

テスト網羅性とソフトウェア信頼性の関係に関する一考察

02005295 鳥取大学 *井上 真二 INOUE, Shinji
鳥取大学 山本 貴史 YAMAMOTO, Takafumi
01702425 鳥取大学 山田 茂 YAMADA, Shigeru

1 はじめに

ソフトウェア信頼度成長モデル (software reliability growth model, 以下 SRGM と略す) [3, 7] は, 現在, ソフトウェアシステムの定量的な信頼性評価手法の基盤技術の一つとして浸透している. 特に, 計数過程である非同次ポアソン過程 (nonhomogeneous Poisson process, 以下 NHPP と略す) に基づく SRGM は, 比較的単純なモデル構造と適用性の高さから, 多くの企業において実用に供されている SRGM の一つであり, その適用事例 [1, 4] も数多く報告されている.

NHPP モデルは, 現在までに様々な特徴をもつモデルが提案されている. その多くは, ソフトウェアの信頼度成長に影響を与える何らかの要因, もしくは関連する現象を反映したモデルである. 例えば, ソフトウェア開発の最終工程であるテスト工程において投入された労力 (テスト労力) を考慮したテスト労力依存型 SRGM [5] や, テスト空間の時間的拡大現象を考慮したテスト空間依存型 SRGM [6] が挙げられる.

本研究では, テスト工程におけるソフトウェア信頼度成長に影響を与える要因として, テスト網羅度 (testing-coverage) [2, 8] を取り上げ, 実現されたソフトウェア信頼性ととの関係と, その適合性について議論する. テスト網羅度には, 対象とするパスの種類によって様々な尺度が存在する. その中で, 実際のテスト工程において比較的容易に計測可能な命令網羅尺度 (C0 testing-coverage measure) を取り上げる. 命令網羅尺度とは, プログラムのすべての命令を通過することを基準としたものである. 具体的には, まず, 命令網羅尺度に関して, 代替的なテスト網羅度関数を構築する. その後, この代替的なテスト網羅度関数と観測されたフォールトデータとの関係を定式化し, 実際のテスト工程において観測されたテスト網羅度およびフォールト発見数データを用い, その適合性を検証する.

2 テスト網羅度関数の定式化

本研究では, 単位時間当りに上昇するテスト網羅度は, 最終的に達成可能なテスト網羅度と任意のテスト時刻におけるテスト網羅度の達成値との差に比例すると考える. すなわち, 次のような微分方程式により定式化されるものとする.

$$\frac{dC(t)}{dt} = \beta(t)[\alpha - C(t)] \quad (1)$$

$(0 \leq C(t) \leq 100, 0 < \alpha \leq 100, \beta(t) \geq 0).$

ここで, $C(t)$ (単位は百分率) は任意のテスト時刻 t におけるテスト網羅度, α (単位は百分率) はテスト開始前に設定される達成可能なテスト網羅度, $\beta(t)$ は任意のテスト時刻 t におけるテスト網羅度上昇率である. と, ところで, 式 (1) において, 任意のテスト時刻 t における

テスト網羅度は, 目標値を超えることはないという強い仮定が存在する. しかしながら, 特に命令網羅尺度の場合, 通常, 目標値は極めて高く設定されるため, このような定式化でも許容できるものとしている.

テスト網羅度は, 実行されたテストケースが効果的に設計されているかどうかの指標としても考えられ, テストケースの設計者の能力に大きく依存する. そこで, 本研究では, テストが進捗するにつれてテストケース設計者の能力が徐々に増加すると仮定する. したがって, 作成されるテストケースに関して, 初期フォールト検出能力 b_{ini} と定常フォールト検出能力 b_{sta} との比を,

$$r = \frac{b_{ini}}{b_{sta}}, \quad (2)$$

として, これをテストケース設計者のフォールト検出能力係数とすれば, 式 (1) の $\beta(t)$ は, 以下のように拡張される.

$$B(C(t)) = b_{sta} \left\{ r + (1-r) \frac{C(t)}{\alpha} \right\}. \quad (3)$$

式 (3) を式 (1) に代入し, 微分方程式を解くと,

$$C(t) = \frac{\alpha(1 - e^{-b_{sta}t})}{1 + z \cdot e^{-b_{sta}t}}, \quad (4)$$

となる. ここで, $z = (1-r)/r$ とした. 本研究では, これを命令網羅尺度に対する代替的なテスト網羅度関数と呼ぶことにする. 式 (4) において, $r = 1$ のとき代替的なテスト網羅度関数の形状は指数形曲線を示し, $r = 0$ のときロジスティック曲線を示す. ところで, 式 (4) は, 代表的な NHPP モデルの一つである習熟 S 字形 SRGM [7] の平均値関数と同一であるが, 本研究のモデルとはパラメータの解釈が異なることに注意する.

3 ソフトウェア信頼性との関係付け

本章では, テスト網羅度とフォールトデータの関連付けについて議論を行い, テスト網羅度を考慮した SRGM の構築の可能性を探る. まず, 任意のテスト時刻 t において発見されたフォールト数を表す計数過程 $\{N(t), t \geq 0\}$ に対して, NHPP を仮定する. NHPP モデルの平均値関数 $H(t)$ を構築するにあたり, 以下のような微分方程式を適用する.

$$\frac{dH(t)}{dt} = b(t)[a - H(t)]. \quad (5)$$

ここで, a はテスト開始前にソフトウェア内に潜在する総期待フォールト数, $b(t)$ は任意のテスト時刻 t における 1 個当りのフォールト発見率を示す. 式 (5) の微分方程式を解くと,

$$H(t) = a \left(1 - \exp \left[- \int_0^t b(x) dx \right] \right), \quad (6)$$

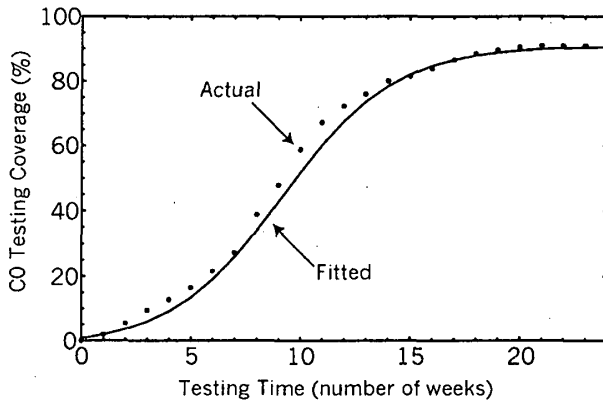


図 1: 推定されたテスト網羅度 (命令網羅尺度).

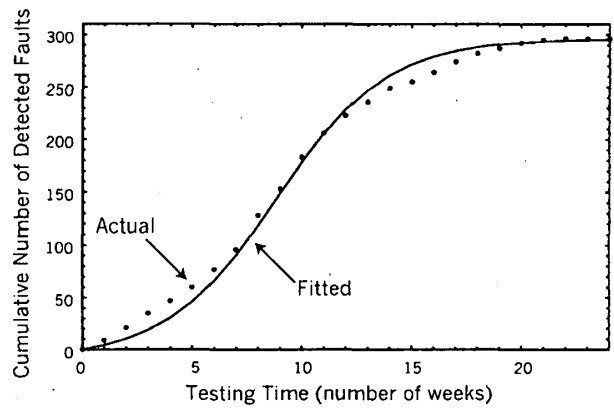


図 2: 推定された平均値関数 (命令網羅尺度).

を得る.

本研究では, 式 (5) において,

$$b(t) = b_s(t) = s \cdot c(t) \quad (0 < s < 1), \quad (7)$$

として, テスト網羅度とフォールトデータの関連付けを行うことにする. ここで, $c(t) \equiv dC(t)/dt$, s はテスト時刻 t において上昇した単位テスト網羅度当りのフォールト検出率である. 式 (6) および式 (7) より,

$$\left. \begin{aligned} H(t) &= a(1 - \exp[-s \cdot C(t)]) \\ C(t) &= \int_0^t c(x) dx \end{aligned} \right\}, \quad (8)$$

を得る. 本研究では, 式 (8) を平均値関数とする NHPP モデルを考える.

式 (8) において,

$$\lim_{t \rightarrow \infty} H(t) = a(1 - \exp[-s \cdot \alpha]), \quad (9)$$

となり, テスト開始前にソフトウェア内に潜在する総期待フォールト数 a には収束しない. つまり, $H(t)$ は, 任意のテスト時刻 t までに, 命令網羅尺度基準の下で発見された総期待フォールト数であることに起因する. すなわち, $a - H(\infty)$ は, 命令網羅尺度以外のテスト網羅基準により検出される総期待フォールト数を指すことになると考えられる.

4 数値例

本研究では, 実際に観測された命令網羅尺度およびフォールト発見数データを用い, 提案したモデルの数値例を与える. 適用する実測データは, 開発規模が 197.2×10^3 (LOC) であるソフトウェアに対するテスト工程において得られたデータである.

まず, 式 (4) に関して, 図 1 に, 推定されたテスト網羅度 (命令網羅尺度) を示す. ここで, 式 (4) における未知パラメータの推定には, 最小 2 乗法を適用し, パラメータ推定値 $\hat{\alpha} = 90.796$, $\hat{b}_{sta} = 0.388$, $\hat{z} = 52.338$ を得た. 次に, 式 (8) に関して, 図 2 に NHPP に基づいて推定された平均値関数を示す. なお, パラメータ推定値は, $\hat{a} = 450.489$, $\hat{s} = 0.018$ である. 図 1 および図 2 から, 推定されたモデルはそれぞれ, 実測データに対して適合していることが伺える.

5 おわりに

本研究では, テスト工程におけるソフトウェア信頼性に影響を与える要因として, テスト網羅度を上げ, その中の命令網羅尺度に関して, 観測されるフォールトデータとの関連付けを行った. しかしながら, 今回提案した NHPP モデルの平均値関数である式 (8) は, $t \rightarrow \infty$ のとき, 検出不可能なフォールト数 $a \cdot \exp[-s \cdot \alpha]$ が存在し, これを SRGM として取り扱うには様々な問題がある.

謝辞

本研究の一部は, 日本学術振興会科学研究費補助金基盤研究 (C)(2) (課題番号 15510129) の援助を受けたことを付記する.

参考文献

- [1] K. Konoun, M. Kaaniche, and J.C. Laprie, "Qualitative and quantitative reliability assessment," *IEEE Software*, vol. 14, no. 2, pp. 77-87, 1997.
- [2] Y.K. Malaiya, M.N. Li, J.M. Bieman, and R. Karcich, "Software reliability growth with test coverage," *IEEE Trans. Reliability*, vol. 51, no. 4, pp. 420-426, 2002.
- [3] J.D. Musa, D. Iannio, and K. Okumoto, *Software Reliability: Measurement, Prediction, Application*, McGraw-Hill, New York, 1987.
- [4] T. Nara, M. Nakata, and A. Ooishi, "Software reliability growth analysis — Application of NHPP models and its evaluation —," *Proc. 1996 High-Assurance Systems Engineering Workshop (HASE'96)*, pp. 222-227, 1996.
- [5] S. Yamada, H. Ohtera, and H. Narihisa, "Software reliability growth models with testing-effort," *IEEE Trans. Reliability*, vol. R-35, no. 1, pp. 19-23, 1986.
- [6] S. Yamada and T. Fujiwara, "Testing-domain dependent software reliability growth models and their comparisons of goodness-of-fit," *Intern. J. Reliability, Quality and Safety Engineering*, vol. 8, no. 3, pp. 205-218, 2001.
- [7] 山田 茂, ソフトウェア信頼性モデル — 基礎と応用 —, 日科技連出版社, 東京, 1994.
- [8] 山田 茂, 木村 光宏, 高橋 宗雄, TQM のための統計的品質管理 — 一般工業製品からソフトウェアまで —, コロナ社, 東京, 1998.