実際の製品を用いた評価実験を実施しており、利用者に対し有用な可視化情報を提供することができ、コード規模増大に対しても優位性のある手法であることも確認されている。

第6章では、拡張性強化手法について説明している。本手法では、メンテナンス性に関する問題構造のパターンを定義したルールと、可視化されたメンテナンス性の構造を比較することにより、問題箇所を自動的に特定し、各ルールに対応した拡張性の強化ガイドを出力している。この手法の特徴は、メンテナンス性に関する問題箇所の定義と問題箇所の検出アルゴリズムであり、定義されたルールについては、本手法の対象を明確にした上で網羅的に導出されている。

さらに、前章と同様、実際の製品を用いた評価実験を実施しており、将来の想定される進化を挙げ、それに対してメンテナンス性の強化が有効であることが確認されている。

第7章では、拡張性評価手法について説明している。本手法では、メンテナンス性の問題有無に関わらず、拡張性の構造について相対的に定量化し、変更許容性と変更容易性という定義した指標により、進化に対する機能毎の拡張性を評価することで、客観的なトレードオフ判断を提供している。この手法の特徴は、進化の大きさの表現とメンテナンス性の定量化指標である。

さらに、前章と同様、実際の製品を用いた評価実験を実施しており、進化型組込みソフトウェアの各機能に対し、2種類の評価指標を用いることで、必要とするメンテナンス性と性能や強化コストのトレードオフの検討が可能であることが確認されている。

第8章では、本研究で用いられている各要素技術に対する関連研究について説明している。提案手法は3つの手法で構成されているため、各手法に関連する研究をそれぞれ説明している。

第9章では、提案内容と評価結果をまとめ、ソースコードのメンテナンスに不安を抱えながらも、機能追加、拡張を繰り返している多くの開発現場において、ソフトウェアの資産価値を高める有効な手段となることが期待できることを示している。

以上で述べてきたように、本研究成果はオリジナリティに富み、実用面でも高い価値を持つものである。よって、本論文は博士(工学)の学位論文として十分な価値を有するものと認める。

以上