

柔軟い制約を用いた自動デバッグ方式（国際共同研究強化）

著者	細部 博史
雑誌名	科学研究費助成事業 研究成果報告書
ページ	1-5
発行年	2020-06-04
URL	http://hdl.handle.net/10114/00024347

令和 2 年 6 月 4 日現在

機関番号：32675

研究種目：国際共同研究加速基金（国際共同研究強化）

研究期間：2016～2019

課題番号：15KK0016

研究課題名（和文）柔らかい制約を用いた自動デバッグ方式（国際共同研究強化）

研究課題名（英文）Automated Debugging with Soft Constraints(Fostering Joint International Research)

研究代表者

細部 博史（Hosobe, Hiroshi）

法政大学・情報科学部・教授

研究者番号：60321577

交付決定額（研究期間全体）：（直接経費） 8,600,000円

渡航期間：12ヶ月

研究成果の概要（和文）：ソフトウェアの自動デバッグ方式の構築を目的として、プログラムの不具合箇所発見・可視化方式を研究開発した。本方式は、不具合箇所を含みうるパスの部分をもさらに分類して複数段階の柔らかい制約で符号化することで、不具合箇所候補のランキングを行う。また、複数段階の柔らかい制約を制約階層として扱う制約解消アルゴリズムを構築した。本アルゴリズムは、外部のSMTソルバを用いて通常の制約問題を繰り返し生成し解消することで、制約階層を解消する。

研究成果の学術的意義や社会的意義

本研究成果の学術的意義は、プログラム中の不具合箇所を発見する問題に対して、制約と呼ばれる概念を用いるアプローチを推し進めることで、不具合箇所の候補に順序を付けられることを示し、さらに、不具合箇所発見処理における基盤となる制約解消の新しい手法を示したことである。本研究成果の社会的意義は、ソフトウェアのデバッグという問題に対して、自動デバッグという観点から新たな可能性を示したことである。

研究成果の概要（英文）：To construct a technology for automated software debugging, we studied a method for localizing and visualizing faults in programs. The method performs the ranking of candidates for faults by encoding possibly faulty program paths with multi-level soft constraints. We also constructed an algorithm for solving multi-level soft constraints as constraint hierarchies. The algorithm solves constraint hierarchies by repeatedly generating and solving ordinary constraint problems by using an external SMT solver.

研究分野：プログラミング

キーワード：ソフトウェア デバッグ 制約

様式 F-19-2

1. 研究開始当初の背景

近年、ソフトウェア工学分野で検証とテストの技術が発達し、バグによってプログラムが意図しない振る舞いや結果を生じる状況を発見しやすくなってきている。しかし、そのような技術を用いる場合でも、デバッグにおいてプログラマはトレースを見たりすることで、バグの原因となっているプログラムの不具合箇所を特定して修正することが必要である。

このようなデバッグの手間を軽減するために、不具合箇所発見 (fault localization) に関する研究が行われている。Ball ら (引用文献①) は、モデル検査器を複数回呼び出して得られた反例のトレースを正しい実行のトレースと比較して、正しい実行に現れない遷移を不具合箇所の候補として提示する手法を提案し、C 言語のプログラムに適用した。Griesmayer ら (引用文献②) は C プログラムの仕様と反例から同じ入力に対する正しいプログラムを求め、不具合箇所の候補を提示する手法を提案した。Jose ら (引用文献③) は反例のトレースと事後条件からブール式を構成し、MaxSAT ソルバを用いて MaxSAT 解において充足不能な節を見つけることで C プログラム中の不具合箇所を発見するツール BugAssist を開発した。

研究代表者は、本研究の基課題の研究分担者、研究協力者らと共同で、制約の概念を用いた不具合箇所発見手法を提案した (引用文献④)。この手法は前述の Jose らの手法を制約の導入によって拡張したものである。本手法は C プログラムに対してモデル検査を適用し、得られた反例のトレースと事後条件から制約充足問題を構成して、問題に含まれる既約実行不能集合を求めることで不具合箇所の発見を行う。

これらの背景を踏まえて、研究代表者らが科研費基盤研究 (B) へ応募し採択されたものが基課題「柔らかい制約を用いた自動デバッグ方式」である。基課題では柔らかい制約の考え方をを用いて効果的な不具合箇所発見を行い、さらに C 言語を対象とするデバッグツールを開発する。より具体的には、極小補正集合を柔らかい制約の枠組みとして採用する不具合箇所発見手法として、研究分担者らが full flow-sensitive trace formula (FFTF) 法 (引用文献⑤) と hardened flow-sensitive trace formula (HFTF) 法 (引用文献⑥) を提案した。FFTF 法は元のプログラム全体を制約充足問題として符号化した上で不具合箇所発見を行い、HFTF 法は不具合箇所を含まないと判断したパスの部分の固い制約で、それ以外の部分を柔らかい制約で符号化することで不具合箇所発見処理の効率を向上する。また、基課題の研究と並行して研究協力者らが LocFaults 法 (引用文献⑦) を提案した。本手法は「逸脱」と呼ばれる処理によって、モデル検査で得られたパスを強制的に変更してプログラムを符号化した上で、極小補正集合に基づく不具合箇所発見を行う。これらは従来手法よりプログラムの広い範囲を扱い、より多くの情報を利用できるために、高精度な不具合箇所発見を行うことができる。

2. 研究の目的

本研究はソフトウェアの自動デバッグ方式の構築を目的とする。基課題では、柔らかい制約の考え方をを用いて不具合箇所発見を行い、さらに C 言語を対象とするデバッグツールを開発した。本課題では基課題を発展させて、柔らかい制約と複数パスを用いたプログラムの不具合箇所発見・可視化方式を研究開発する。特に、基課題で構築したプログラム符号化方式の拡張、それに対して柔らかい制約を組み合わせた不具合箇所発見方式の構築、発見した不具合箇所候補をユーザに提示する可視化方式の構築に取り組む。

3. 研究の方法

FFTF 法、HFTF 法、LocFaults 法を包含する新たなプログラム符号化方式を研究開発する。本課題では新たに、LocFaults 法に類似する処理で得られる複数のパスを FFTF 法、HFTF 法のように同時に扱うことで、モデル検査で得られるトレースに比較的近いパスに関して効率的なプログラム符号化を可能にする。

次に、プログラム符号化方式に対して基課題と同様の制約処理手法を組み込んだ不具合箇所発見方式を研究開発する。不具合箇所発見では複数の候補を提示するために複数の解が必要になるため、制約の柔らかさの決め方が重要である。このため、本課題では前述のプログラム符号化方式に適した制約階層 (引用文献⑧) の適用方法を検討する。

さらに、発見された不具合箇所の候補を可視化してユーザに提示する技術を研究開発する。不具合箇所発見の研究における別の流れとして、Tarantula (引用文献⑨) に代表される複数のテストケースの実行パスの可視化を用いるものがある。本課題では、不具合箇所発見の解を効果的に可視化することで、不具合箇所の候補の理解を容易化する技術を構築する。

4. 研究成果

(1) 自動デバッグツール

柔らかい制約と複数パスを用いたプログラムの不具合箇所発見方式を構築した。本方式は、基課題で構築した HFTF 法を拡張したものである。HFTF 法では、不具合箇所を含まないと判断

したパスの部分を固い制約で、それ以外の部分を柔らかい制約で符号化していた。

本方式では、不具合箇所を含みうるパスの部分をさらに分類して複数段階の柔らかい制約で符号化することで、不具合箇所候補のランキングを実現した。柔らかい制約の表現には、制約階層を用いた。通常の制約階層では最良の解候補のみが解とされるが、本方式では制約階層が内部的に解候補の比較を行うことを利用して解候補のランキングを行い、それに基づいて不具合箇所候補のランキングを行っている。また、プログラム中の対応する構文要素の情報を利用して、柔らかい制約によるパスの符号化をより詳細に行うようにすることで、プログラムの不具合箇所候補のランキングをより詳細に決定するようにした。

本方式の実装は、基課題で開発したデバッグツール SNIPER を拡張することで行った。本研究ではさらに、不具合箇所候補を可視化し、ユーザに提示するグラフィカルユーザインタフェースを備えたツールを開発した。

(2) 制約解消系

複数段階の柔らかい制約を制約階層として扱う制約解消アルゴリズムを構築した。制約階層は、階層的な優先度として制約の柔らかさを表現する、柔らかい制約の枠組みである。制約階層で各制約は、強さと呼ばれる階層的な選好が割り当てられ、解は強い制約を可能な限り多く満たすように定められる。このような制約階層を視覚的アプリケーションのために解くアルゴリズムの研究がこれまでに多く行われている。特に線形制約からなる制約階層を解く効率的なアルゴリズムが存在する。しかし、非線形制約を含む制約階層を扱うことは依然として困難である。

本アルゴリズムは、制約階層を直接解く代わりに、外部の SMT ソルバを用いて、通常の制約問題を繰り返し生成し解消する。より具体的には、制約階層の優先度の高いほうから順に制約の誤差を最小化することで、制約階層の解を近似的に求める。その際、特定の優先度を扱うために、その優先度の制約全体の誤差の上限値を二分探索によって徐々に制限するようにしている。これらの処理によって制約の優先度が消去されるため、外部の SMT ソルバを利用することが可能となる。

本アルゴリズムに基づき、制約解消系を実装した。SMT ソルバとして Z3 を用いた。本制約解消系は、Z3 に類似する API を採用し、本アルゴリズムを実装するクラス `ch.Solver` を提供する。必須制約と選好制約は `ch.Solver` のインスタンスに追加され、解は `ch.Solver` のメソッド `solve` を呼び出すことで計算される。

<引用文献>

- ① T. Ball, M. Naik, and S. K. Rajamani: From Symptom to Cause: Localizing Errors in Counterexample Traces, Proc. ACM POPL, pp. 97-105, 2003.
- ② A. Griesmayer, S. Staber, and R. Bloem: Automated Fault Localization for C Programs, Proc. Workshop on Verification and Debugging, Vol. 174-4 of ENTCS, pp. 95-111, 2007.
- ③ M. Jose, and R. Majumdar: Cause Clue Clauses: Error Localization Using Maximum Satisfiability, Proc. ACM PLDI, pp. 437-446, 2011.
- ④ H.B. Ban, H. Hosobe, S. Nakajima, D.N. Nguyen, M. Rueher, and F. Weigl: A Constraint-Based Approach to Error Localization, 日本ソフトウェア科学会第 28 回大会萌芽セッション, pp. 2B-5:1-6, 2011.
- ⑤ S.-M. Lamraoui and S. Nakajima: A Formula-based Approach for Automatic Fault Localization of Imperative Programs, Proc. ICFEM, Vol. 8829 of LNCS, pp. 251-266, 2014.
- ⑥ S.-M. Lamraoui, S. Nakajima, and H. Hosobe: Hardened Flow-Sensitive Trace Formula for Fault Localization, Proc. ICECCS, pp. 50-59, 2015.
- ⑦ M. Bekkouche, H. Collavizza, and Michel Rueher: LocFaults: A New Flow-Driven and Constraint-Based Error Localization Approach, Proc. ACM SAC, pp. 1773-1780, 2015.
- ⑧ A. Borning, B. Freeman-Benson, and M. Wilson: Constraint Hierarchies, Lisp Symbolic Comput., Vol. 5, No. 3, pp. 223-270, 1992.
- ⑨ J.A. Jones, M.J. Harrold, and J. Stasko: Visualization of Test Information to Assist Fault Localization, Proc. ICSE, pp. 467-477, 2002.

5. 主な発表論文等

〔雑誌論文〕 計4件（うち査読付論文 4件 / うち国際共著 0件 / うちオープンアクセス 0件）

1. 著者名 Kei Takei and Hiroshi Hosobe	4. 巻 2
2. 論文標題 A 2-by-6-Button Japanese Software Keyboard for Tablets	5. 発行年 2018年
3. 雑誌名 Proceedings of the 13th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP2018)	6. 最初と最後の頁 147-154
掲載論文のDOI (デジタルオブジェクト識別子) 10.5220/0006652701470154	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -
1. 著者名 Satoru Imura and Hiroshi Hosobe	4. 巻 10901
2. 論文標題 A Hand Gesture-Based Method for Biometric Authentication	5. 発行年 2018年
3. 雑誌名 Lecture Notes in Computer Science (HCI International 2018)	6. 最初と最後の頁 554-566
掲載論文のDOI (デジタルオブジェクト識別子) 10.1007/978-3-319-91238-7_43	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -
1. 著者名 Teppei Yajima and Hiroshi Hosobe	4. 巻 -
2. 論文標題 A Japanese Software Keyboard for Tablets That Reduces User Fatigue	5. 発行年 2018年
3. 雑誌名 Proceedings of the 42nd IEEE International Conference on Computers, Software and Applications (COMPSAC2018)	6. 最初と最後の頁 339-346
掲載論文のDOI (デジタルオブジェクト識別子) 10.1109/COMPSAC.2018.00054	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -
1. 著者名 Hiroshi Hosobe	4. 巻 -
2. 論文標題 Solving Hierarchical Soft Constraints with an SMT Solver	5. 発行年 2020年
3. 雑誌名 Proceedings of the 12th International Conference on Computer and Automation Engineering (ICCAE2020)	6. 最初と最後の頁 42-46
掲載論文のDOI (デジタルオブジェクト識別子) 10.1145/3384613.3384654	査読の有無 有
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

〔学会発表〕 計2件（うち招待講演 1件 / うち国際学会 0件）

1. 発表者名 細部博史
2. 発表標題 ソフト制約のモデル化・解決とその応用
3. 学会等名 人工知能学会第109回人工知能基本問題研究会（招待講演）
4. 発表年 2019年

1. 発表者名 細部博史
2. 発表標題 SMTソルバーを用いた制約階層解消法
3. 学会等名 人工知能学会第112回人工知能基本問題研究会
4. 発表年 2020年

〔図書〕 計0件

〔産業財産権〕

〔その他〕

-

6. 研究組織

	氏名 (ローマ字氏名) (研究者番号)	所属研究機関・部局・職 (機関番号)	備考
主たる 渡航先 の主たる 海外 共同 研究者	ルエア ミシェル (Rueher Michel)	ニースソフィアアンティボリス大学・I3S研究所・教授	