

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 25 年 4 月 10 日現在

機関番号：14603

研究種目：基盤研究（C）

研究期間：2010～2012

課題番号：22500027

研究課題名（和文） ソフトウェア開発プロセスにおける細粒度データ解析システムの開発

研究課題名（英文） Research on the system to investigate fine-grained software process data analysis

研究代表者

飯田 元（IIDA HAJIMU）

奈良先端科学技術大学院大学・情報科学研究科・教授

研究者番号：20232126

研究成果の概要（和文）：

本研究では、細粒度レベルのソフトウェア開発者の行動パターンを形式的手法を用いて解析し評価することで、個人およびチーム規模でのソフトウェア開発プロセスの自動的な分析・評価を行うシステムを開発することを目的として、1) プロセスの複雑度メトリクスの提案と評価、2) プロセス遵守度とソフトウェア品質の間の関連についての統計的分析を行った。また、保守対象ソフトウェアの機能抽出技術や、ソーシャルネット分析・トピック抽出分析によるプロセス分析技術についても提案と評価を行った。

研究成果の概要（英文）：

This research aimed at developing the system which performs automatic analysis and evaluation of the software development process quality in an individual and a team scale based on the formal analysis of the action patterns of the software development person.

The following investigation has been made in this research:

- 1) The proposal and evaluation of a process of the degree metrics of complexity.
- 2) The statistical analysis about the relation between the degree of process observance and software quality.

Moreover, technique for feature extraction from the program code and technique for process-analysis by social net analysis or topic extraction analysis was performed.

交付決定額

(金額単位：円)

	直接経費	間接経費	合計
2010 年度	1,000,000	300,000	1,300,000
2011 年度	1,200,000	360,000	1,560,000
2012 年度	1,100,000	330,000	1,430,000
総計	3,300,000	990,000	4,290,000

研究分野：情報学

科研費の分科・細目：計算基盤・ソフトウェア

キーワード：ソフトウェア工学、ソフトウェア開発効率化・安定、リアルタイムマネジメント、可視化、情報システム

1. 研究開始当初の背景

1980年代に Osterweil らによるプロセスプログラミングの提唱以降、形式的なプロセスモデルは、シミュレーション以外の用途では十分に活用されず、近年では可読性の高いドキュメンテーション手法としての側面が強調されてきている。

一方、ソフトウェアプロセスの改善は依然として大部分の情報サービス産業にとって重要な課題であるが、CMMIモデル等を用いた組織プロセスの評価と改善[2]には多大な実施コストがかかるため、一定規模を持たない組織には適用が困難である。また、近年普及してきている、短納期かつ反復的な軽量プロセスに対しても評価の視点や粒度が異なるため、適用は容易ではなかった。

研究代表者らはこれまでに、コンポーネント指向プロセスモデルや定量的開発プロセス管理技術についての研究に携わり、適切な管理パターンが適用されたソフトウェア開発プロセスを容易に構成し、過不足のない測定活動に基づいた開発管理の確立を容易にする手法などを開発してきた。一方で、研究次世代 IT 基盤構築のための研究開発・H19～23（文部科学省）「エンピリカルデータに基づくソフトウェアタグ技術の開発と普及」においては、ソフトウェアタグの可視化技術に関する課題を分担し、ソフトウェアプロセスデータの可視化手法に関しても研究を進めていた。

2. 研究の目的

本研究の目的は、これらの成果に基づく形でモデル化されたソフトウェアプロセスや、それに対応づけられたプロセス実行データを対象に、従来行われていなかった細粒度レベルでのソフトウェアプロセスの形式的分析を行うツールを開発することで、個人やチームレベルでのプロセス品質やプロジェクトの状態把握や改善を効率的に行う手法を開発することであった。具体的には、以下の二つの研究目標を定めた。

1) 細粒度プロセスのための形式的モデルの開発

個人やチームレベルでのプロセスは、バグ管理データやプロダクトの改変履歴、その他様々な開発ツールの実行中に自動的に収集される様々な作業記録の集積(リポジトリ)の解析により特定できるが、実際に

は複数の作業が交錯して記録されている。当初は、それらを分離可能なモデルを想定していたが、実際の開発データの評価を通じ、正規表現モデルによる基礎的なモデルを採用し、そのモデルの元でのプロセス品質とプロダクト品質との相関を実証的に示すことを第一の目的とした

2) プロセスモデルに基づいた定量的なプロセス評価指標の開発

1)で定義した形式的モデルに基づいた定量的なプロセス評価指標、たとえば、プロセスの実行順序の遵守度合い、反復プロセスのターンアラウンドタイム、計画と実績の間の乖離、などの情報に基づいて、プロセスが適切に運用されているかどうか、プロジェクトに問題が発生していないか、等について定量的な判断が行えるようなプロセス指標を新たに定義し、それらのメトリクスを計測するツールを試作と、実開発プロジェクトのデータを用いた有効性の検証を行うこととした。

3. 研究の方法

1) まず、分析対象をソフトウェアの保守プロセスに定め、細粒度プロセスモデルの検討と分析ツールの試作を行った。具体的には、オープンソースソフトウェアの開発リポジトリより保守フェーズにおけるバグ追跡システムのログデータとソースコードの改変履歴データを得て、デバッグプロセスを単純な有限状態遷移図でモデル化し、モデルと実際の開発データとの間の照合可能性について検討を行った。

想定する特徴点として、例えば、繰り返し修正が行われるものの、なかなか解決されない障害、あるいは、長期にわたって作業が滞ったり、対処されずに放置された障害などについて着目し、プロセスの振る舞いの観点からの特徴（状態の遷移回数や遷移の時間間隔など）を定量化し、それらと実際のソフトウェア品質や作業対象となったバグの複雑さとの関連の有無を実際に検証しすることとした。

2) 次に、提案モデルに基づく定量的なプロセス評価手法の開発を目的として以下の取組みを行った。

2-1) オープンソースプロジェクトを対象としたデバッグプロセスと品質との関連の分析：

開発リポジトリ中のバグ追跡システムのログデータを用い、実開発プロセスの振舞と有限状態遷移モデルと実際の間の照合を行い、個々の障害除去プロセスの細粒度における特徴（繰り返し修正が行われる、長期にわたって作業が滞る、規定された手順を逸脱しているなど）を定量化したメトリクス、および、障害除去のために施されたソースコードの修正内容の複雑度やプログラムスライシングに基づく規模などの定量的メトリクスと、最終的なソフトウェアの品質との間の相関を調査した。具体的な試行対象として、WxWidget と、FileZilla を対象に、保守フェーズにおけるバグ追跡システムのログデータを用いた実開発プロセスの振舞と有限状態遷移モデルと実際の間の照合を行った。

保守プロセスの細粒度分析に基づいたプロジェクト診断手法整備の一部として、個々の障害除去プロセスの細粒度における特徴（繰り返し修正が行われる、長期にわたって作業が滞る、規定された手順を逸脱しているなど）を定量化したメトリクスと、障害除去のために施されたソースコードの修正内容の複雑度やプログラムスライシングに基づく規模などの定量的メトリクスと、最終的なソフトウェアの品質との間の相関を調査・分析した。

2-2) 請負開発による大規模システム（大学の教務システム）開発プロジェクトを対象にしたプロセス複雑度と品質との関連の分析：

プロセス内で派生した微細プロセス群をもとにした複雑度メトリクスとそれに基づいて組織固有の品質予測モデルを構築するモデルを構築する手法を提案し、当該プロジェクトのデータによる評価を行なった。

2-3) リファクタリングプロセスと品質との関連の調査：

2-1) が個々のバグ修正のためのプロセスを対象として、その作業系列と品質との関連に注目しているのに対し、この調査では、ソフトウェアの再設計（リファクタリング）プロセスに視点を移し、プロセスと品質との間の関連を調べた。本調査では、SZZ アルゴリズムや UMLDiff アルゴリズムなど既存手法を応用することで、ソフトウェアのリファクタリング作業系列をリポジトリから抽出する手法を考案し、適用例として、リファクタリングプロセスの実施と、その後の開発プロセスにおける欠陥導入率との関連を調査した。

4. 研究成果

- 1) 保守プロセスにおけるプログラム修正のための微細プロセスにおける遵守度の指標を定義し、遵守度の低いプロセスと、個々の修正内容の複雑さ、修正後のソフトウェア品質との間の統計的相関を確かめた結果、複雑化修正を行うプロセスでは遵守度が低くなる傾向があること、遵守度の低いプロセスでは品質の低いプロダクトの得られる可能性が高いことなどを実データに基づいて示すことができた。
- 2) プロセス内で派生した微細プロセス群の数に基づくプロセス複雑度メトリクスと、それに基づいて組織固有の品質予測モデルを構築するモデルを構築する手法を提案し、当該プロジェクトのデータによる評価を行なった。評価結果により、提案したプロセス複雑さメトリクスとソフトウェア品質の間には有意な相関がみられ、プロセス進行中の作業見直などに活用できることを示した。さらに、これまでの提案内容を系統的に統合するためのメタ手法について方向性を示した。
- 3) リファクタリングプロセスの詳細な分析のためのフレームワーク（各種ツールの統合的使用とデータの分析方法や使用アルゴリズム）を示すことができた。これにより、リファクタリングに的を絞った保守プロセスの細粒度分析のための基盤技術を整えることができた。基盤を用いた、より本格的な分析は今後の課題である。

これらの成果についてはそれぞれ論文文化を行ない、国内外論文誌や研究会での口頭発表を行った。研究終了後はこれらの成果をより高次に統合することでソフトウェア開発プロセスを複数視点により自動的に評価する環境へと発展させていく予定である。

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2件）

- ① Raula Gaikovina Kula, Kyohei Fushida, Norihiro Yoshida, and Hajimu Iida, "Micro Process Analysis of Maintenance Effort: an Oss Case Study Using Metrics Based on Program Slicing," JOURNAL OF SOFTWARE: EVOLUTION AND PROCESS,

Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/smr.1572, October 2012 (査読有) .

- ② 尾花将輝, 花川典子, 飯田元, "ソフトウェア開発プロセスの並列作業に基づくプロセスの複雑さの提案," 日本ソフトウェア科学会 コンピュータソフトウェア, 29(4), pp. 278-292, 2012年 (査読有) .

[学会発表] (計 17件)

- ① 山田悠太, 藤原賢二, 吉田則裕, 飯田元, "トピック抽出に基づく開発者の活動に着目したリポジトリ可視化手法," 情報処理学会研究報告 ソフトウェア工学研究会報告, 2012-SE-178(16), 広島, 2012年11月.
- ② Raula Gaikovina Kula, Kyohei Fushida, Norihiro Yoshida, and Hajimu Iida, "Experimental Study of Quantitative Analysis of Maintenance Effort Using Program Slicing-Based Metrics," In Proceedings of International Workshop on Software Quality and Management, pp. 50-57, Hong Kong, December 2012 (査読有) .
- ③ Raula Gaikovina Kula, Ana Erika Camargo Cruz, Norihiro Yoshida, Kazuki Hamasaki, Kenji Fujiwara, Xin Yang, and Hajimu Iida, "Using Profiling Metrics to Categorize Peer Review Types in the Android Project," In IEEE International Symposium on Software Reliability Engineering Workshops, pp. 146-151, Dallas, USA, November 2012 (査読有) .
- ④ 平山 力地, 吉田 則裕, 飯田元, "スライスに基づく凝集度を用いて自動分割を行うプログラム理解支援手法," 電子情報通信学会技術報告, 112(164), pp. 127-132, 函館, 2012年7月.
- ⑤ 濱崎一樹, 藤原 賢二, 吉田 則裕, Raula Gaikovina Kula, 伏田 享平, 飯田元, "Android Open Source Project を対象としたパッチレビュー活動の調査," 情報処理学会研究報告 ソフトウェア工学研究会報告, 2012-SE-176(12), 東京, 2012年5月.
- ⑥ 吉岡 俊輔, 吉田 則裕, 伏田 享平, 飯田元, "近傍ハッシュ法を用いた2段階のクラスタリングに基づく Near-miss クローンの検出," 電子情報通信学会技術報告, 111(481), pp. 37-42, 沖縄, 2012年3月.
- ⑦ Norihiro Yoshida, Masataka Kinoshita, and Hajimu Iida, "A Cohesion Metric Approach to Dividing Source Code into Functional Segments to Improve Maintainability," In Proceedings of the 16th European Conference on Software Maintenance and Reengineering (CSMR 2012), pp. 365-370, Szeged, Hungary, March 2012 (査読有) .
- ⑧ Xin Yang, Raula Gaikovina Kula, Ana Camargo Cruz, Norihiro Yoshida, Kazuki Hamasaki, Kenji Fujiwara, and Hajimu Iida, "Understanding OSS Peer Review Roles in Peer Review Social Network (Person)," In Proceedings of the 19th Asia-Pacific Software Engineering Conference (APSEC 2012), pp. 709-712, Hong Kong, 2012 (査読有) .
- ⑨ Shunsuke Yoshioka, Norihiro Yoshida, Kyohei Fushida, and Hajimu Iida, "Scalable Detection of Semantic Clones Based on Two-Stage Clustering," International Symposium on Software Reliability Engineering (ISSRE 2011), Hiroshima, Japan, November 2011 (査読有) .
- ⑩ Kenji Fujiwara, Kyohei Fushida, Norihiro Yoshida, and Hajimu Iida, "An Approach to Investigating How a Lack of Software Refactoring Effects Defect Density," In IEICE Technical Report, 111(107), pp. 59-62, Seoul, Korea, June 2011.
- ⑪ Masaki Obana, Noriko Hanakawa, and Hajimu Iida, "A Process Complexity-Product Quality (PCPQ) Model Based on Process Fragment with Workflow Management Tables," Proceedings of the 12th Product-Focused Software Process Improvement (PROFES 2011), Bari, Italy, pp. 171-185, 2011 (査読有) .
- ⑫ 尾花 将輝, 花川 典子, 飯田元, "ソフトウェア開発プロセスの複雑さに着目した PCPQ モデルの構築," ソフトウェアエンジニアリングシンポジウム 2011, 東京, 2011年 (査読有) .
- ⑬ Raula Gaikovina Kula, Kyohei Fushida, Norihiro Yoshida, and Hajimu Iida, "Using Program Slicing Metrics for the Analysis of Code Change Processes," In Proceedings of 2010 International Workshop on Empirical Software Engineering in Practice (IWESEP 2010), pp. 53-58, Nara, December 2010 (査読有) .
- ⑭ Masaki Obana, Noriko Hanakawa, Norihiro Yoshida, and Hajimu Iida,

"Process Fragment Based Process Complexity with Workflow Management Tables," In Proceedings of 2010 International Workshop on Empirical Software Engineering in Practice (IWESEP 2010), pp. 7-12, Nara, December 2010 (査読有) .

⑮ Raula Gaikovina Kula, Kyohei Fushida, Shinji Kawaguchi, and Hajimu Iida, "Analysis of Bug Fixing Processes Using Program Slicing Metrics," Proceedings of the 11th Product-Focused Software Process Improvement (PROFES 2010), LNCS 6156, pp. 032-046, Limerick, Ireland, June 2010 (査読有) .

⑯ 藤原 賢二, 伏田 享平, 吉田 則裕, 飯田元, "ソースコード履歴情報に基づくリファクタリングと欠陥の関係分析," 平成 22 年度 情報処理学会関西支部 支部大会 講演論文集, 大阪, 2010 年 9 月 (査読有) .

6. 研究組織

(1) 研究代表者

飯田 元 (IIDA HAJIMU)
奈良先端科学技術大学院大学・情報科学研究科・教授
研究者番号 : 20232126

(2) 研究分担者

なし

(3) 連携研究者

なし