

パステスト向き有向グラフ簡約法に基づくテスト網羅率の効果

4K-2

中所 武司

(日立製作所システム開発研究所)

堀内 健二

(同左 武蔵工場)

1. はじめに

プログラムのテスト技法の一つとして、パステスト法がある。これは、プログラム内の実行パスの網羅性に注目するものであるが、実際は、これを簡略化して、プログラム内の全分岐のテスト実行をめざす分岐テスト法が用いられている。しかしながら、この方式では、全この分岐を対等に扱っているために、品質の過大評価や冗長なテストデータ選択などの欠点があった。

そこで、我々は、パステストに本質的な分岐だけに着目する新しい網羅率尺度 C_{pr} を考案したが、本方式で用いるパステスト向き有向グラフ簡約法およびその応用については既に報告した。今回は、Pascal プログラムを対象としたテスト網羅率 (C_{pr}) 判定ツール SCORE* を開発したので、その適用効果について述べる。

2. パステスト向き有向グラフ簡約法

本方式の概要を述べる。

まず、有向グラフの2つの3-7Pと8子に関する次の概念を導入する。

[定義] P を含む任意のパスが必ず8子を含む時、8子を P の相統子3-7と呼ぶ。

[定義] 他の3-7の相統子になりえない3-7を原始3-7と呼ぶ。

次に、ノード x からノード y への3-7 (x, y) が相統子となる条件を4種類示す。なお、 $IN(x)$, $OUT(x)$ はノード x の入力、出力3-7数、 w は隣接ノード、 $DOM(w)$, $IDOM(w)$ は w の支配子、逆支配子の集合、 A は3-7の集合である。

(R1) $IN(x) \neq 0 \wedge OUT(x) = 1$

(R2) $IN(y) = 1 \wedge OUT(y) \neq 0$

(R3) $OUT(x) \geq 2 \wedge x \in IDOM(w)$

但し、 $\forall w \in \{w | (x, w) \in A \wedge w \neq y\}$

(R4) $IN(y) \geq 2 \wedge y \in DOM(w)$

但し、 $\forall w \in \{w | (w, y) \in A \wedge w \neq x\}$

これら4条件を相統子消去規則として適当な順序で適用することにより、プログラムに対応する制御フローグラフを原始3-7のみで構成される相統子簡約グラフに変換できる。

表. 例題の有向グラフ簡約規則適用数

規則	R1	R2	R3	R4	原始3-7(PR)
適用数	348	40	5	10	81

① C_1 の対象3-7数: 136個 ($R1+R3+R4+PR$)

② C_{pr} の対象3-7数: 81個 (PR)

③ 比率(②/①): 60%

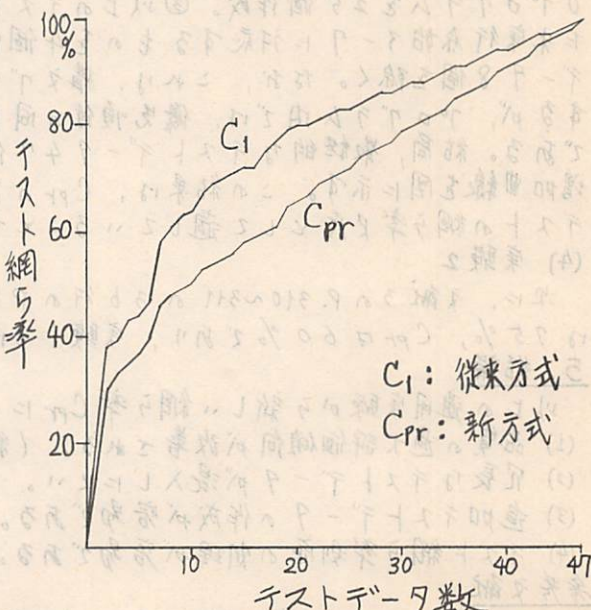


図. テスト網羅率の比較 (一例)

3. 新しいリスト網ら率 C_{pr} 測定ツール

従来は、全ての分岐ターフに注目したリスト網ら率 C_1 がよく用いられてきたが、ハスリストの観点では、相続ターフは被相続ターフ(上記 P)の網ら情報を相続するため、注目する必要はない。そこで、原始ターフのみに注目する新しい尺度 C_{pr} を導入し、その Pascal 用測定ツール SCORE を開発した。

$$C_{pr} = \frac{\text{実行済み原始ターフ数}}{\text{原始ターフの総数}} \quad (\text{参考: } C_1 = \frac{\text{実行済み分岐数}}{\text{分岐の総数}})$$

なお、本ツールは 47 エイズから成り、各々、次の機能を有する。① Pascal プログラムを制御フローグラフに変換、② 2 章の有向グラフ簡約法の適用、③ 被ハスプログラム上の原始ターフ対応箇所に測定コード埋込み、④ 実行結果に基づく C_{pr} の計算と未実行部分の検出。

4. 適用効果

(1) 例題プログラム

適用実験の例題として、N. Wirth の著書から PLO パーサ (P. 314 ~ 319) を選んだ。全体は 283 行で、誤り(1箇所)は修正した。

(2) 有向グラフ簡約法の効果

2 章の相続消去規則を適用して表の結果を得た。従来の C_1 で対象とする分岐ターフは 136 個あったが、そのうちの 55 個は $R2, R3, R4$ によって消去され、原始ターフは 81 個になった。結局、リスト対象ターフは 40% 削減された。

(3) 実験 1

リスト網ら率測定用テストデータは次の手順で作成した。① 文献上の PLO の構文グラフ上のハスを一つずつ網らしていくように 26 個の正しい PLO プログラムを作成。② 次に、エラー処理手続き error の引用箇所に対応する誤った PLO プログラムを 25 個作成。③ 以上のテストデータを実行後、本ツールで検出した未実行原始ターフに対応するものを 4 個作成。④ 最初の 26 個のうち、冗長なデータ 8 個を除く。なお、これは、構文グラフ上では何々の演算子毎にハスを有するが、プログラム内では、優先順位の同じ演算子ばかりで比較しているためである。結局、最終的なテストデータ 47 個を順に実行した時の C_{pr} および C_1 の増加曲線を図に示す。この結果は、 C_{pr} の方が C_1 より線形特性を示し、ハスリストの網ら率尺度として適していることを示している。

(4) 実験 2

次に、文献上の P. 310 ~ 311 の 36 行の PLO プログラムを実行した場合、 C_1 は 75%, C_{pr} は 60% であり、実験 1 と同じ結果が得られた。

5. 結論

以上の適用実験から新しい網ら率 C_{pr} について次のことが言える。

- (1) 品質の過大評価傾向が改善される。(線形特性に近い)
- (2) 冗長なテストデータが混入しにくい。(ハスリストに本質的分岐のみに注目)
- (3) 追加テストデータの作成が容易である。(対象とするターフが少ない)
- (4) テスト網ら率測定の処理が容易である。(測定箇所が少ない)

参考文献

- 1) 中野：ハスリストに本質的な分岐に着目した網ら率尺度の提案，情報論文誌 23.5, 545-552 (1982)
- 2) 中野他：ハスリスト向き有向グラフ簡約法とその基に於ける整枝法，情報処理学会，439-440 (1982)
- 3) N. Wirth: Algorithms+Data Structures=Programs, Prentice-Hall, Inc. (1976).