

C++ソフトウェアにおけるバグとメトリクスデータとの相関

5ZC-9

篠原 英二

Eiji SHINOHARA

浜田 信

Makoto HAMADA

山田 哲靖

Tetsuyasu YAMADA

NTT ネットワークサービスシステム 研究所

NTT Network Service Systems Laboratories

1. はじめに

一般的にソフトウェア規模と発生バグ数には強い相関が有ると言われている。しかし、バグの発生要素には規模のみでなく複雑性が関連するということがこれまでも報告されている(1)(2)。

本稿では改版されたソフトウェアでのバグ数と複雑性を表すソフトウェアメトリクス(以下、メトリクスとする)の関連について分析を行い改版ソフトウェア対してのメトリクスの有効的な利用手段について報告する。

2. 相関調査(1)

C++言語により開発されたあるソフトウェアの初期開発ソフトウェアと改版ソフトウェアに対して、発生したバグ数を規模及び複雑性を表すメトリクスとの相関を調査した。

調査したソフトウェアは、大規模交換ソフトウェアである。

また、今回適用したメトリクスは以下の2つである。以下にその特性を示す。

(a) MTH (クラス内メソッド数)

クラスに含まれるメソッド数を示すメトリクスである。

(b) WMC (クラス内メソッドの複雑さの総和)

クラスに含まれるメソッドの複雑度を示すメトリクスである。

調査結果を表1に示す。

表1.バグ数と規模/メトリクスの相関(1)

	規模	MTH	WMC
初期開発ソフトウェア	0.753	0.591	0.691
改版ソフトウェア	0.586	0.592	0.651

表1は初期開発ソフトウェアと改版ソフトウェアそれぞれで発生したバグ数と規模/MTH/WMCの3項目についての相関を示したものである。

初期開発ソフトウェアでの相関を見ると、規模およびWMCで相関値が約0.7を示している。

これに対して、改版ソフトウェアではMTH、WMCの相関は、初期開発ソフトウェアとほとんど同じ値を示しているが、規模との相関値が約0.6と低下している。

以下で改版ソフトウェアで規模との相関が低下したことを考察する。

3. 考察

今回調査した改版ソフトウェアは初期開発ソフトウェアに機能追加、改造を加えたものである。

本改版ソフトウェアの特徴としてはオブジェクト指向言語であるC++により開発しているため、「継承」を利用した差分プログラミングにより変更するソース量が少ないことが挙げられる。

このため、改版ソフトウェアでは初期開発ソフトウェアに比べてバグ数と規模との相関が低下したと考えられる。

また、改版するに従い、新たな機能の盛り込み等により、ソフトウェアの複雑性は増加していくものと考えられることから、バグ数とメトリクスの相関は高くなると考えられる。

これら改版ソフトウェアの特徴を考慮した場合、ソース規模の比較的小さいクラスでのバグ数と規模/メトリクスの相関を調査する必要があると考える。

4. 相関調査(2)

3項での考察を踏まえ、初期開発ソフトウェア/改版ソフトウェアそれぞれに対してソース規模の比較的小さい(2kl以下)クラスでのバグ数と規模/メトリクスとの相関を調査した。

相関値を表2に各メトリクスとの相関図を図1～図4にそれぞれ示す。

表2.バグ数と規模/メトリクスの相関(2)

	規模	MTH	WMC
初期開発ソフトウェア	0.518	0.249	0.391
改版ソフトウェア	0.466	0.622	0.578

表2は表1と同じく、初期開発ソフトウェアと改版ソフトウェアについて、発生したバグ数と規模/MTH/WMCの相関を表したものである。

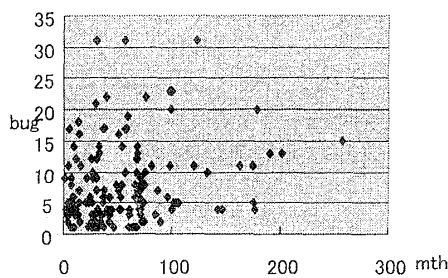


図1 バグと mth との相関(初期開発)

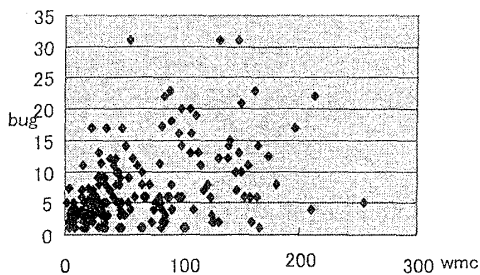


図2 バグと wmc との相関(初期開発)

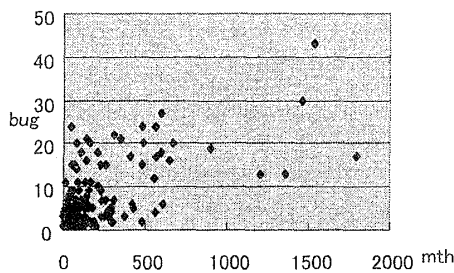


図3 バグと mth との相関(改版)

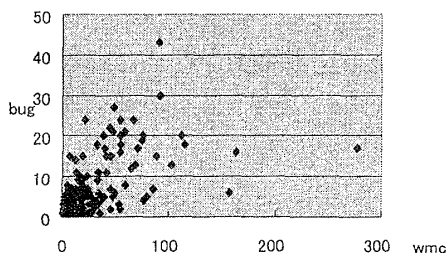


図4 バグと wmc との相関(改版)

初期開発ソフトウェアでは、規模/MTH/WMCのいずれの項目も表1に比べ、相関値が大きく低下している。

この理由は、改版ソフトウェアでは規模が小さくなるため、初期開発ソフトウェアよりバグ数と規模の相関が表出されにくくなるためと考える。

一方、改版ソフトウェアでの各項目の相関を見ると規模との相関では、初期開発ソフトウェアを下回っているが、MTHとWMCでは逆に初期開発ソフトウェアの相関値を上回っている。

これについては、改版ソフトウェアにおける複雑性の増加により、規模にあまり関係することなく個々のクラスのメトリクスが表出したためと推定している。

5. まとめ

本稿では以下のことが明らかとなった。

- (1) 改版ソフトウェアは規模が小さいために、初期開発ソフトウェアのように規模からバグ数を推定する方法はあまり有効ではない。
- (2) 改版ソフトウェアは規模が小さいことにより、クラスのメトリクスが規模で平均化されことなく表出しやすい。
このため、バグ数を推定する場合には、クラスの特徴を表すMTH、WMCといったメトリクスを用いた方法が有効であると考えられる。

6. おわりに

C++言語により改版されたソフトウェアにおける、バグとソフトウェアメトリクスとの相関についての調査結果を報告した。

今後は、今回調査できなかったメトリクス(たとえばクラス間の結合)について分析していきたい。

参考文献

- (1) 石井 久美他、「メトリクスデータによるオブジェクト指向ソフトウェアの評価」、1998年信学技報 SSE-98-124、1999.
- (2) 篠原他、「バグの影響範囲とプログラムの複雑度との相関」、1999年信学会総合大会 B-6-70、1999.