

審査の結果の要旨

氏名 馬 雷

本論文は題目「Automatic Verification and Testing for Software with Multiple Versions-- Improving Software Quality in the Era of Multiple Versions --

(複数バージョンのあるソフトウェアの自動検証・検査 -- 複数バージョン管理時代のソフトウェアの品質向上 --)」と称し、ソフトウェア工学において重要性を近年とみに増している複数バージョンを持つソフトウェアの品質管理、特に検証とテストを効率的に管理できるフレームワークについて研究したものである。

本論文は6章より構成される。

第1章は、序論であり、本論文の研究背景を述べたものである。すなわち、ソフトウェアプロダクトラインやソフトウェア進化がソフトウェア工学で重要視される中、複数バージョンを管理することが普通になった近年のソフトウェアのプロダクト管理のもと、検証・テストがその状況に正しく対処できず、結果抜本的に効率を高める解が強く求められるようになった状況を述べている。

第2章は、本研究に先行する結果として発表されてきた成果の主なものについて論じている。

第3章は、複数バージョンのソフトウェアを効率的に管理するために Project Centralization を導入する。Centralization は、複数のソフトウェアプロダクトをコンポーネントにもつ「プロジェクト」を単一のソフトウェアプロダクトに統合する技術である。オブジェクト指向言語を前提とすると、クラスの管理が本質的に重要であるが、その管理の時に特に問題になるバージョンの衝突を検知、回避するために Centralization はクラスを単位としてクラス間の依存グラフを定義する。さらにバージョン衝突の検知、回避をグラフ採色問題として形式化した。Centralization の適用、つまり衝突の回避のためのヒューリスティックアルゴリズムを提案し、それがグラフのサイズの点で十分最適に近い解をしかも高速に生成することを実際の複数バージョンを持つソフトウェアプロダクトラインで示すことができた。

第4章は、Project Centralization を分散環境で動作するソフトウェアに適用した。これを特に Process Centralization と称し、その有効性を示した。Project

Centralization に加えて、複数バージョンのサーバ/クライアントが別プロセスとして同時に動作する状況に、実行時のセマンティクス保存を含めて対応する必要があるが、Centralization の手法を拡張することで問題なく解決できることを示した。Process Centralization 後の分散アプリケーションは、従来から用いられてきたツールを用いて検証が可能になる。Java で書かれた具体的な分散アプリケーションに対して Process Centralization を施し、モデル検査器 Java Path Finder で検証を行った。

第5章は、検証と並んで品質管理に広く用いられているソフトウェアテストを複数バージョンのソフトウェアプロジェクトに適用することを試みる。テストングとしては、フィードバック指向自動テスト方式を採用した。テストングに適用するためにクラス単位の Centralization をメソッドレベルまで細粒化し、具体的なソフトウェアプロジェクトに適用することができた。さらに、実験に使用したテストングソフトウェア Randoop の性能を上げるため、静的解析、さらに動的解析を徹底的に適用して、テストのカバー率を大幅に改善することができた。この効果として実際に市場に出されているソフトウェアに対して従来発見できていなかったバグを発見することができ、Centralization とは独立にテストングの分野で大きな貢献をすることができた。

第6章は以上をまとめて結論とし、さらに将来のこの分野における発展の方向について論じている。

以上要するに、本論文は、ソフトウェアの生産管理工程で当然のものとなった複数バージョンの存在を前にして従来著しく効率の劣っていた検証・テストに対し、Centralization の方法を本質的に拡張、展開することで、その効率を飛躍的に高め、さらにその適用範囲をソフトウェアプロダクトラインと分散アプリケーションのほとんどをカバーするまでに広げ、ソフトウェアの生産管理工程を本質的に合理化する方法論を提示でき、さらに実際に適用したという点で、ソフトウェア工学に寄与するところが少なくない。

よって本論文は博士（工学）の学位請求論文として合格と認められる。