

分散開発環境に対する確率微分方程式モデルに基づく 最適リリース問題に関する一考察

02302815 鳥取環境大学 *田村 慶信† TAMURA Yoshinobu
02101865 法政大学 木村 光宏†† KIMURA Mitsuhiro
01702425 鳥取大学 山田 茂††† YAMADA Shigeru

E-mail: †tamura@kankyo-u.ac.jp, ††kim@k.hosei.ac.jp, †††yamada@sse.tottori-u.ac.jp

1 はじめに

現在のソフトウェア開発環境は、クライアント/サーバ・システムや Web プログラミング、オブジェクト指向開発、ネットワーク環境での分散開発といった新しい開発形態が多用されるようになってきている。

本研究では、こうした分散ソフトウェア開発環境を対象として、確率微分方程式モデルに基づくソフトウェアの最適リリース問題について議論する。ここで、テスト工程および運用段階で発生するコスト要因から導出される総ソフトウェアコストを最小にする総テスト時間を最適リリース時刻として求める。特に、総ソフトウェアコストの確率分布を考慮して、最適リリース時刻の信頼限界についても議論する。

2 分散開発環境に対する確率微分方程式モデル [1]

分散開発環境を対象としたソフトウェア信頼度成長モデル (software reliability growth model, 以下 SRGM と略す) として、以下に示すような分散開発環境の総合テスト工程におけるフォールト発見過程を、連続的に変動していく確率過程でモデル化した確率微分方程式 (stochastic differential equation, 以下 SDE と略す) に基づく SRGM が提案されている。以下では、この SRGM を SDE モデルと呼ぶ。

$$E[N_c(t)] = m_0 \left[1 - \left\{ \sum_{i=1}^n p_i e^{-b_i t} + \sum_{j=1}^m p_{n+j} (1 + b_{n+j} t) e^{-b_{n+j} t} \right\} e^{\frac{\sigma^2}{2} t} \right] \quad (1)$$

ここで、 $N_c(t)$ はテスト時刻 t までに発見・修正された総フォールト数を表し、式 (1) はその期待値である。また、 m_0 はテスト開始前に潜在していた総期待フォールト数を、 $b_l (l = 1, 2, \dots, n+m)$ は l 番目のコンポーネントに対する残存フォールト 1 個当りのフォールト発見率を、パラメータ $p_l (l = 1, 2, \dots, n+m)$ は l 番目のコンポーネントに対する重み、すなわちテストにおける重要度を表している。また、 σ は未知の定数パラメータである。本モデルは、既存のコンポーネントが n 個、新規開発のコンポーネントが m 個使われたものと仮定している。

3 最適リリース問題

3.1 ソフトウェアコストの定式化

総ソフトウェアコストを定式化するために、以下のコストパラメータを定義する。

- $c_{1,i}$: コンポーネント i の単体テストで発見されるフォールト 1 個当りの修正コスト ($c_{1,i} > 0$),
- $c_{2,i}$: コンポーネント i の単体テストでの単位時間当りのテストコスト ($c_{2,i} > 0$),
- c_{1c} : 総合テストで発見されるフォールト 1 個当りの修正コスト ($c_{1c} > 0$),
- c_{2c} : 総合テストでの単位時間当りのテストコスト ($c_{2c} > 0$),
- c_{3c} : 運用後に発見されるフォールト 1 個当りの保守コスト ($c_{3c} > 0, c_{3c} > c_{1i}, c_{3c} > c_{1c}$).

ここでは、各サブシステムについて累積発見フォールト数データの成長曲線の形状により、以下に示す非同次ポアソン過程 (nonhomogeneous Poisson process, 以下 NHPP と略す) に基づく指数形 SRGM と遅延 S 字形 SRGM を用いた信頼性評価法の適用を前提とする [2]:

- 指数形 SRGM

$$H_e(t) = a(1 - e^{-bt}) \quad (a > 0, b > 0), \quad (2)$$

- 遅延 S 字形 SRGM

$$H_d(t) = a \{ 1 - (1 + bt)e^{-bt} \} \quad (a > 0, b > 0). \quad (3)$$

ここで、 $H_e(t)$ および $H_d(t)$ は NHPP モデルにおける平均値関数であり、時間区間 $(0, t]$ において発見される期待累積フォールト数を表す。また、 a および b は、それぞれ最終的に発見される総期待フォールト数およびフォールト発見率を表す定数パラメータである。

さらに、各コンポーネントの総合テスト工程への遅延の影響による予定どおりに運用できないために発生したユーザに対するペナルティ (補償) などを考える必要性があり、単体テスト終了時刻が総合テスト開始時刻を越えた場合にはペナルティコストが課せられるものとする。コンポーネント $i (i = 1, 2, \dots, n, \dots, n+m)$ の単体テストから総合テスト工程への引き渡し遅延時間 $(t_i - t_{di})$ に対するペナルティコスト関数を以下のように定式化する。

$$G_i(t_i) = \begin{cases} c_{3i} \{ e^{k_i(t_i - t_{di})} - 1 \} & (t_i > t_{di}) \\ 0 & (t_i \leq t_{di}). \end{cases} \quad (4)$$

ここで、 t_{di} はコンポーネント i の単体テスト開始時刻から総合テスト開始時刻までの時間 ($t_{di} > 0$) を表し、 $c_{3i} (> 0)$ および $k_i (> 0)$ は定数パラメータを表す。よって、以下のような各コンポーネントのテストに要する期待コストの式が得られる。

$$C_i(t_i) = c_{1,i} H_i(t_i) + c_{2,i} t_i + G_i(t_i) \quad (i = 1, 2, \dots, n, \dots, n+m). \quad (5)$$

ここで、 $H_i(t_i)$ は i 番目のコンポーネントに対して適用された NHPP に基づく指数形 SRGM または遅延 S 字形 SRGM

の平均値関数を表す。さらに、ペナルティコスト関数において、コンポーネント i の単体テスト終了時刻が総合テスト開始時刻を越えない場合を、

$$t_{di} = t_i,$$

と仮定する。

これにより、総ソフトウェアコストを

$$\begin{aligned} Cost(N_c(t), t) = & \sum_{i=1}^{n+m} C_i(t_i) + c_{1c}t + c_{2c}N_c(t) \\ & + c_{3c}\{m_0 - N_c(t)\}, \end{aligned} \quad (6)$$

により定式化する。 $N_c(t)$ は確率変数であることから、 $Cost(N_c(t), t)$ は確率変数となる。

3.2 ソフトウェアコストの信頼区間と最適リリース時刻

本研究では、ソフトウェアコストの関数を確率変数として扱っていることから、コストの分布関数の値が $(1-0.01\alpha)/2$ となるコストと、 $(1+0.01\alpha)/2$ となるコストを求め、ソフトウェアコストの $\alpha\%$ 区間を求めることが可能となる。この区間を、ソフトウェアコストの $\alpha\%$ 信頼区間とする。このとき、ソフトウェアコストの $\alpha\%$ 信頼区間の上下限は、それぞれ

$$\begin{aligned} C_U(t) = & E[Cost(N_c(t), t)] \\ & + \beta_1(t)\sqrt{Var[Cost(N_c(t), t)]}, \end{aligned} \quad (7)$$

$$\begin{aligned} C_L(t) = & E[Cost(N_c(t), t)] \\ & - \beta_2(t)\sqrt{Var[Cost(N_c(t), t)]}, \end{aligned} \quad (8)$$

と与えられる。ここで、 $C_U(t)$ および $C_L(t)$ は、それぞれソフトウェアコストの $\alpha\%$ 信頼区間の上限および下限である。但し、 $\beta_1(t)$ 、 $\beta_2(t)$ は、それぞれ時刻 t における標準正規分布関数の $100(1 \pm \alpha)/2$ パーセント点である。

4 数値例

実際のテスト工程（総合テスト）において観測されたデータを適用した数値例を示す。

図1は推定されたソフトウェアコストのサンプルパスを表す。次に、推定された総期待ソフトウェアコストの時間変化の様子を図2に示す。図2から、最適リリース時刻は $T^* = 32.735$ と推定され、このときの総期待ソフトウェア

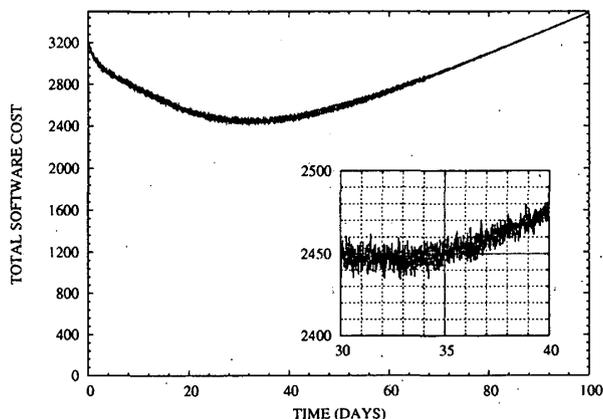


図1: 推定されたソフトウェアコストのサンプルパス。

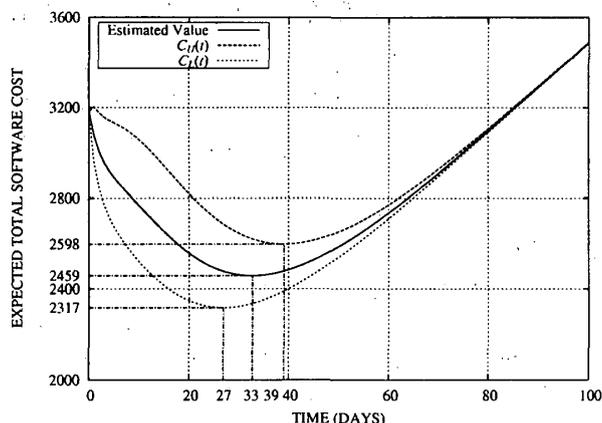


図2: 推定された総期待ソフトウェアコスト。

コストは 2458.5 となった。また、ソフトウェアコストの関数を確率変数として扱っているため、ソフトウェアコストの分布関数の値が 0.05 となるコストと、0.95 となるコストを求めることが可能となる。図2には、ソフトウェアコストの 90% 信頼区間の変化が示されており、 $C_U(t)$ および $C_L(t)$ においてソフトウェアコストが最小となる時刻はそれぞれ $T_U^* = 39.092$ および $T_L^* = 26.903$ となることが確認できる。開発管理者は、この情報をもとに、テスト工程を終了してユーザにソフトウェアを引き渡すのに最適な時期を定量的に把握することが可能となる。特に、統計的信頼区間に基づく存在範囲を用いることにより、より現実的な最適リリース時刻およびソフトウェアコストの見積りが可能となる。

5 おわりに

本研究では、総期待ソフトウェアコストを最小にする時刻を求めるだけでなく、最適リリース時刻を決定するための評価基準であるソフトウェアコストの関数を確率変数として扱うことにより、ソフトウェアコストの $\alpha\%$ 信頼区間を求めることができた。この $\alpha\%$ 信頼区間を用いることにより、最適リリース時刻および総期待ソフトウェアコストの存在範囲を確認し、ソフトウェア開発管理上において重要となるより現実的な最適リリース時刻を求めることが可能となる。すなわち、従来はソフトウェアコストの期待値を最小にする確定的な最適リリース時刻が求まるだけであった。しかし、本研究では、ソフトウェアフォールト数の分布が対数正規分布になることからソフトウェアコストの分布が分かり、ソフトウェアコストを最小にすることが $\alpha\%$ 確実となる最適リリース時刻を求めることが可能となった。

謝辞

本研究の一部は、文部科学省科学研究費基盤研究 (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] 田村慶信, 木村光宏, 山田茂, “分散開発環境に対するソフトウェア信頼度成長モデル: 確率微分方程式アプローチとその推定,” 日本応用数理学会論文誌, vol. 11, no. 3, pp. 121-132, 2001年9月.
- [2] 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.