柔らかい制約を用いた自動デバッグ方式

細部, 博史 / HOSOBE, Hiroshi

(雑誌名 / Journal or Publication Title)
科学研究費助成事業 研究成果報告書
(開始ページ / Start Page)
1
(終了ページ / End Page)
4
(発行年 / Year)
2017-06-09

様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

平成　２９　年　　６　月　　９　日現在

機関番号：　３２６７５
研究種目：　基盤研究(B)（一般）
研究期間：　2012～2016
課題番号：　２４３０００１０

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

研究課題名（英文）Automatic Debugging by Using Soft Constraints

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）　　13,800,000円

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，プログラムの誤り特定手法に関する研究を行った．そのためのアプローチとして制約の概念を採用した．具体的には，入力，プログラム，事後条件から制約充足問題を構成した上で，違反の原因となった制約を特定し，プログラム中の対応する部分を誤り箇所の候補として提示するようにした．本研究では特に柔らかい制約を用いた自動デバッグ方式を構築し，Ｃ言語を対象とする自動デバッグツールを開発した．

研究成果の概要（英文）：We studied automatic software debugging by focusing on the localization of faults in programs. For this purpose, we adopted the notion of constraints. Specifically, we construct constraint satisfaction problems from inputs, programs, and post-conditions, and then localize constraints that cause the violations to indicate the corresponding program parts as candidates of the faults. We particularly constructed soft constraint-based methods for automatic debugging, and developed an automatic debugging tool for C programs.

研究分野：　情報学

キーワード：　ソフトウェア　デバッグ　制約

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てい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誤り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，誤り特定に関する研究が行われた．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誤り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．Groce ら[引用文献②]は擬ブールソルバーを用いた最適化によって反例のトレースに類似する正しい実行を計算するＣプログラムのための反例理解支援ツール explain を開発した．Griesmayer ら[引用文献③]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，誤り箇所の候補を提示する手法を提案し，ある仮定のもとで本当の誤りが候補に必ず含まれることを示した．Jose ら[引用文献④]は反例のトレースと事後条件からブール式を構成し，最大充足可能性判定（maximum satisfiability; MaxSAT）ソルバーを用いて MaxSAT 解において充足不能な節を見つけることでＣプログラム中の誤り箇所を特定するツール BugAssist を開発した．C プログラムではなく一般的なソフトウェアのモデルを対象とした研究として，熊澤と玉井[引用文献⑤]はモデルの反例とそれに近い編集距離を持つ正例の間の差分を提示することで誤り箇所の候補とその修正候補を求める手法を提案した．本研究課題の研究分担者（中島）[引用文献⑥]は FODA フィーチャ図で記述されたモデルを命題論理によって解釈し，その充足不能コアを求めることでモデルの誤り箇所の候補を特定する手法を提案した．

本研究課題の研究代表者（細部），研究分担者（中島），研究協力者（Rueher）[引用文献⑦]は共同で制約の概念を用いた誤り特定手法を提案した．本手法は前述の 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（irreducible infeasible set; IIS）を求めることで誤り特定を行う．IIS は線形制約からなる問題が解を持たない状況を分析するためのものであり，制約を１つ削除すると解を持つがそれ自体は解を持たないような部分問題を指す．このようにプログラムの誤り特定で IIS を用いる場合には，対象となる問題が線形に限定される欠点があるが，その一方で線形性を用いた強力かつ効率的な誤り箇所候補の絞り込みが可能になる利点がある．

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し，特にプログラムの誤り特定手法の開発に重点を置く．そのためのアプローチとして制約の概念を採用する．本研究は前項に述べた制約を用いた誤り特定手法を発展させるものであり，プログラムから制約充足問題を構成することで誤り箇所を特定する．

３．研究の方法

(1)柔らかい制約を活用した誤り特定方式の構築

本研究項目は利用する制約の枠組みを高度化することで効果的な誤り特定を行うことを主眼として，制約プログラミングの分野で研究されている柔らかい制約の活用によって誤り箇所の候補を適確に絞り込む方式を構築する．

(2)変異プログラムを援用した誤り特定方式の構築

本研究項目は制約を高度に利用することで効果的な誤り特定を行うことを主眼とする．研究項目(1)では 1 つのトレースから制約充足問題を構成する方法を採るのに対して，本研究項目ではプログラムの制御フローを考慮する．

(3)柔らかい制約を用いた自動デバッグ方式の構築

研究項目(1)の成果である誤り特定のための柔らかい制約の枠組みと，研究項目(2)の成果である制御フローを考慮した誤り特定手法を統合することで，高度な制約の枠組みを高度に利用した誤り特定方式を構築する．

４．研究成果

(1)新しい誤り特定手法の構築

full flow-sensitive trace formula (FFTF)と呼ぶ，新しい命令型プログラム符号化方式に基づく誤り特定手法を構築した．本手法はプログラムの制御フローを考慮し，複数の誤り箇所を持つプログラムを扱うことを可能にするという特徴を持つ．

本手法は，trace formula (TF)に基づく誤り特定手法である．この種の最も単純な手法である前述の Jose らの手法では，有界モデル検査によって得られた反例を用い，プログラムの１つのフローを符号化することで，TF を作成していた．しかし，この方法は，分岐条件の誤りのような制御に関する誤りを扱うことができない．この問題を解決するために，Christ ら[引用文献⑧]は，反例と部分的な制御フローを用いてプログラムを符号化する flow-sensitive trace formula を提案した．

FFTF は，Christ らの符号化方式をさらに発展させ，プログラムの制御フロー全体，すなわち，制御フローグラフを符号化することで，TF を作成するものである．これによって，Christ らの符号化方式では，制御に関する誤りによって他の誤りが隠されてしまうような複数の誤りがある状況も，FFTF は符号化することができる．

さらに，FFTF を用いて，複数の誤り箇所の候補を求めるアルゴリズムを構築した．本アルゴリズムは，複数の反例に対応する極小補正集合を計算した後に，その結果を組み合わせることで誤り箇所の候補を求める．極小補正集合の計算には，MaxSAT を用いている．

(2)誤り特定手法の効率化

前項の FFTF を用いた誤り特定手法では，従来手法よりも TF が巨大になるために，スケーラビリティの問題が生じる．この問題を解決するために，hardened flow-sensitive trace formula (HFTF)と呼ぶ符号化方式に基づく誤り特定手法を構築した．

HFTF は固い制約を積極的に用いてプログラムを符号化する．FFTF 等の従来の TF に基づく手法では，誤り特定のために MaxSAT や類似の問題を解く必要があり，その際に TF の大部分を柔らかい制約として扱っていた．一方，HFTF では TF の可能な部分を固い制約に変えることで，MaxSAT における解の探索を効率化する．

TF において固い制約とすべき部分を求めるために，HFTF ではコンコリック実行を利用する手法を構築した．事前のコンコリック実行によって，実行が成功する入力と失敗する入力の両方を計算し，これを用いて，失敗に関与しないプログラムの部分を求める．HFTF では，このような部分は必ず充足できると考え，固い制約としてプログラムを符号化する．

(3)誤り特定ツールの開発

FFTF または HFTF に基づく誤り特定手法を用いて，C プログラムの誤り特定を行うツール SNIPER を開発した．本ツールは，事前条件と事後条件が付与された C プログラムを入力として受け取り，事後条件が満たされなくなるような入力が存在する場合に，その原因となる誤り箇所の候補を自動的に特定し，結果として提示する．本ツールの開発には C++ を用いた．

C プログラムから TF を構成するために，Clang と LLVM を用いている．これらによって，C プログラムから静的単一代入形式の中間表現を生成する．さらに，有界モデル検査と同様の方法でループと再帰を特定の上限回数まで展開した結果を用いて，FFTF または HFTF に基づく TF を構成する．

誤り特定のための入力の生成と，誤り箇所の候補の計算のために，Yices 1 を用いている．Yices 1 は背景理論付き充足可能性判定のためのソルバーであり，MaxSAT をサポートしている．

本ツールは，GNU General Public License v3 に基づくオープンソースソフトウェアとして GitHub 上で公開した．レポジトリには，本ツールのプログラムに加えて，そのビルドや使用の方法を説明するドキュメントと，本ツールの評価に用いたベンチマークプログラムも含めている．

＜引用文献＞

①T. Ball, M. Naik, and S.K. Rajamani, From Symptom to Cause: Localizing Errors in Counterexample Traces, Proc. ACM POPL, 97-105, 2003.

②A. Groce, S. Chaki, D. Kroening, and O. Strichman, Error Explanation with Distance Metrics, Softw. Tools Tech. Trans., 8(3), 229-247, 2006.

③A. Griesmayer, S. Staber, and R. Bloem, Automated Fault Localization for C Programs, ENTCS (Proc. V&D), 174(4), 95-111, 2007.

④M. Jose and R. Majumdar, Cause Clue Clauses: Error Localization Using Maximum Satisfiability, Proc. ACM PLDI, 437-446, 2011.

⑤熊澤努，玉井哲雄，モデルに基づく誤り特定と反例修正候補の提示，ソフトウェアエンジニアリングシンポジウム論文集，55-62，2009.

⑥S. Nakajima, Semi-Automated Diagnosis of FODA Feature Diagrams, Proc. ACM SAC, 2191-2197, 2010.

⑦H.B. Ban, H. Hosobe, S. Nakajima, D.N. Nguyen, M. Rueher, and F. Weitl, A Constraint-Based Approach to Error Localization, 日本ソフトウェア科学会第 28 回大会，2B-5, 1-6, 2011.

⑧J. Christ, E. Ermis, M. Schäf, and T. Wies, Flow-Sensitive Fault Localization. Proc. VMCAI, 189-208, 2013.

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 6 件）

①Si-Mohamed Lamraoui and Shin Nakajima, A Formula-based Approach for Automatic Fault Localization of Multi-fault Programs, Journal of Information Processing, 査読有, 24(1), 88-98, 2016. DOI:10.2197/ipsjjip.24.88

②Si-Mohamed Lamraoui, Shin Nakajima, and Hiroshi Hosobe, Hardened Flow-sensitive Trace Formula for Fault Localization, Proceedings of the 20th International Conference on Engineering of Complex Computer Systems (ICECCS2015), 査読有, 50-59, 2015.

DOI:10.1109/ICECCS.2015.16
③Shin Nakajima and Si-Mohamed Lamraoui, Fault Localization of Timed Automata Using Maximum Satisfiability, Lecture Notes in Computer Science (Proc. SOFL+MSVL2015), 査読有, 9559, 72-85, 2015.
DOI:10.1007/978-3-319-31220-0_6
④Shin Nakajima and Si-Mohamed Lamraoui, Fault Localization of Energy Consumption Behavior Using Maximum Satisfiability, Lecture Notes in Computer Science (Proc. CyPhy2015), 査読有, 9361, 99-115, 2015.
DOI:10.1007/978-3-319-25141-7_8
⑤Si-Mohamed Lamraoui and Shin Nakajima, A Formula-Based Approach for Automatic Fault Localization of Imperative Programs, Lecture Notes in Computer Science (Proc. ICFEM2014), 査読有, 8829, 251-266, 2014.
DOI:10.1007/978-3-319-11737-9_17

〔学会発表〕(計6件)
①Si-Mohamed Lamraoui and Shin Nakajima, SNIPER: An LLVM-based Automatic Fault Localization Tool for Imperative Programs, 電子情報通信学会ソフトウェアサイエンス研究会, 2015.3.9-2015.3.10, 沖縄県青年会館(沖縄県・那覇市).
②Hiroshi Hosobe, A Soft Constraint-Based Approach to Error Localization, 4th Asian Workshop of Advanced Software Engineering (AWASE2014), 2014.10.11-2014.10.12, 北京(中国).
③Si-Mohamed Lamraoui and Shin Nakajima, SNIPER: A Tool for Automatically Localizing Errors in Imperative Programs,ソフトウェアエンジニアリングシンポジウム2013「プログラム・デバッグ自動化の現状と今後」ワークショップ, 2013.9.9-2013.9.11, 東洋大学白山キャンパス(東京都・文京区).
④Si-Mohamed Lamraoui and Shin Nakajima, Automated Error Localization with Weighted Partial Maximum Satisfiability, 電子情報通信学会ソフトウェアサイエンス研究会, 2013.7.25-2013.7.26, 北海道立道民活動センター(北海道・札幌市).
⑤Hiroshi Hosobe, Shin Nakajima, and Michel Rueher, A Constraint-Based Approach to Error Localization, Fifth CSPSAT & ASP Seminar, 2012.10.11, 神戸大学六甲台キャンパス(兵庫県・神戸市).

〔その他〕
ホームページ等
SNIPER
https://github.com/lamraoui/sniper

６．研究組織
(1)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

(2)研究分担者
中島　震 (NAKAJIMA, Shin)
国立情報学研究所・アーキテクチャ科学研究系・教授
研究者番号：60350211

(3)研究協力者
RUEHER, Michel
LAMRAOUI, Si-Mohamed