

タグを用いたリスクベースドテストの効率化

西森 雅峰[†] 金田 重郎[†] 芳賀 博英[†] 井田 健太[†] 佐々木 亮太[†]

同志社大学インテリジェント情報工学科[‡]

1. はじめに

近年のソフトウェア開発の分野では、開発規模は大きくなる一方で短納期かつ高品質なシステムづくりが要求されている。しかし品質を上げるために欠かせないテストをすべて実施することは不可能であり、リスクの高い箇所に優先的にテストを配置する必要がある。

そこで本稿では、リスクベースドテストでのリスクマトリクスの「バグの起こりやすさ」部分に着目し、数量化Ⅱ類を用いたリスクマトリクス作成時間の短縮およびバグ検出率、バグ効率を向上させる一手法を提案すると同時に、その有効性の検証結果を報告する。

2. 研究分野の概要とその問題点

2.1. 概要

リスクベースドテストでのリスク (r) は、「もしバグがあった場合の影響度」(d) と「バグの起こりやすさ」(p) の積 ($r = p * d$) により計算される。この d および p を計算する一手法として d および p に対応するそれぞれ m, n 個のリスク因子とテスト対象項目の関係を表形式で表したリスクマトリクスというツールを使って計算する方法がある。

2.2. リスク値計算式

リスクベースドテストでのリスクマトリクスを用いた項目 i のリスク値計算 r_i は、次の演算式 (1) で示される。

$$r_i = wp \cdot vp_i * wd \cdot vd_i \quad (1)$$

ここで wd, vd_i はそれぞれ d を求めるためのリスク因子の重みおよびそのリスク因子の項目 i の評価値を表す m 次元ベクトル、 wp, vp_i は、 p を求めるためのリスク因子の重みおよびそのリスク因子の項目 i の評価値を表す n 次元ベクトルである。

2.3. 従来手法の問題点

従来手法では、

- (a) リスク因子を必ず評価する必要がある。
- (b) 評価理由が他人にわかりづらい。
- (c) リスクの再評価を行いづらい。
- (d) バグの発生確率は、リスク因子の評価値順になるとは限らない。
- (e) リスク因子の評価をすべて固定したランク (大中小や 1~5 など) で評価しなければならない。

という課題があった。

3. 提案手法

3.1. 提案手法

前述の問題点への一つのアプローチとして、リスク因子をランクで評価するのではなく、数量化Ⅱ類で重みをつけた質的データ (以降タグと表記) を用いることで「バグの起こりやすさ」部分のリスクマトリクスを構成する。さらに、項目ごとにリスク因子を評価しやすいものだけ評価することで、高速化を図る。具体例としては、「プログラムの複雑さ」、「納期までの猶予度合い」が p のリスク因子として使われているとする。従来手法では、この評価項目として大中小が用いられていた。一方本研究手法ではタグとして「プログラムの複雑さ」では“複雑”, “少し複雑”, “単純”, “不明”, 「納期までの猶予」では“短い”, “不明”を採用することで、従来手法よりリスク因子を評価しやすくしている。このことにより、評価理由が他人にもわかりやすく、結果としてリスクの再評価も行いやすくなる。

さらに、従来の手法では、評価値の順と「バグの起こりやすさ」の順は異なる場合があった。たとえば、「プログラムの複雑さ」が複雑な順にバグの発生確率が高いとは限らない。この問題点もスコアを用いることで解決している。

3.2. 数量化Ⅱ類

数量化Ⅱ類とは、説明変数が質的データで与えられ、この質的データから、質的データである外的基準を求める判別分析方法である。

The Efficiency Improvement of Risk Based Testing With Tag

[†]Faculty of Engineering, Doshisha University

[‡]Masataka Nishimori, S Kaneda, H Haga, K Ida, R Sasaki

3.3. スコア算出法

今回は、リスク因子を説明変数、テスト結果でのバグの有無を基準変数として、テストデータにタグ付けを行い、無作為に 50 件テストデータを抽出し数量化Ⅱ類を用いてカテゴリースコアを求めその値をタグの重みとした。

本研究手法での p はテスト対象項目ごとに用いられているタグの重みの項目ごとの合計点として、「バグの起こりやすさ」とした。

この重み係数をもとに、テスト対象項目ごとに p を求め、影響度 d は従来手法を用いて求め、式(1)を用いてリスク値を求めた。図 1, 2 に横軸をテスト数、縦軸を従来手法および本手法で求めたリスク値の分布を示す。

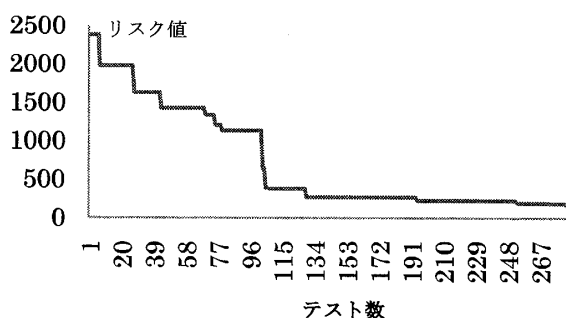


図 1: 従来手法でのリスク分布

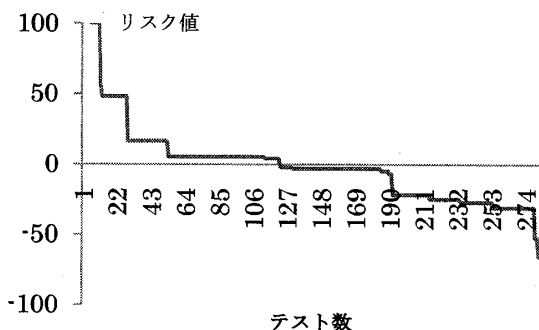


図 2: 本研究手法でのリスク分布

4. 適用結果とその考察

今回は、全テストケースを実施した場合の通常のテストと、従来手法、本研究手法の 3 通りで検証を行った。

検証項目は、リスクマトリクス作成時間比、バグ検出率 (バグ数/全バグ数)、バグ効率 (Σ 検出バグの影響度/テスト数) である。

リスクマトリクスの作成時間比は従来手法の約 86%であった。これは、従来手法では判断基準が多岐に渡るので評価に時間がかかっていたのに対し、本研究手法は評価基準が単純である分だけ、結果として時間が短縮されたのだと考え

られる。

図 3, 4 に横軸にリスク値の高い順に優先付けたテスト、縦軸にそれぞれバグ検出率、バグ効率とした場合の本研究手法、従来手法、通常テストでの結果を示す。

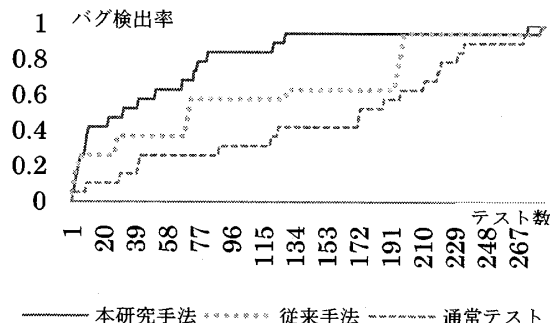


図 3: 各手法でのバグ検出率とテストケースの関係

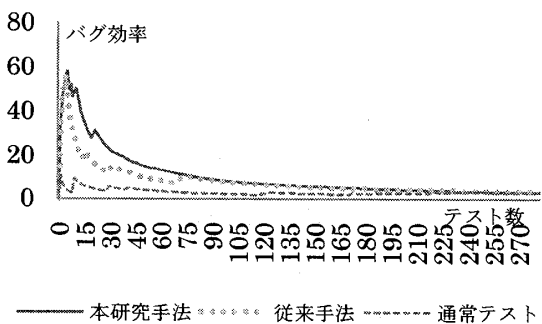


図 4: 各手法でのバグ効率とテストケースの関係

バグ検出率、バグ効率ともに本研究手法が従来手法を上回っている理由としては、多元的リスク因子に対してうまく重み付けができたためと考えられる。

5. おわりに

今後は、バグ効率向上のためにスコア導出法の改良と、項目へのある程度の自動的なタグの振り分けによるさらなるリスクマトリクス作成時間の短縮を図るつもりである。

参考文献

- [1] 石田智亮, 石山康介, 米田征弘, 西康晴, "Risk Based Testing の実践方法と適応事例からの考察", <http://www.jasst.jp/archives/jasst08e/pdf/C2-1.pdf>
- [2] S Amland, "Risk Based Testing and Metrics", EuroSTAR 99, Barcelona, Spain, 1999 - amland.biz
- [3] LH Rosenberg, R Stapko, A Gallo, "Risk-Based Object Oriented Testing", - 24th SWE, 1999 - Citeseer
- [4] James Bach, "Heuristic Risk-Based Testing", First published in Software Testing and Quality Engineering Magazine, 11/99