B-010

項書き換え系を用いた文字列による検索のための形式仕様の正規化と暗黙の条件の抽出

Standardization of Formal Specification and Extraction of Implicit Constraints by Term Rewriting System

檜垣 廉　Ren Higaki
織田 健　Takeshi Oda

1　はじめに

我々は、形式手法 B Method の信頼性保証の枠組みを利用した部品再利用による高信頼ソフトウェア合成手法である MSSS（モデル充足ソフトウェア合成）手法を提案している。MSSS 手法では、均一な粒度に細分化した形式仕様をキーとし文字列一致で要求を満たす部品を検索するが、細分化の際に暗黙の条件を十分に明示することができず、情報が失われてしまうことが課題であった。本研究では、文字列一致による検索のための形式仕様の正規化と連携した暗黙の条件の抽出手法の改善を図る。

2　背景と研究の目的

2.1　形式手法 B Method

B Method は、集合論と一階述語論理に基づいた仕様記述と、抽象的な仕様であるモデルから具体的な実装への段階的詳細化を行うことでソフトウェアの信頼性を保証する形式手法の一種である [1]。モデルや実装は主に制約条件と代入文から構成され、専用の B 言語で記述される。詳細化の各段階で定理証明器により仕様の無矛盾性と詳細化の整合性を機械的に検証できる。

2.2　MSSS 手法と字面統一

MSSS 手法は、B Method の信頼性保証の枠組みを用いて高信頼部品の生成と要求を記述したモデルからソフトウェアの合成を行う手法である [2]。部品生成時には、モデルは原則一代入文ごとに細分化され、対応する実装の各部分と組になって部品としてリポジトリに登録される。ソフトウェア合成時にも、要求モデルは同一の手法で細分化され、検索キーとなる。部品の検索は、計算量削減のため定理証明を使わず文字列一致によって行う。三鍋は、数学的に等価な仕様が文字列上でも一致するように字面を統一する手法を提案している [3]。また、その手法は型情報や式構造、合流性を考慮することで改良されているが、未だ確立していない [4]。

2.3　暗黙の条件の抽出

細分化時には、核となる代入文に含まれない変数の条件は細分化モデルに記述しないため、暗黙的な条件があると必要な情報が失われてしまうことがある。例えば、$aa \leq bb \ \vee \ bb \leq cc$ のような条件では、推移律によって暗黙的に示された条件である $aa \leq cc$ が明示されておらず、aa と cc のみを含む代入文があった場合、これらの条件は全て抽出されず、適切に細分化が行われない。そこで中村は、事前の推論規則の適用による暗黙の条件の抽出を提案している [2]。しかし、その推論規則の数は十分ではなく、必要な条件を取りこぼす可能性が高い。

2.4　研究の目的

従来手法では、書き換え後の演算子の種類を限定しすぎたため式構造が複雑化し、等価な式を字面上で一致させられないことが多かった。本研究では、書き換え規則の見直しと字面統一手順の変更により、暗黙の条件の抽出を促進した形式仕様の正規化手法の提案を目的とする。

3　従来の字面統一と暗黙の条件の抽出

3.1　従来手法の概要

以下の従来のモデルの細分化に伴う字面統一と暗黙の条件の抽出手法は、中村と三鍋によって提案され、檜垣が手順の追加と改良を行ったものである [2][3][4]。

型推論 書き換え規則のパターンの一部として利用するために型推論を随時行う。

代入文の書き換え B 言語の多様な代入文全体を一定の範囲内の等価な表現に書き換える。

プリミティブ化 演算を限られた種類の（プリミティブな）演算の組み合わせに書き換える。

式構造統一化 冗長な表現の削除と、一定の方針に従って式の構造を統一する処理を行う。書き換え規則は合流性が保証されたものを使用する。

推論 暗黙の条件を導出し、条件に書き加える。

操作分割 操作を一代入文単位に分割する。

制約条件抽出 それぞれの代入文において、登場する変数のみに関する制約条件を抽出する。着目する代入文に登場しない変数が含まれる条件は抽出しない。

構文整列・変数名置換 ＋や × のような可換な演算子のオペランドを一意に並び替え、変数名を登場順により一意に定める。

3.2　従来手法の課題

3.2.1　過度なプリミティブ化による推論の阻害

プリミティブ化により推論規則で使用する演算子が限定され、式構造統一化規則や推論規則に出現する演算子も少数で済むようになる。しかし、従来手法ではプリミティブな演算子を限定しすぎたために式の構造が複雑になり、式構造の統一を阻害した。プリミティブな演算子の範囲をある程度広げることでこの問題は回避できる可能性があったが、その妥当な基準を定めるのが困難だった。複雑な式にマッチする規則を整備しきれなくなる問題は推論規則においても同様に発生すると考えられる。

3.2.2　特殊な演算子への対応

従来手法では、考えられるプリミティブ化規則のうち、B Method の開発環境である Atelier B の証明器で書き換え前後の等価性が証明できたもののみを使用していた。

$$s_1 \cap s_2 \longrightarrow \{x \mid x \in s_1 \wedge x \in s_2\}$$
$$m..n \longrightarrow \{x \mid x \in \mathbb{Z} \wedge m \leq x \wedge x \leq n\}$$
$$\mathrm{id}(s) \longrightarrow \{x, y \mid x \in s \wedge y \in s \wedge x = y\}$$
※ id(s)は恒等写像

図 1: 内包表記を用いたプリミティブ化の例

よって、Atelier B に内蔵されているルールの不足により、数学的な定義通りに書き換えられない演算子が見られた。これは特に一般性の低い B 言語の特殊なデータ構造に関わる演算子に多く、再利用性を低下させていた。

3.2.3 冗長表現の削除のタイミング

推論によって、重複する条件が複数導出されたり、冗長な形になったりすることがある。式構造統一化を推論後に施す方法が考えられるが、必要な暗黙の条件まで削除してしまう可能性がある。また、推論元として必要な条件が事前に削除されてしまう可能性も指摘されていた。

3.2.4 条件数が爆発する推論規則

暗黙の条件の抽出に必要な推論規則の中には、書き換え後に書き換え元のパターンが再度現れ、式の数が爆発するものがある。従来手法ではそのような規則も含めつつ、一定数まで条件が増えると適用を止めていた。しかし、必要な条件が導出される前に不要な条件が多く導出されて推論が停止する可能性があった。

4 字面統一と暗黙の条件の抽出手法

4.1 手法の概要

本手法では、従来手法の流れを踏襲しつつ、プリミティブな演算子の範囲を一定の基準に従って広げることで式構造の複雑化を防ぐ。それに伴い、増えた演算子に対応できるよう従来手法で用いたものと等価な式構造統一化規則と推論規則を追加する。さらに、推論において条件付きの規則を用いることで、条件数の爆発を抑止する。

4.2 プリミティブ化

プリミティブ化における式の複雑化の主因は、図 1 のような、集合や関係の演算の一部を集合の内包表記に統一する規則であった。集合の内包表記は本質的には集合演算を条件式を用いて表すことであり、論理演算の種類が少ないことを利用してプリミティブな演算子の範囲を狭めるのが狙いであったが、関係の演算に含まれる条件が多く、複雑化を招いた。そこで、集合や関係の演算は集合や関係の演算の範囲内でプリミティブ化を行い、新たに集合の内包表記が出現しないようにする。

4.3 式構造統一化

増加したプリミティブな演算子に全て対応できるように、従来の書き換え規則に等価な書き換え規則を追加する。また、集合の内包表記は一般には他の演算子で表すことができないが、できるだけ戻すような書き換えを行う。これにより式の複雑化を防ぐ。

4.4 推論

式構造統一化と同様にプリミティブな演算子の数を多く設定した従来と等価な推論規則を用意する。さらに、

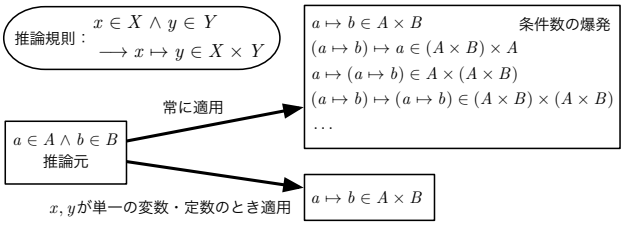

図 2: 条件付きの推論規則の例

図 2 のように、そのまま適用すると爆発する規則は、特定の条件を満たすときのみ適用可能とし追加する。これらの規則を適用し、導出された暗黙の条件を追記する。

4.5 冗長表現の削除

推論中と推論後は、式構造統一化規則に含まれる冗長表現を削除する規則のうち、$p \wedge p \to p$ や $x + 0 \to x$ といった、明らかに暗黙の条件を消去しないもののみを適用することで、推論で導出された暗黙の条件を保持したまま字面の統一と無駄の削減を図る。

5 考察

本研究の内容は手法の提案に留まっているため、今後実験によって以下の 2 点を検証する必要がある。

信頼性の保証 書き換えや、暗黙の条件の抽出は全体の数学的意味を変化させてはならない。よって、規則の適用前後の表現が等価であることを証明する必要がある。ただし、Atelier B の定理証明器のルールにない規則を用いるため、一部は自動証明できず手作業で証明する必要があると考えられる。

従来手法への影響 プリミティブな演算子の増加、規則の追加によって、従来手法で可能であった暗黙の条件の抽出や字面統一の例が本手法でも可能であるかを確かめる必要がある。もし字面を統一できない場合、式構造統一化規則をその都度追加することで対応できるのか、それとも根本的に書き換えの方針を変更する必要があるのかを検討しなければならない。

6 終わりに

本研究ではモデルの細分化における暗黙の条件の抽出と関連する文字列一致による検索のための仕様の正規化の改良方法を提案した。今後は実際のモデルを用いた実験を行い、字面統一と暗黙の条件の抽出の適切な適用が可能な範囲や、書き換え前後の仕様の等価性を検証する必要がある。さらに、MSSS 手法の実現のためにシステムの実装も進めていきたい。

参考文献

[1] 来間啓伸. B メソッドによる形式仕様記述. 近代科学社, 2007.

[2] 中村丈洋. B Method における部品再利用によるソフトウェア合成と高信頼ソフトウェア部品の整備. 電気通信大学 電気通信学研究科 博士 (工学) 学位論文. 2013.

[3] 三鍋孝介, 織田健. 文字列一致による数学的等価性判定可能なモデル分割アルゴリズム. 第 12 回情報科学技術フォーラム論文集 vol.1 pp.271-272. 2013.

[4] 檜垣廉, 織田健. 文字列一致による等価性判定のための形式仕様の正規化. 情報処理学会第 84 回全国大会講演論文集 vol.1 pp.297-298. 2022.