

テスト条件生成支援ツールの開発 CEGTest

－原因結果グラフ技法を利用したテスト設計－

2010年1月28日
ニフティ株式会社
加瀬 正樹

はじめに

- 名前 加瀬 正樹
- 所属 ニフティ株式会社 <http://www.nifty.co.jp/>
- 業務 会員向けメールサービスの開発・運用
- その他

TEF(ソフトウェアテスト技術者交流会)

<http://www.swtest.jp/wiki/index.php>

WACATE(ソフトウェアテストワークショップ)

<http://wacate.jp/>

目次

1. 組合せテスト
2. 原因結果グラフ技法
3. 支援ツール「CEGTest」の紹介
4. 効果測定
5. まとめ
6. 今後の取り組み

組合せテスト

- 複数の入力条件下で発生する欠陥を検出するために実施するテスト
- 入力条件数が多くなれば、組合せパターンは**爆発的**に増加する
- さまざまなテスト設計技法を利用して、できるだけ少ないテストケースで組合せを網羅する

直交表系	デシジョンテーブル系
直交表 HAYST法® All-Pair法	デシジョンテーブル 原因結果グラフ技法 CFD法

原因結果グラフ技法

入力やイベント(=原因)の組合せと、出力(=結果)との関係をグラフ化し、デシジョンテーブルを作成するテスト設計技法。論理関係の網羅性を高めることができる。

1. 仕様から原因結果グラフを作成
2. 原因結果グラフからデシジョンテーブルを作成
3. デシジョンテーブルからテストケースを作成

CEG作成

DT作成

TC作成

原因結果グラフ技法はムズカシイ

- ✓ 仕様から適切なグラフが作成できない！
- ✓ デシジョンテーブルの作成ができない！
- ✓ 誤りがないか検算しなくてはならない！
- ✓ 無償のテストツールが見当たらない！



仕様から適切なグラフが作成できない

- 原因結果グラフ技法で最も重要な作業
- グラフの質は作成者のスキルに依存しやすく、最終成果物（テストケース）の質を左右する
- 制約（特にREQやMASK）が難しい

REQ Aが真であるためにはBが真であることが必要

MASK Aが真ならばBは真偽が決定できない

CEG作成

DT作成

TC作成

デシジョンテーブルの作成ができない

- お手本が少ない、難しい
三賢者、テストを語る (JaSST'07 Tokyo)
ソフトウェア・テストの技法 (G.J. Myers)
- ノードが多かったり、論理関係が複雑になると
手作業でのデシジョンテーブル作成が困難
- 原因結果グラフを再作成したら、デシジョン
テーブルも再作成しなくてはならない

CEG作成

DT作成

TC作成

誤りがないか検算しなくてはならない

- 手作業だと間違える可能性がある
- グラフの見直しによって、デシジョンテーブルを再作成するので、そのたびに検算する必要がある

CEG作成

DT作成

TC作成

無償のテストツールが見当たらない

- 調べてみると、、、
AGENT (情報処理学会 1980)
SVA (情報処理学会1988)
Bender RBT
(<http://benderrbt.com/bendersoftware.htm#rbt>)

ツールがないなら、自分で作ってしまおう

CEG作成

DT作成

TC作成

支援ツール「CEGTest」の紹介

- Webブラウザで操作できる
- キーボードからの表入力ではなく、直接マウスで原因結果グラフが描ける
- デシジョンテーブルは自動生成
- 原因結果グラフを編集しながら、自動生成されたデシジョンテーブルが確認できる

CEG作成

DT作成

TC作成

支援ツール「CEGTest」の紹介

実際にツールを使ってみよう

例) JaSSTに参加する条件は？

効果測定

CEGTestを利用することでどの程度の**時間短縮**ができるか検証する

1. CEGTestを利用して、原因結果グラフの作成とデシジョンテーブルの作成を行う
2. Excelを利用して、デシジョンテーブルの作成を行う
3. 方法1と方法2とで比較・評価を行う

効果測定

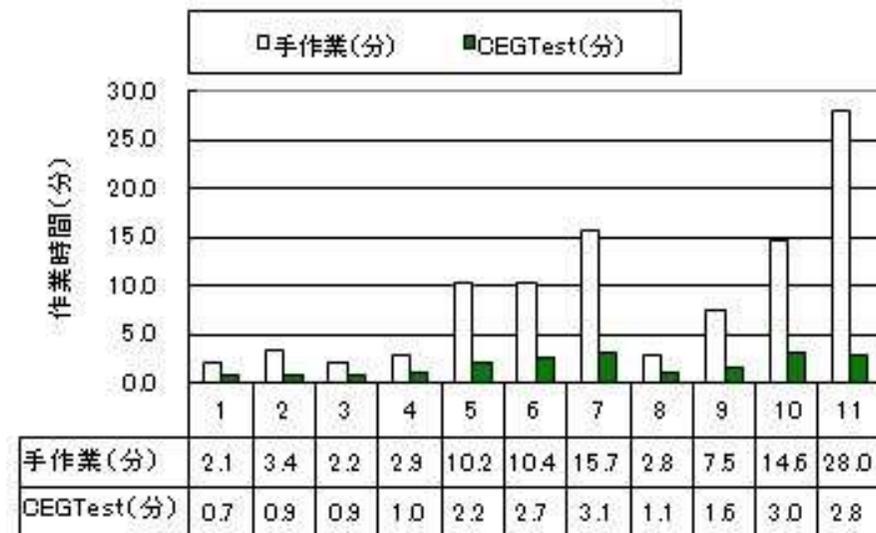
論理関係をもつテスト対象11個を使って検証した

No	ノード数	集合の制約数	順序の制約数	論理展開式数	テスト数
1	4	1	0	6	3
2	8	1	0	9	4
3	5	0	0	6	3
4	6	0	0	7	5
5	11	0	0	15	5
6	12	0	0	18	4
7	10	2	0	23	6
8	5	1	1	6	3
9	7	2	2	8	3
10	13	1	2	22	6
11	14	3	0	25	8

評価

- CEGTestの利用で 約80% の時間短縮
- ノード数が多いほど効果が高い
- 論理展開式が 20 を超えると変換誤りしやすい

より多くの時間を分析・
グラフ作成にまわせる



まとめ

- 組合せテストの設計技法として原因結果グラフ技法があるが、扱いにくい点がある
- Webブラウザベースの支援ツール「CEGTest」を開発し、組合せの計算処理を自動化
- 自動化で時間短縮した分を、グラフ作成・修正にあてることができた

今後の取り組み

グラフ作成時
の工夫

テスト実施との
連携

制約に関する
テスト

CEG作成

DT作成

TC作成

ご清聴ありがとうございました

CEGTest

<http://softest.cocolog-nifty.com/labo/CEGTest/>

問い合わせ

kase.masaki@gmail.com