

保証期間を考慮したソフトウェアの最適リリース問題に関する一考察

01702425 鳥取大学工学部 山田茂 YAMADA Shigeru

1. まえがき

ソフトウェアの開発プロセスの最終段階では、ソフトウェア内に潜入したフォールトを発見・除去することを目的としてテスト工程が設けられている。このとき、観測されるテストデータを用いて、ソフトウェアの品質・信頼性評価や出荷時期の見積り等も同時に実施される。このために、テスト工程におけるフォールト発見現象を、確率・統計論に基づく計数過程とみなしたソフトウェア信頼度成長モデル^[1~4]により記述する方法がよくとられる。

本稿では、テスト工程および運用段階における保守コストを分析し、これらをもとに導出される総期待ソフトウェアコストが最小となるように、ユーザにリリースする最適時期を決定する問題、いわゆるソフトウェアの最適リリース問題^[2,4~7]を議論する。ここで、ソフトウェアのリリース後、所定の期間に対しては、検出されたフォールトの処理コストを開発者が負担するような保証期間を設けるものとする^[8]。

2. ソフトウェア信頼度成長モデル

ソフトウェア開発の最終段階におけるテスト工程においては、フォールトの発見・修正が所定の手順に従って繰返し行われる。このようなフォールト発見現象を記述し、テスト工程におけるソフトウェアの信頼性評価を行うために、テスト時刻 t までに発見された総フォールト数を表す確率変数 $N(t)$ を導入し、これを計数過程とみなして $\{N(t), t \geq 0\}$ とする。この確率変数 $N(t)$ に対し、非同次ポアソン過程^[1~4] (nonhomogeneous Poisson process, 以下 NHPP と略す) を導入し、 $N(t)$ の期待値を $H(t)$ とすれば、ソフトウェア信頼度成長モデルは、

$$\Pr\{N(t) = n\} = \frac{H(t)^n}{n!} \exp[-H(t)], \quad (1)$$

$$H(t) = \int_0^t h(t) dt, \quad (2)$$

と定式化される。ここで $H(t)$ は、NHPP の平均値関数 (mean value function) と呼ばれ、テスト時刻 t までに発見される総期待フォールト数を表す。また $h(t)$ は、瞬間フォールト発見率を表し、NHPP の強度関数 (intensity function) と呼ばれる。

本稿では、単位テスト時間当りに発見されるフォールト数が、その時点において残存するフォールト数に比例するものと仮定して構築された指数形ソフトウェア信頼度成長モデル (exponential software reliability growth model)^[1~4] を考える。すなわち、

$$H(t) \equiv m(t) = a(1 - e^{-bt}) \quad (a > 0, b > 0), \quad (3)$$

$$h(t) \equiv h_m(t) \equiv \frac{dm(t)}{dt} = abe^{-bt}, \quad (4)$$

である。ここで、 a はテスト開始前にソフトウェア内に潜する総期待フォールト数、 b は任意のテスト時刻における1個当りのフォールト発見率を表す。このモデルは、ソフトウェア故障発生現象を記述したり、大規模ソフトウェアシステムの信頼性評価を行うときによく適用される。

3. ソフトウェア保守コストモデル

2. で述べた指数形ソフトウェア信頼度成長モデルに基づいて、テスト工程で発見されたフォールト数に関するデータ解析を行って信頼性評価を実施した後、運用段階に移行す

るのに最適な時刻を決定するという問題が開発管理上の一つの関心事となる。これは、適切な評価基準を導入し、実施テスト時間とテストにより確認されるソフトウェアの信頼性ととの関係を反映するように定式化され、ソフトウェアの最適リリース問題 (optimal software release problem)^[2,4~7] と呼ばれる。

本稿においては、評価基準として代表的なコスト要因^[9] を考慮した総期待ソフトウェアコストを採用し、これを最小化するような最適リリース時刻を決定する方策について議論する。ここで、開発者側がリリース後の保守コストを負担するような所定の保証期間^[7]を設定し、その長さがある確率分布に従う場合を考える。ソフトウェア保守コストモデルを定式化するために、以下の諸量を定義する。

c_0 : 最小限必要なテストコスト ($c_0 > 0$).

c_t : 単位時間当りのテストコスト ($c_t > 0$).

c_w : 保証期間中に発見・修正されるフォールト1個当りの保守コスト ($c_w > 0$).

T : ソフトウェアの総テスト時間、すなわちリリース時刻 ($T \geq 0$).

$C_w(T)$: 総テスト時間に関係したリリース後の保証期間におけるフォールト処理コスト。

$W(t)$: テスト終了時点から測定した保証期間の長さに対する分布関数 ($t \geq 0$).

テスト終了時点以降のソフトウェアの保証期間が分布関数 $W(t)$ に従うことから、ソフトウェアのテスト開始後より保証期間が終了するまでに要する総期待ソフトウェアコストは、

$$WC(T) \equiv c_0 + c_t T + C_w(T), \quad (5)$$

により与えられる。したがって、テスト工程および運用段階におけるフォールト発見現象が、式(3)の平均値関数 $m(t)$ をもつ NHPP モデルにより記述されるものと仮定すると、式(5)の $C_w(T)$ は次の3つの場合が考えられる (図1参照)。

[ケース1]: 保証期間は T_w (一定)、すなわち $W(t) = U(t - T_w)$ 、かつテスト終了時点で信頼度成長が停止する。ここで、 $U(t) = 1 (t \geq 0), U(t) = 0 (t < 0)$ とする。このとき、

$$C_w(T) = c_w h_m(T) T_w, \quad (6)$$

である。

[ケース2]: 保証期間は T_w (一定) かつテスト終了後も信頼度成長が継続する。このとき、

$$C_w(T) = c_w [m(T + T_w) - m(T)], \quad (7)$$

である。

[ケース3]: 保証期間は一般分布 $W(t)$ に従い、かつテスト終了後も信頼度成長が継続する。このとき、

$$C_w(T) = c_w \int_0^\infty [m(T + T_w) - m(T)] dW(T_w), \quad (8)$$

である。

4. 最適リリース方策

本稿では、保証期間の分布関数に切断正規分布

$$\frac{dW(t)}{dt} = \frac{1}{A\sqrt{2\pi}\sigma} \exp\left[-\frac{(t-\mu)^2}{2\sigma^2}\right] \quad (\mu > 0, \sigma > 0), \quad (9)$$

$$A = \frac{1}{\sqrt{2\pi}\sigma} \int_0^\infty e^{-\frac{(t-\mu)^2}{2\sigma^2}} dt, \quad (10)$$

を仮定する。なお、保証期間が一般分布 $W(t)$ に従い、かつテスト終了時点で信頼度成長が停止する場合は、[ケース 1] と同等となる。

3. の三つのケースについて、式(5)の総期待ソフトウェアコスト $WC(T)$ が最小となるような最適リリース時刻 $T = T^*$ を求めると、以下の最適方策を得る。

[最適方策 1] ([ケース 1])

(i) $h_m(0) > c_t/(c_w b T_w)$ ならば、最適リリース時刻は

$$T^* = \frac{1}{b} \ln\left[\frac{c_w a b^2 T_w}{c_t}\right], \quad (11)$$

である。

(ii) $h_m(0) \leq c_t/(c_w b T_w)$ ならば、最適リリース時刻は $T^* = 0$ である。

[方策 2] ([ケース 2])

(i) $h_m(0) > c_t/[c_w(1 - e^{-bT_w})]$ ならば、最適リリース時刻は

$$T^* = \frac{1}{b} \ln\left[\frac{c_w a b(1 - e^{-bT_w})}{c_t}\right], \quad (12)$$

である。

(ii) $h_m(0) \leq c_t/[c_w(1 - e^{-bT_w})]$ ならば、最適リリース時刻は $T^* = 0$ である。

[方策 3] ([ケース 3])

(i) $h_m(0) > c_t/(c_w \gamma)$ ならば、最適リリース時刻は

$$T^* = \frac{1}{b} \ln\left[\frac{c_w \gamma}{c_t}\right], \quad (13)$$

である。

(ii) $h_m(0) \leq c_t/(c_w \gamma)$ ならば、最適リリース時刻は $T^* = 0$ である。

ここで、

$$\gamma = 1 - \frac{e^{\frac{b^2\sigma^2}{2}} - b\mu\{1 - \Phi(-\frac{\mu}{\sigma} + b\sigma)\}}{1 - \Phi(-\frac{\mu}{\sigma})}, \quad (14)$$

$$\Phi(y) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^y e^{-\frac{u^2}{2}} du, \quad (15)$$

とする。

5. むすび

本稿では、ソフトウェアの開発管理上重要な問題として、適切なリリース時期を保守コストモデルに基づいて定量的に決定する最適リリース問題について考察を行った。ここで、NHPPに基づく指数形ソフトウェア信頼度モデルにより、テスト工程および運用段階におけるフォールト発見事象を記述した。特に、運用段階におけるソフトウェア保守のための費用を開発者が負担するような保証期間を考慮した。さらに、一般性をもたせるために、保証期間の長さがある確率分布に従うものとして保守コストモデルを記述した。

今後の課題として、本論文では指数形ソフトウェア信頼度成長モデルのみを取り扱ったが、対象とするソフトウェアによって信頼度成長の様相は異なるので、他のソフトウェア信頼度成長モデルを適用して最適リリース方策を議論する必要がある。

謝辞

本研究を進めるにあたりご協力頂いた、鳥取大学工学部の鶴田淳君に感謝致します。

参考文献

- [1] Musa, J. D., Iannino, A. and Okumoto, K. : *Software Reliability: Measurement, Prediction, Application*, McGraw-Hill, New York(1990).
- [2] 山田 茂: “ソフトウェア信頼性評価技術”, HBJ 出版局(1989).
- [3] 山田 茂, 高橋 宗雄: “ソフトウェアマネジメントモデル入門—ソフトウェア品質の可視化と評価法”, 共立出版(1993).
- [4] 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(1994).
- [5] Foreman, E. H. and Singpurwalla, N. D. : Optimal time intervals for testing-hypotheses on computer software errors, *IEEE Trans. Reliab.*, Vol. R-28, No. 3, pp. 250-253(August 1979).
- [6] Koch, H. S. and Kubat, P. : Optimal release time of computer software, *IEEE Trans. Software Eng.*, Vol. SE-9, No. 3, pp. 323-327(May 1983).
- [7] Okumoto, K. and Goel, A. L. : Optimum release time for software system based on reliability and cost criteria, *J. Syst. Software*, Vol. 1, No. 4, pp. 315-318(1980).
- [8] Shooman, M. L. : *Software Engineering: Design, Reliability, and Management*, McGraw-Hill, New York(1983).
- [9] Boehm, B. W. : *Software Engineering Economics*, Prentice-Hall, New Jersey(1981).
- [10] 山田 茂: “ソフトウェア保守コストモデルに基づく保証期間を考慮した最適リリース問題”, 情報処理学会論文誌, Vol.35, No.10, pp. 2197-2202(1994年10月).

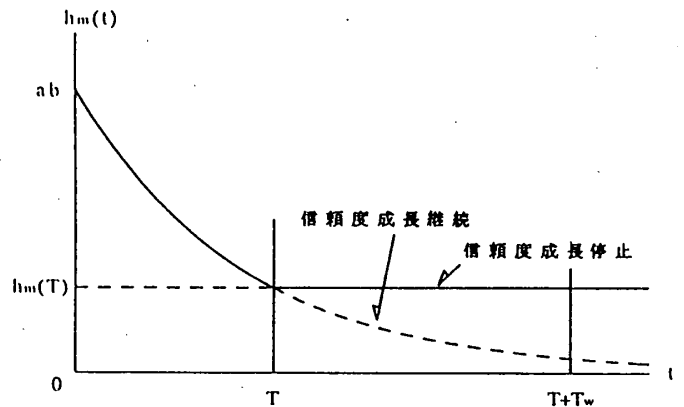


図1. 保証期間のソフトウェア信頼度成長過程