

2種類のソフトウェア故障発生原因を考慮した不完全デバッグモデル

01307475 鳥取大学 *得能貢一 TOKUNO Koichi
01702425 鳥取大学 山田茂 YAMADA Shigeru

1 まえがき

本論文では、Yamada et al. [1] の不完全デバッグモデルを拡張して、2種類のソフトウェア故障発生原因の存在を考慮したソフトウェア信頼度成長モデルについて議論する。一つは、修正が可能であり一度ソフトウェアから除去されると再発することのないフォールトであり、もう一つは、発見時に修正可能であっても再発したり、修正することによってソフトウェア内の他のモジュールに影響を及ぼして、当該フォールトに対する修正作業がソフトウェア故障率の低減に寄与しないようなフォールトである。前者および後者のフォールトに起因するソフトウェア故障の発生率は、それぞれ幾何減少型および一定のハザードレートで記述する。そこで、あるテスト時刻までに修正される累積フォールト数を表す確率変数を定義して、このモデルをマルコフ過程 (Markov process [2, 3]) によって定式化する。また、本モデルに基づいて、ソフトウェアの信頼性を評価するために有用な定量的尺度を導出する。最後に、ソフトウェアの信頼性評価の数値例を示す。

2 モデルの記述およびソフトウェアの信頼性評価尺度

本論文では、ソフトウェア故障を引き起こす原因であるフォールトには、次の2種類のタイプが存在するものと仮定する。

タイプ1フォールト：修正が可能であり一度ソフトウェアから除去されると再発することのないフォールト

タイプ2フォールト：フォールト修正作業を施してもソフトウェア故障率の低減に寄与しないようなフォールト

また、タイプ1フォールトおよびタイプ2フォールトに起因するソフトウェア故障を、それぞれF1およびF2と呼ぶことにする。

さらに、本論文で考慮する完全デバッグの定義を述べる。ソフトウェアに施されるデバッグ作業の目的は、潜在するフォールトを発見・修正・除去し、ソフトウェアの信頼性を向上させることにある。よって、デバッグ作業を行うことによってハザードレートを減少させることができた場合を完全デバッグであったとする。

ここで、本論文で構築されるソフトウェア信頼度成長モデルに対して、以下の仮定を設ける。

A1. ソフトウェア故障が発生するごとに、その原因となるフォールトを修正・除去するためのデバッグ作業が実施される。

A2. デバッグ作業は、確率 p ($0 \leq p \leq 1$) で成功し、確率 $q (= 1 - p)$ で失敗する。ここで、確率 p を完全デバッグ率と呼ぶことにする。

A3. F1に対するハザードレートは時間に関して一定であるが、フォールトが修正されるごとに幾何級数的に減少する。また、F2に対するハザードレートはフォールト修正の有無に関わらず一定とする [4]。

A4. 発生したソフトウェア故障は、F1とF2のいずれであるかは区別できない。

$\{X(t), t \geq 0\}$ をテスト時刻 t までに修正された累積フォールト数を表す計数過程とすると、 $\{X(t), t \geq 0\}$ は状態遷移確率が完全デバッグ率 p により支配されるマルコフ過程を形成する [2]。すなわち、仮定 A2 から、任意のテスト時刻 t において i 個のフォールトがすでに修正されているとき、新たにソフトウェア故障が発生すると、

$$X(t) = \begin{cases} i & (\text{確率 } q) \\ i+1 & (\text{確率 } p), \end{cases} \quad (1)$$

となる。

さらに、仮定 A3およびA4から、修正されたフォールト数が i 個のとき、新たにソフトウェア故障が発生する時間間隔 τ に対するハザードレートは、

$$z_i(\tau) = Dk^i + \theta \quad (D > 0, 0 < k < 1, \theta \geq 0, \tau \geq 0), \quad (2)$$

で与えられる。ここで、 D および k は、それぞれF1に対する初期ハザードレートおよびハザードレートの減少係数を表す。また、 θ はF2に対するハザードレートを表す。式 (2) は、テスト工程の初期段階ではソフトウェア故障の発生頻度は高く、完全なフォールト修正作業がソフトウェアの信頼性向上に大きく貢献するが、フォールト除去作業が進むにつれて、デバッグ作業が完全であってもソフトウェアの信頼性向上があまり見込めないようなソフトウェア故障発生現象を記述している [1, 5]。このとき、 $\{X(t), t \geq 0\}$ が状態 i から状態 j へ遷移する1ステップ推移確率は、

$$Q_{ij}(\tau) = P_{ij}(1 - e^{-(Dk^i + \theta)\tau}), \quad (3)$$

により与えられる。ただし、 P_{ij} は状態 i から状態 j への遷移確率を表し、

$$P_{ij} = \begin{cases} q & (j = i) \\ p & (j = i + 1) \\ 0 & (\text{その他}), \end{cases} \quad (4)$$

で与えられる。

上述したモデルに基づいて、ソフトウェアの信頼性評価尺度が導出できる。

n 個のフォールトが修正されるのに要する時間を表す確率変数 S_n に対する分布関数は、

$$\left. \begin{aligned} G_n(t) &= \sum_{i=0}^{n-1} A_i^n (1 - e^{-p(Dk^i + \theta)t}) \\ A_0^n &\equiv 1 \\ A_i^n &= \prod_{\substack{j=0 \\ j \neq i}}^{n-1} \frac{k^j + \theta/D}{k^j - k^i} \end{aligned} \right\} \quad (5)$$

で与えられ、平均および分散は、それぞれ

$$E[S_n] = \sum_{i=0}^{n-1} \frac{1}{p(Dk^i + \theta)} \quad (6)$$

$$\text{Var}[S_n] = \sum_{i=0}^{n-1} \frac{1}{p^2(Dk^i + \theta)^2} \quad (7)$$

となる。

また、任意のテスト時刻 t までに修正される累積フォールト数の期待値および分散は、式(5)を用いると、それぞれ

$$E[X(t)] = \sum_{n=1}^{\infty} G_n(t) \quad (8)$$

$$\text{Var}[X(t)] = \sum_{n=1}^{\infty} (2n-1)G_n(t) - \left\{ \sum_{n=1}^{\infty} G_n(t) \right\}^2 \quad (9)$$

と表される。

さらに、 $(l-1)$ 番目と l 番目のソフトウェア故障発生時間間隔を表す確率変数 X_l ($l = 1, 2, \dots$) に対する信頼度関数 (reliability function) は、

$$R_l(x) = \sum_{i=0}^{l-1} \binom{l-1}{i} p^i q^{l-1-i} e^{-(Dk^i + \theta)x} \quad (10)$$

となる。また、このときの平均ソフトウェア故障時間間隔 (mean time between software failures) は、

$$E[X_l] = \sum_{i=0}^{l-1} \binom{l-1}{i} \frac{p^i q^{l-1-i}}{Dk^i + \theta} \quad (11)$$

となる。

3 数値例と考察

式(5)の修正された累積フォールト数が n 個になるまでの時間 S_n に対する分布関数 $G_n(t)$ と θ の関係を図1に示す。ここで、 $n = 10$ とする。この図から、 θ が大きくなるほど、ソフトウェア故障の発生頻度が高くなり、それに伴ってデバッグ作業の実施回数も多くなることが予想されるので、所定のフォールト数を修正するのに要する時間は小さくなることが示される。

また、式(10)のソフトウェア信頼度 $R_l(x)$ とソフトウェア故障発生数 l の関係を図2に示す。この図より、ソフトウェア故障が発生するごとにソフトウェアの信頼性が向上していく様子が確認される。

参考文献

- [1] Yamada, S., Tokuno, K. and Osaki S., "Software reliability measurement in imperfect debugging environment and its application", *Reliability Engineering and System Safety*, vol. 40, no. 2, pp. 139-147 (1993).
- [2] Goel, A. L. and Okumoto, K., "A Markovian model for reliability and other performance measures of software system", *Proc. National Computer Conf.*, pp. 769-774 (1979).
- [3] Ross, S. M., "Stochastic Processes", John Wiley & Sons, New York (1983).
- [4] Moranda, P. B., "Event-altered rate models for general reliability analysis", *IEEE Trans. Reliab.*, vol. R-28, no. 5, pp. 376-381 (Dec. 1979).
- [5] 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).

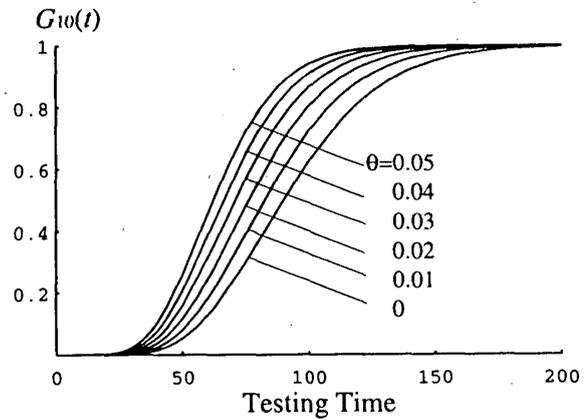


図1. θ と $G_n(t)$ の関係 ($n = 10, p = 0.9, D = 0.2, k = 0.9$).

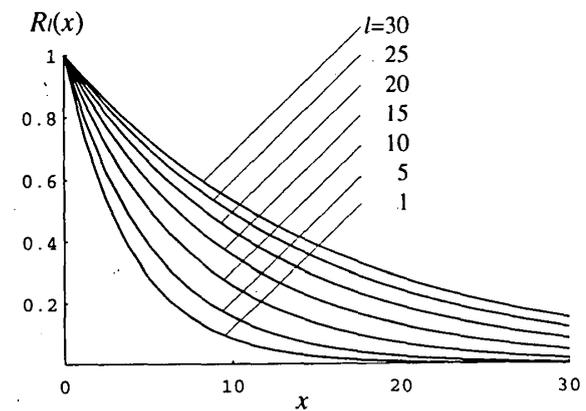


図2. ソフトウェア故障数 l と $R_l(x)$ の関係 ($\theta = 0.05, p = 0.9, D = 0.2, k = 0.9$).