B-008

形式手法B Methodの細粒度部品の結合による高信頼ソフトウェアの合成

High Reliability Software Synthesis by Combining Software Components of B Method

高橋 宏夢† 織田 健†
Hiromu Takahashi Takeshi Oda

1 はじめに

我々は形式手法 B Method を部品再利用に適用することで、入力モデルを満たす高信頼実装を自動合成する手法を提案している [1]。部品結合では依存関係を考慮した部品の結合順序を定める必要がある。また部品が操作呼び出しを行っている場合、変数の値変化が明示的でなくなる。本研究ではモデルの記述を参照しつつ部品を適切に結合する手法を提案する。また合成された実装の機能を保証するための適切な粒度の部品を提案する。

2 研究背景

2.1 形式手法 B Method

形式手法の一種である B Method は仕様記述からコード導出までの一連の工程を支援する [2]。B Method は仕様を数学的記法で記述したモデルの無矛盾性と、モデルから実装への詳細化における整合性を検証できる。

2.2 ソフトウェア自動合成手法

我々は入力モデルを満たすソフトウェアを自動合成する手法を提案してきた (図 1)。部品生成では既存ソフトウェアのモデルを細分化し、次に実装から各細分化モデルを満たす部品を抽出してリポジトリに登録する。ソフトウェア自動合成では部品生成と同様のモデル細分化を行い、リポジトリから等価な細分化モデルを持つ部品を検索する。次に検索で得られた部品を結合することで、入力モデルを満たす合成実装を得る。

2.3 モデル細分化

モデル細分化では代入文を単位としてモデルを分割する。モデルに複数の代入文を含む IF 文がある場合、各代入文が実行される条件を否定と論理積の組み合わせにより導出することで IF 文を分割する。またモデルの構文である ANY 文は、指定した制約を満たす非決定的な値を生成して計算を行う。モデルに ANY 文がある場合、ANY 文を非決定的値を生成して出力する操作と、非決定的値を入力として受け取って参照する操作に分割する。

2.4 部品結合に関する課題

部品結合には以下の課題が存在する。

2.4.1 部品の結合順序

モデルの代入文は同時に実行されるため、各代入文は他の代入文の影響を受けない。一方、実装の代入文は逐次的に実行されるため、入力モデルの代入文に対応する部品の間には依存関係が存在する。そのため、部品の結合順序によってはある部品が他部品の影響を受けることで機能が変化してしまい、入力モデルを満たさなくなる。

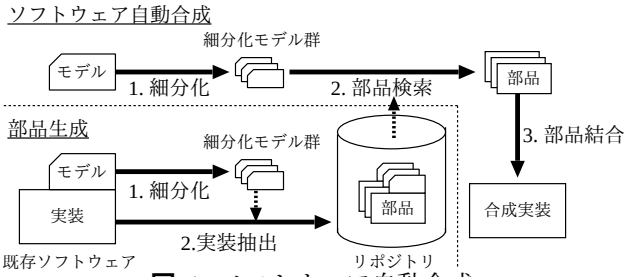

図 1: ソフトウェア自動合成

さらに、部品が標準ライブラリの操作を呼び出している場合は変数の値変化が明示的でないため、部品の記述から依存関係を機械的に判別できない。

2.4.2 部品粒度

部品検索では細分化モデルの機能一致による検索を行うため、細分化モデルおよび部品の粒度が細かいほど部品の再利用性が高くなる。

しかし粒度をあまりに細かく設定すると部品の依存関係が複雑になり、相互に依存する部品が発生しやすくなる。部品が相互に依存する場合、部品の結合順序に関わらず他部品の影響を受ける部品が必ず発生するため、2.4.1 節より入力モデルを満たさなくなる。

2.4.3 合成実装における冗長な条件式の発生

2.3 節より、入力モデルの細分化において 1 つの IF 文が複数の IF 文に分割される場合がある。このとき入力モデルの 1 つの IF 文に対して合成実装に複数の IF 文が発生するが、合成実装の各 IF 文の条件式が同じ条件を表現している場合があり冗長である。

3 依存関係の推定と部品粒度の規定

モデルの振る舞いはモデル変数、実装の振る舞いは実装変数で表現され、モデル変数と実装変数は関係で結び付けられる。ここで実装はモデルを詳細化したものであるため、モデル変数が更新、参照されるとき、対応する実装変数も更新、参照される。

モデルを 1 代入文単位で細分化すると仮定した際、モデルの代入文 S_i が更新する変数に対して参照を行う代入文群として $S_{Ref\,i}$ を発見した場合、$S_{Ref\,i}$ に対応する実装の代入文群 $S'_{Ref\,i}$ は、S_i に対応する実装の代入文 S'_i に依存すると推定できる (図 2)。さらに図 2 のモデルの代入文 S_1 と S_4 のように、更新する変数を相互に参照する代入文群がある場合、対応する実装の代入文群が相互に依存すると推定できる。

よって、モデルにおいて相互に参照する代入文群は分割せずに 1 細分化モデルにし、それ以外の代入文群は 1 代入文単位で分割する。このように細分化モデルを規定することで、部品の間に相互依存関係が発生しなくなる。

†電気通信大学大学院情報理工学研究科情報学専攻

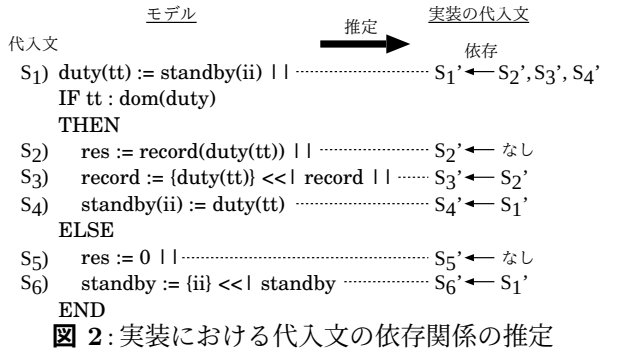

図 2: 実装における代入文の依存関係の推定

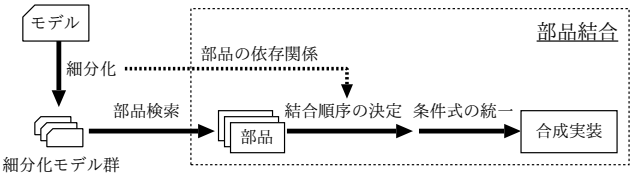

図 3: 部品結合の流れ

4 部品の結合手法

ソフトウェア自動合成において、提案した粒度の部品を結合する手法を述べる。

4.1 部品結合の流れ

部品結合の流れを図 3 に示す。まず入力モデルは 3 節と同様の手法で細分化され、相互に参照する代入文群が 1 細分化モデルに内包される。各細分化モデルをキーとする部品検索により、1 つの細分化モデルに対して 1 つの部品が得られる。

部品を結合する際は、各部品が他部品の影響を受けない順序を決定する。その際、入力モデルの細分化において推定された部品の依存関係を利用する。部品の結合順序が定まったら、IF 文に関して冗長な条件式を統一する。このとき、統一した条件式の評価が代入文の影響を受けないようにする。以上より、入力モデルを満たす合成実装が生成される。

以下、各手順の詳細を述べる。

4.2 部品の結合順序の決定

推定した部品の依存関係から、部品の結合順序を以下の手順で決定する。

1. 依存先でない部品群を先頭に配置しソート済にする
2. ソート済になった部品群を依存関係から除外する
3. 残りの部品群に対して手順 1, 2 を繰り返す

結果として、各部品が他部品の影響を受けない順序が定まる (図 4)。

4.3 条件式の統一

部品の結合後に残る冗長な条件式を統一する。

入力モデルの IF 文において条件式 p の THEN 部にある代入文群を V_t、対応する実装の代入文群を V_t' とする。このとき V_t' の各代入文は全て同じ条件下で実行されるため、V_t' を持つ各 IF 文の条件式の評価値は一致する。

よって、モデルの代入文群 V_t の最も先頭にある代入文 S_{t1} に着目し、対応する実装の代入文 S_{t1}' を持つ IF 文の条件式の評価値を局所変数 X に格納する。ただし、条件式の評価は代入文よりも先に実行する。次に、V_t' の

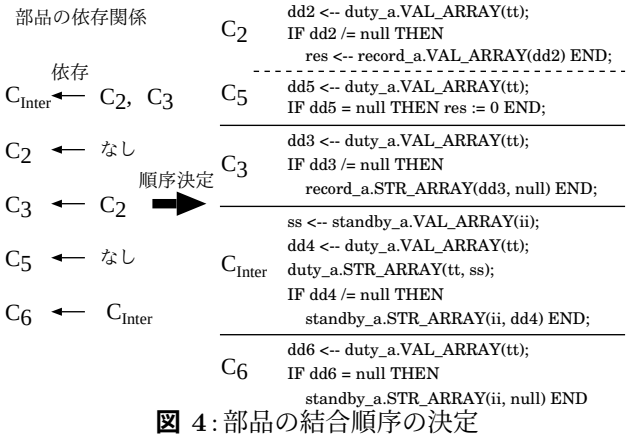

図 4: 部品の結合順序の決定

各代入文を持つ全ての IF 文に局所変数 X の値を参照させることで条件式を統一する。

以上を条件式 p の ELSE 節に関しても同様に行う。

4.4 ANY 文に対応する部品の結合

入力モデルに ANY 文がある場合の部品の結合手法を述べる。まず部品検索の結果、非決定的値の生成操作に対応する部品群 (C_{Def}) と非決定的値の参照操作に対応する部品群 (C_{Ref}) が発見される。C_{Def} の各部品はそれぞれ 1 つの非決定的値を実装するだけであり、部品間に依存関係は発生しない。一方、C_{Ref} の各部品は 4.3 節までの手順に従って結合する。

また C_{Def} が生成する値を C_{Ref} に渡すために、1 つの非決定的値に対応する C_{Def} の出力と C_{Ref} の入力の名前を一致させる。例えば非決定的値の名前が var であるとき、それに一意の識別子を付加した var_ANY を C_{Def} の出力名と C_{Ref} の入力名にする。そして C_{Def} と C_{Ref} を結合する際は、C_{Def} の出力 var_ANY を局所変数として再定義し、C_{Ref} にその局所変数の値を参照させる。

5 考察

小規模のモデルを用いて実験を行った。提案した粒度の部品を扱うことで、結合する部品の間に相互依存関係が発生しなかった。その結果、各部品が他部品の影響を受けない部品の結合順序が定まり、入力モデルを満たす合成実装を得られたと考えられる。また IF 文の条件式を統一することで、合成実装における冗長な条件式が排除されることを確認した。

6 おわりに

本研究では、合成実装の機能を保証するための部品の粒度と結合手法を提案した。今後はモジュール構造を考慮した部品の結合が課題である。

参考文献

[1] 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, 2014.

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