

7K-3

テスト網羅率測定の一方法

篠田 晃 小田英雄 中川 豊
NTT ソフトウェア研究所

1. はじめに

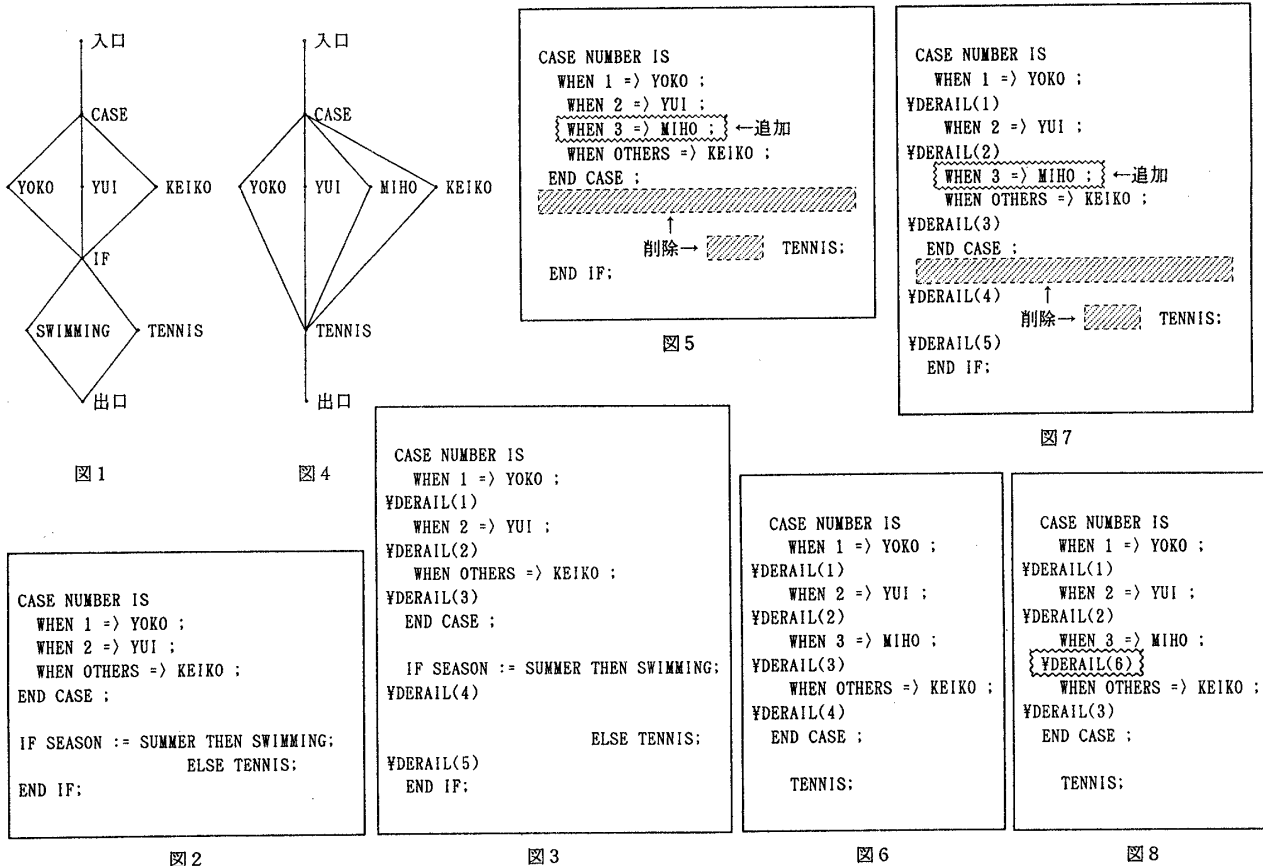
プログラムのデバッグ進捗を評価する方法として、バグ検出目標数を定めておき、それと比較して評価する方法と網羅率を測定して評価する方法がある。これまでのテスト網羅率測定は、オブジェクト生成時に測定ポイントを埋め込んでいたため、リコンパイルの都度、最初からやり直しており、作業量が多く実用には不向きであった。我々はソースプログラムに直接測定ポイントを埋め込む方法により、ソース修正前のテスト網羅率測定終了状態からテスト網羅率測定を継続する方法を考案した。この方法を取り入れたエッジ解析によるテスト網羅率測定ツールを開発したので報告する。

※：本来、ステートメントからステートメントへの制御の移行を示すものであるが、本ツールでは、C₁テストカバレージ指標に基づく解析を行っており、そのため、ある分岐ステートメントから次の分岐ステートメントへの制御の移行を示すものをエッジとした。

2. 従来方式の問題点

図1の構造を持つプログラム図2のテスト網羅率測定のため、図3のようにディレールが埋め込まれる。これを実行させると、ディレールを通過する都度、測定ルーチンでカウントが行われ、テスト網羅率が測定できる。

次に、なんらかの理由(仕様変更やバグ等)のため、図1から図4に構造を変更するため、図5のようにプログラムを修正したとする。従来方法では、その都度図6のようにディレールの番号が順番に与えられてしまう。つまり、ディレール番号とエッジの対応が、前回のテストと異なってしまうため、ソースプログラム修正前のテスト網羅状況を生かすことができず、測定を最初から繰り返すことになる。



3. ソースプログラム修正部分の把握

前回のテスト網羅状況を利用するためには、ソースプログラム修正前の情報を利用して未修正部分のエッジを抽出することにより、修正部分と未修正部分を把握することが必要である。ソースプログラム修正前の情報は、ディレール設定位置とディレール情報（本ツールでは、主として番号）である。これらの情報を利用する条件として、ディレール設定位置を固定することとエッジを一意に識別することが必要となる。

(1)ディレール設定位置を固定する方法

ディレール設定位置を固定するため、ディレールをソースプログラムに直接埋め込むことにした。このため、ディレールの埋め込まれたソースプログラムが修正対象となる。

(2)エッジを一意に識別する方法

ディレールは、各エッジの先頭に埋め込む。ディレール番号をエッジに対し個々に与えることによりエッジを一意に識別できる。個々のディレールに対応しているディレール情報は、ユーザの意思で削除しない限り残すことにより、前回までの網羅状況を生かすことができる。

(3)未修正部分のエッジを抽出する方法

ソースプログラム修正に際し、修正部分にディレールがある場合には、これを取り除いておくことにより、自動的にディレールが再設定される。新たに設定されたディレール、すなわち、修正エッジに対しては、新しい番号が設定される。これにより、未修正エッジの抽出が可能となる。

4. 測定方法

本ツールの測定方法を図9に示す。

- (1)ソースプログラム解析：エッジ解析によりディレール設定箇所/不要箇所の抽出を行う。
- (2)ディレール設定：自動及び手動によりディレールを設定する。
自動の場合：ソース解析結果に基づいたエッジへのディレール設定と無効ディレールの削除
手動の場合：重点箇所および実行文にディレール設定が必要な場合に使用する。
- (3)開発支援データベース：開発情報を収集・管理・利用するデータベースであり、これを使用し、ディレール情報とテスト網羅率情報を収集、管理している。

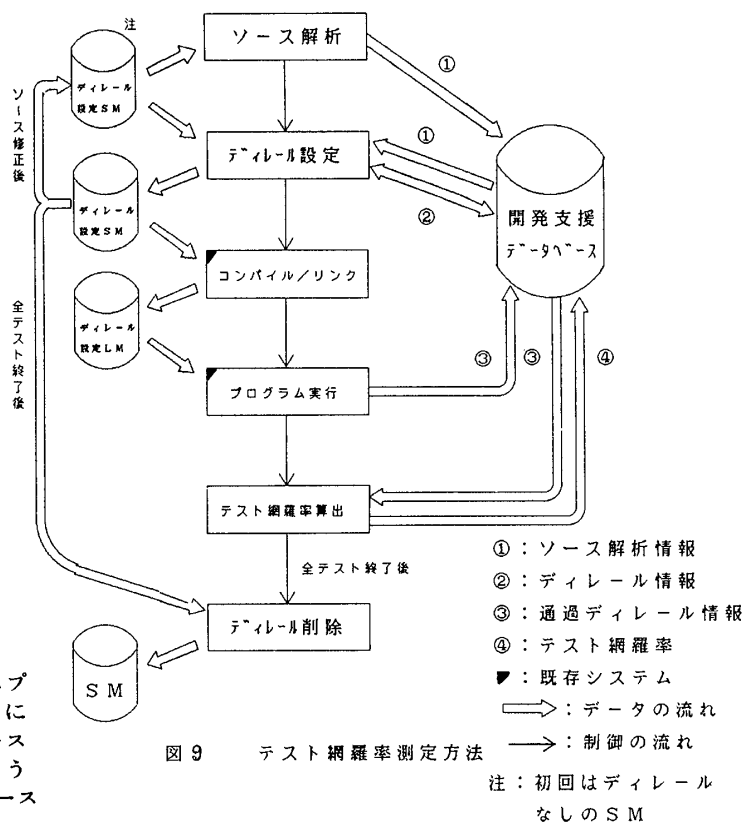


図9 テスト網羅率測定方法

5. 効果

本ツールを使用した場合、図3のソースプログラムを直接修正するので、図7のようになる。次にテスト網羅率測定のため、ソース解析を行いディレール設定すると図8のようになる。(YDERAIL(4), YDERAIL(5)は、ソースプログラム解析で不要と判断され、ディレール設定で削除される。) 図5では、追加された”WHEN 3 => MIHO ;”にYDERAIL(3)が与えられているが、これは元のソースプログラムでは、”WHEN OTHERS => KEIKO ;”のディレールである。また、YDERAIL(4)に対しても元の位置ではない。

図8では、YDERAIL(1), YDERAIL(2), YDERAIL(3)の位置は変わらず、追加部分に対しては、新規にYDERAIL(6)が与えられるため、YDERAIL(1), YDERAIL(2), YDERAIL(3)に対しては前回のテスト網羅状況が利用できる。

今、図3の状態ですべてのテスト網羅率が100%とする。図6のYDERAIL(1), YDERAIL(2), YDERAIL(3), TENNISはテスト済みである。YDERAIL(6)は新規に追加された部分であり、まだテストデータは流されていない。この場合、本ツールでは、図6の実行の時点でテスト網羅率75%の状態からスタートすることができる。また、テストの重点箇所はYDERAIL(6)の部分にすればよい。

6. 今後の課題

従来のテスト網羅率測定ツールとの比較を行い、本テスト網羅率測定ツールの有効性を確かめる。