

ソフトウェア信頼度成長モデル の適用に関する一考察

3M-3

松沢 寿典 菅野 文友
東京理科大学工学研究科

1.はじめに

システムの信頼性向上のためには、ハードウェアとソフトウェアの両面から信頼性の向上を図る必要がある。ハードウェアは、信頼性の向上と大幅なコストの低減が図られてきたが、ソフトウェアは様々な面で立ち遅れている。

そこで、ソフトウェアをライフサイクル全体でとらえ、高品質のソフトウェアを効率良く実現することが重要視されている。そのためには、ソフトウェア開発の各段階における有効な技術の開発と同時に、ソフトウェア信頼性の評価方法に関する研究が進められている。その一つに、テスト工程でバグによって引き起こされるソフトウェア故障の時系列現象の側面から、ソフトウェア信頼性をとらえるソフトウェア信頼度成長モデルがある。しかし、このモデルを適用する場合、いくつかの問題点がある。

2.モデルの適用における問題点

モデルの適用にあたっては、さまざまな仮定が設けられているため、適用領域が狭いという問題点がある。また、テスト実施時には、予めテスト項目を設定して行なうのが一般的であるが、信頼度成長モデルでは、テスト項目の質、量、およびテストの進め方などがまったく考慮されていない。さらに、実際にモデルを適用してみると、一度に多量のバグが出たり、反対にしばらくバグが出ずにテストが停滞したりするとバグ数の予測値がふらつき、どこが飽和点かの判断がしにくい場合がしばしばある。

3.テスト項目の追加時におけるモデルの取り扱い

ここでは、テスト実施時にテスト項目が追加された場合についてどう取り扱うかを考えてみた。まず、あるプログラムについて最初 N_0 個のテスト項目で、テストを実施していたとする。そして、ある時刻 t_1 で、テスト項目の数を N_1 個追加したときのことを考えてみる。ここでは信頼度成長モデルは、Goel-Okumoto NHPP を用いて総バグ数を推定すると、 t_1 までのバグの推定値 $Y(t)$ は (1) 式のようなになる。ところが、 t_1 以降のテスト効率を考えてみると、 t_1 以前のときの $N_0 / (N_0 + N_1)$ 倍となっている。したがって、 t_1 以降の N_0 個のテスト項目で検出できるバグの推定値は (2) 式のようなになると考えられる。

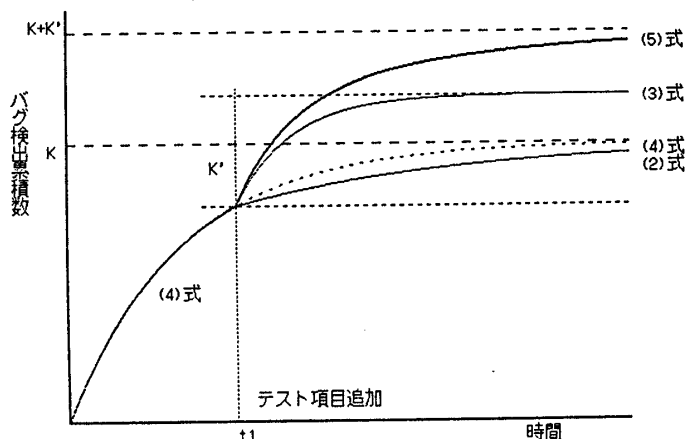


図1 テスト項目の追加時におけるモデルの取り扱い

On the software reliability growth models
Hisanori MATSUZAWA, Ayatomo KANNO
Faculty of Engineering, Sience University of Tokyo

$$Y(t) = K \{1 - \exp[-\phi t]\} \quad (1)$$

$$Y(t) = K \{1 - \exp[-\phi \cdot \{t_1 + (t - t_1)N_0 / (N_0 + N_1)\}]\} \quad (2)$$

また、追加した N_1 個のテスト項目に対しては、 t_1 からテストを実施したものと考えて、 t_1 以降の計測したバグ数から (2) 式で求めたバグの推定値を引いた残りのバグ数をもとに、パラメータを推定する。その場合、追加した N_1 個のテスト項目で検出できるバグの推定値は、次のようになる。

$$Y(t) = K' \{1 - \exp[-\phi'(t - t_1)]\} \quad (3)$$

したがって、時刻 t でのバグ推定値 $Y(t)$ は、表される (図 1)。

$$Y(t) = K \{1 - \exp[-\phi t]\} \quad (0 \leq t \leq t_1) \quad (4)$$

$$Y(t) = K \{1 - \exp[-\phi \{t_1 + (t - t_1)N_0 / (N_0 + N_1)\}]\} + K' \{1 - \exp[-\phi'(t - t_1)]\} \quad (t_1 < t) \quad (5)$$

4. ソースコードを考慮したモデルの取り扱い

ここでは、対象となるソースコードについて考えてみる。例えば図 2 のようなソースコードに対してテストを行なうと仮定すると、④に致命的なバグがあった場合は、④以降のソースコードに対しては、④が修正されるまでテストが行なえない。したがって、最初のテストでは、100ステップに対してだけ、行なっていることになる。そして④が修正されると、残りの 200ステップに対しては、この時点からテストを開始したと考えられる。そしてテスト項目数にステップ数を対応させると、前記のテスト項目を追加したときと、まったく同様に考えることができる。

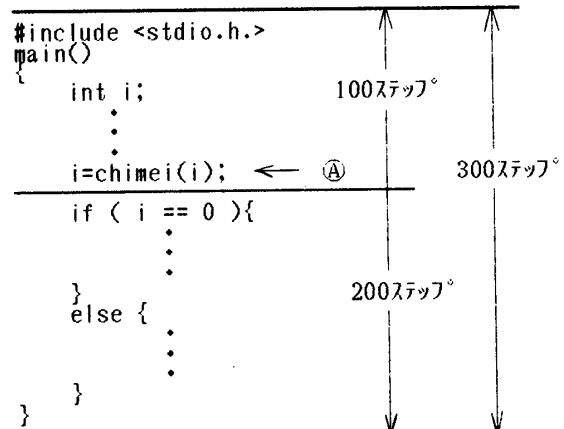


図2 テストの対象となるソースコードの例

5. まとめ

3のテスト項目の追加と4のソース・コードを考慮したときの取り扱いは、以下の値を考えることで統一的に扱うことができる。

$$\frac{\text{テスト項目} \times \text{テストの対象となっているステップ数}}{\text{プログラムの総ステップ数}}$$

この値が増せば、3の方法でより高精度のバグの推定が行なえよう。

6. おわりに

テスト項目の追加とテストの進行状況に応じたモデルの取り扱いについて吟味した。今後は、この妥当性に対して検討していく必要がある。

参考/引用文献

- [1] 大寺浩志, 山田茂, 成久洋之: “ソフトウェア信頼度成長モデルによるテスト工程管理”, 電子情報通信学会論文誌D Vol.J70-D No.5 pp.889-895.(1987.5).
- [2] 菅野文友: 「ソフトウェア・エンジニアリング」, 日科技連出版社, (1979). 以上.