

テスト環境の変更コストを削減するテストケースの生成手法 The Test Case Generation Method for Reducing the Change Costs of Test Environment

橋本 裕介[†] 酒井 三四郎[†]
Yusuke Hashimoto Sanshiro Sakai

1. はじめに

ソフトウェアを開発するにあたり、テストは開発工程の多くを占める。そのため、テストの効率化は工数の短縮や品質の改善に繋がる。テスト工数を減らす手法としては同値分割のようなテストケース数を減らす手法が知られているが、テストの準備を効率化する方法はあまり議論されない。テスト環境の変更を伴うテストではそれにかかる時間や労力を考慮する必要があり、既存のテスト手法ではあまり考慮されない。テストの中でも自動化できない UI テストやシステムテストでは多くのコストが必要となる。

本研究では一定の信頼性を持つテストを効率的に実行するために、コストを考慮したペアワイズなテストケースセットを作ることを考える。テストケース間におけるテスト環境の変更にかかるコストを数値化することでコストの概念を抽象化する。テストケースを生成する際にこのコストを考慮することでテストにかかるコストを削減し、テスト工程を効率化する。

2. 関連研究

紀本らの研究[1]では、テストの実行にかかるコストを定義し、2つの手法でペアワイズテストのコスト削減を試みた。1つ目の手法ではテストケースを生成する際に、テストケースの数だけでなくコストを考慮しながら生成する。2つ目の手法では生成済みのテストケースセットを並び替えることでテスト実行のコストを削減する。

Czerwonka[2]によると、ペアワイズなテストケースを作る際、テストケース生成を2段階に分けることで特定のパラメータ間の組み合わせを減らす手法がある。この生成手法は Microsoft のペアワイズテストケース生成ツールである PICT に実装されている。ただし、この手法ではテストケースの数は増加する。

紀本らの2つ目の手法では変更コストの大きいパラメータを優先的に考慮するが、全体的なコストの大きさは考慮されない。本研究ではこの手法を改良し、より効率の良いソート手法を提案する。

また、Czerwonka による手法を用いて変更コストの大きいパラメータ間の組み合わせ数を減らすことにより、変更コストを削減することが考えられる。この手法で生成したテストケースを評価実験に含めることで効率的な生成手法を考察する。

3. 提案手法

3.1 変更コストの定義

あるテストケースを実行して次のテストケースに移る際、2つのテストケース間で異なるテスト環境を変更する必要がある。変更にかかる労力や時間はテスト環境、つまりテストケースのパラメータによって異なり、本研究ではこれを変更コストと呼ぶ。例えばハードウェアの接続やネットワークの設定を変えるのはコストが大きい、ソフトウェアの設定をひとつ変えるコストは小さく済む。このようにテストパラメータに変更コストを定義して変更回数、特に変更コストの大きいパラメータの変更回数を減らすことでテスト工程の効率化を図る。自動化によって手動変更の必要がない場合や、前のテストケースにかかわらず常に一定のコストがかかる場合の変更コストは0とする。例えば毎回キーボードから数値を入力する場合は、労力が前のテストケースに依存しないので変更コストは0である。

3.2 変更コストの数値化

紀本らの研究をもとに、変更コストの算出方法を定義する。 k 個のパラメータを持つテストケースが N 個あり、 n 個目のパラメータの変更コストが C_n であるとき、 $i-1$ 番目のテストケースから i 番目のテストケースへの変更コストを $Cost(i)$ と定義する。ただし、2テストケース間で j 個目のパラメータの値が同じとき $r_j=1$ 、異なるとき $r_j=0$ とする。

$$i = 1 \text{ のとき } Cost(i) = \sum_{j=1}^k C_j$$

$$i \geq 2 \text{ のとき } Cost(i) = \sum_{j=1}^k C_j \cdot r_j$$

また、このときのテストケースセット全体の変更コスト C を次のように表すことができる。

$$C = \sum_{i=1}^N Cost(i)$$

3.3 貪欲法を用いたソート手法

この手法では、作成済みのテストケースセットを整理することでテスト実行にかかる総変更コストの削減を図る。最善の並べ方を求めるには全並べ方を試して比較する方法があるが、テストケース数が多くなると莫大な時間がかかってしまう。ここでは貪欲法を用いることで現実的な時間で良い並べ方を探す。この手法では n 個の整理済みテストケースがあるとき、 n 番目の整理済みテストケースから最も少ない変更コストで済む未整理テストケースを $n+1$ 番目として整理済みテストケースセットに加える。このように先頭から順に、その場その場で最善に見える並べ方で並べていく貪欲法のアルゴリズムでテストケースをソートし、総変更コストの小さいテストケースセットを作る。

[†] 静岡大学大学院総合科学技術研究科 Shizuoka University Graduate School of Integrated Science and Technology

表 1 パラメータに対するコストの割り当て

条件	コストの割り当て		
	パラメータ数3	パラメータ数5	パラメータ数7
差が小さい	1, 2, 3	1, 2, 3, 4, 5	1, 2, 3, 4, 5, 6, 7
差が大きい	1, 4, 8	1, 2, 4, 8, 16	1, 2, 4, 8, 16, 32, 64
すべて同じ	1, 1, 1	5, 5, 5, 5, 5	7, 7, 7, 7, 7, 7, 7
0を含む	0, 2, 3	0, 0, 3, 4, 5	0, 0, 0, 4, 5, 6, 7

4. 評価実験

パラメータに適当な変更コストを設定し、PICT で生成したペアワイズなテストケースセットをソートすることで総変更コストを計算した。テストケースの生成手法は PICT による通常の手法とパラメータを階層分けして組み合わせ数を減らす手法の 2 種類、ソート手法は紀本らによる単純なソートと本研究で提案する貪欲法ソートの 2 種類、計 4 パターンの手法で実験した。パラメータを階層分けする手法では、変更コストの大きい 3 パラメータ間の組み合わせ数を減らすことで総変更コストの削減を試みる。

テスト対象は本研究の対象であるテスト環境の変更を伴うテストを想定し、パラメータ数は 3, 5, 7 の 3 パターン、水準数は 3 または 5 の 2 パターンとして計 6 パターンのテスト対象で実験した。ただし、どのパラメータも水準数は同じものとする。パラメータへのコストの割り当ては表 1 に示すように差の大小で 3 パターン、変更コスト 0 のパラメータを含む 1 パターンの計 4 パターンの割り当てを用意した。

5. 実験結果

5.1 結果

パラメータ数 5, 水準数 3 の場合のコストの割り当て及び生成手法による総変更コストを比較したグラフを図 1 に示す。テストケース数は通常の生成では 12, 階層分けを使うと 30 である。また、パラメータ数 7, 水準数 5 の場合のものを図 2 に示す。テストケース数は通常の生成で 40, 階層分けで 132 である。手法ごとの総変更コストの大小関係は、図に示した 2 パターン以外でも同じような傾向が見られた。

5.2 考察

今回実験したすべてのコスト割り当てや生成手法に対して、貪欲法ソートによる総変更コストは単純なソートによるものよりも小さくなった。そのため、貪欲法ソートはよりコスト削減効果が大きいといえる。

パラメータを階層分けする手法は多くの場合、総変更コストを増加させてしまった。これは生成の際にテストケース数が増えてしまい、それに伴って総変更コストも増加したと考えられる。しかし、コスト 0 のパラメータを含む場合に関しては通常の生成手法よりも総変更コストを小さくすることがあった。この場合もテストケース数は多くなっているため、実際のテストでは総変更コストとテストケース数の両方を考慮してテストケースセットを選択する必要がある。

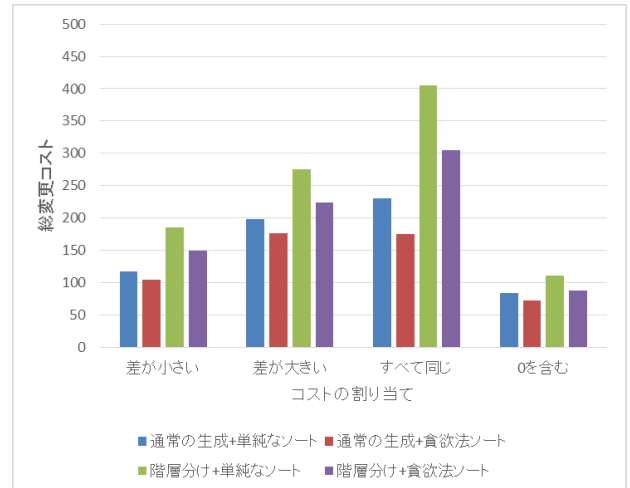


図 1 パラメータ数 5, 水準数 3 の結果

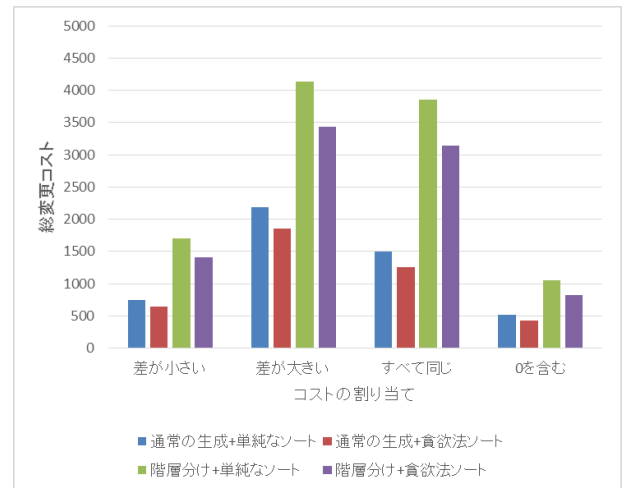


図 2 パラメータ数 7, 水準数 5 の結果

6. おわりに

本研究ではテストの準備にかかるコストの削減を目的とし、テストパラメータにコストを設定してコストの総和が小さくなるようにテストケースを整列してコスト削減を試みた。貪欲法を用いたソート手法により、テストケース数を変えずにパラメータの変更コストを削減することができた。しかし、今回の手法ではテスト環境の変更にかかるコストの削減だけを目的とし、テストの実行にかかるコストなどは考慮していないため、実際のテストで活用していくためにさらなる改善ができると考えている。

参考文献

- [1] 紀本 眞, 土屋 達弘, 菊野 亨, “テスト実行コストを考慮したペアワイズテストセット生成法の提案”, 電子情報通信学会技術研究報告. DE, データ工学, Vol.107(254), pp.47-50, (2007).
- [2] Jacek Czerwonka, “Pairwise Testing in the Real World: Practical Extensions to Test-Case Scenarios”, <http://msdn.microsoft.com/en-us/library/cc150619.aspx>, (2008), (2016/02/16 アクセス).