エラー保全時間を考慮した最適ソフトウェア・リリース政策に関する一考察

土肥　　正[†]・富岡　恒雄・海生　直人

（受付　2005 年 4 月 18 日）

あ　ら　ま　し

本稿ではエラー保全時間が実際のサービス時間に依存し，ソフトウェア・リリース後の保証期間が既知である場合の総期待費用を最小にする最適ソフトウェア・リリース政策を，指数形ソフトウェア信頼度成長モデルを適用することにより議論する．

キー・ワード　ソフトウェア，リリース，総期待費用，最適政策，保全時間

1. ま　え　が　き

ソフトウェアをユーザに納入する時機を決定する問題は重要なテーマである．この問題は最適ソフトウェア・リリース問題と呼ばれ多くの著者によって議論されている[3−12]．これらのモデルではソフトウェア内に残存するエラーは発見と同時に瞬時に修正・除去されると仮定されている．しかしながら，エラーの修正・除去という保全活動には保全時間が無視できない場合もある．

本稿ではエラーの修正・除去時間が無視できない場合をモデル化し，最適なソフトウェア・リリース政策を導出する．すなわち，エラー保全時間が実際のサービス時間に依存し，ソフトウェア・リリース後の保証期間が既知である場合の総期待費用を最小にする最適ソフトウェア・リリース政策を議論する．第 3 節で最初に一般的な場合を議論し，次に期待エラー保全時間を線形関数とする場合を議論する．さらに第 4 節では指数形ソフトウェア信頼度成長モデルを適用することにより最適ソフトウェア・リリース政策を議論する．

[†] 広島大学大学院工学研究科情報工学専攻，東広島市

2. モデルと仮定

本論文で扱うモデルおよび仮定は以下のものである．

（1）　エラー保全（修正）時間が実際のサービス時間（ソフトウェア運用時間）に依存し，ソフトウェア・リリース後の保証期間が既知である場合の，総期待費用を最小にする最適ソフトウェア・リリース政策を議論する．

（2）　ソフトウェア内に残存するエラーによりソフトウェア障害（故障）が非定常ポアソン過程に従って発生する．

（3）　以下の諸量を導入する：

- $X(t)$　時刻 t までになされたエラー修正時間を含まない累積サービス時間．
- T_R　エラー修正時間を含まない総テスト時間．すなわち累積サービス時間 $X(t) = T_R$ でソフトウェアはリリースされる．
- T_G　ソフトウェア・リリース後のエラー修正時間を含まない保証期間．
- T_{LC}　$= T_R + T_G$，ソフトウェアのライフ・サイクル，すなわち累積サービス時間 $X(t) = T_{LC}$ でソフトウェアは廃棄される．
- $r(x)$　$X(t) = x$ でエラーが発見される率，すなわち $r(x)\Delta$ は $X(t) = x$ でエラーが発見されていないという条件の下で微小間隔 $(x, x + \Delta]$ 内でエラーが発見される確率の近似値を与える．非定常ポアソン過程の強度関数．
- $R(x)$　$= \int_0^x r(t)dt$．非定常ポアソン過程の平均値関数．
- V_x　$X(t) = x\ (0 \le x \le T_R)$ でエラーが発見されたときのエラー修正時間．
- W_{x-T_R}　$X(t) = x\ (T_R < x \le T_{LC})$ でエラーが発見されたときのエラー修正時間．
- $E[\cdot]$　期待値を示す演算子．
- k_R　$X(t) = x\ (0 \le x \le T_R)$ でのエラー発見に対する単位時間当りのエラー修正費用．
- k_G　$X(t) = x\ (T_R < x \le T_{LC})$ でのエラー発見に対する単位時間当りのエラー修正費用．
- k_o　エラー修正時間を含まない累積サービス時間におけるテスト期間 $[0, T_R]$ における単位時間当りのテスト費用．
- $C(T_R)$　ソフトウェアのライフ・サイクルおよびエラー修正期間における総期待費用．
- T_R^*　評価関数である総期待費用 $C(T_R)$ を最小にする最適なソフトウェア・リリース時刻（総テスト時間）T_R．

（4）　エラー修正時間を含まない累積サービス時間において製品であるソフトウェア内に存在するエラーは $X(t) = x$ におけるソフトウェア障害の発生を経由することにより，テスト期間 $[0, T_R]$ においてはエラー修正時間 V_x を伴い，また保証期間 $(T_R, T_{LC}]$ においてはエラー

修正時間 W_{x-T_R} を伴って確率 1 で除去される．

（5） 各エラーはそれぞれ独立であるとする．すなわち各エラー間に依存性はないとする．

（6） エラー修正時間を含まない累積サービス時間での時刻 $X(t)=T_R$ でソフトウェア・テストは終了し，ソフトウェアがユーザにリリース（納入）される（この場合，全てのエラーが除去されているとは限らない）．

このモデルおよび仮定のもとでソフトウェアのライフ・サイクルおよびエラー修正期間における総期待費用を最小にする最適ソフトウェア・リリース政策を以下議論する．

3. 解 析

前節で示した仮定のもとでのソフトウェアのライフ・サイクルおよびエラー修正期間における総期待費用 $C(T_R)$ は，

$$C(T_R)=k_oT_R+k_R\int_0^{T_R}E[V_x]\,r(x)dx+k_G\int_{T_R}^{T_R+T_G}E[W_{x-T_R}]\,r(x)dx,\ 0\le T_R<\infty \tag{3.1}$$

となる．また，$C(\infty)\to\infty$ および

$$C(0)=k_G\int_0^{T_G}E[W_x]\,r(x)dx \tag{3.2}$$

となる．

総期待費用 $C(T_R)$ の導関数を $q(T_R)$ とすると，

$$\begin{aligned}q(T_R)&=k_o+k_R\ E[V_{T_R}]\,r(T_R)\\&\quad+k_G\left\{E[W_{T_G}]\,r(T_R+T_G)-E[W_0]\,r(T_R)+\int_{T_R}^{T_R+T_G}\frac{\partial E[W_{x-T_R}]}{\partial T_R}r(x)dx\right\}\end{aligned} \tag{3.3}$$

となる．さらに，

$$\begin{aligned}q'(T_R)&=k_R\left\{E'[V_{T_R}]\,r(T_R)+E[V_{T_R}]\,r'(T_R)\right\}\\&\quad+k_G\left\{E[W_{T_G}]\,r'(T_R+T_G)-E[W_0]\,r'(T_R)\right.\\&\quad+\left.\frac{\partial E[W_{x-T_R}]}{\partial T_R}\right|_{x=T_R+T_G}\cdot r(T_R+T_G)-\left.\frac{\partial E[W_{x-T_R}]}{\partial T_R}\right|_{x=T_R}\cdot r(T_R)\\&\quad\left.+\int_{T_R}^{T_R+T_G}\frac{\partial^2E[W_{x-T_R}]}{\partial T_R^2}r(x)dx\right\}\end{aligned} \tag{3.4}$$

となる．

ここで以下の様に期待エラー修正時間を線形関数

$$E[V_x]=\alpha_1 x+\beta_1,\ \alpha_1\ge 0,\ \beta_1\ge 0,\ 0\le x\le T_R, \tag{3.5}$$

および

$$E[W_{x-T_R}]=\alpha_2(x-T_R)+\beta_2,\ \alpha_2\ge 0,\ \beta_2\ge 0, T_R<x\le T_{LC} \tag{3.6}$$

と定義する．但し，パラメータ α_i および β_i $(i = 1, 2)$ は特にことわらない限り $\alpha_i > 0$，$\beta_i > 0$ とする．パラメータ β_i $(i = 1, 2)$ はそれぞれテスト段階および運用（保証）段階におけるエラー修正・除去に対して必要とされる固定時間を表す．また，期待時間 $\alpha_1 x$ および $\alpha_2(x-T_R)$ は $X(t) = x$ における期待保全時間の変動部分（時間）である．このとき，総期待費用 $C(T_R)$ および関連諸関数は式(3.1)－(3.4)より，

$$\begin{aligned}C(T_R)&=k_o T_R+k_R\int_0^{T_R}(\alpha_1 x+\beta_1)\,r(x)dx\\&\quad+k_G\int_{T_R}^{T_R+T_G}\left\{\alpha_2(x-T_R)+\beta_2\right\}r(x)dx,\ 0\le T_R<\infty,\end{aligned} \tag{3.7}$$

$$C(0)=k_G\int_0^{T_G}(\alpha_2 x+\beta_2)\,r(x)dx, \tag{3.8}$$

$$\begin{aligned}q(T_R)&=k_o+k_R(\alpha_1 T_R+\beta_1)\,r(T_R)\\&\quad+k_G\left\{(\alpha_2 T_G+\beta_2)\,r(T_R+T_G)-\beta_2 r(T_R)-\alpha_2\int_{T_R}^{T_R+T_G}r(x)dx\right\},\end{aligned} \tag{3.9}$$

$$q(0)=k_o+k_R\beta_1 r(0)+k_G\left\{(\alpha_2 T_G+\beta_2)r(T_G)-\beta_2 r(0)-\alpha_2\int_0^{T_G}r(x)dx\right\} \tag{3.10}$$

および $q(\infty)>0$ となり，さらに，

$$\begin{aligned}q'(T_R)&=k_R\left\{\alpha_1 r(T_R)+(\alpha_1 T_R+\beta_1)\,r'(T_R)\right\}\\&\quad+k_G\left[(\alpha_2 T_G+\beta_2)\,r'(T_R+T_G)-\beta_2 r'(T_R)-\alpha_2\left\{r(T_R+T_G)-r(T_R)\right\}\right]\end{aligned} \tag{3.11}$$

となる．以下の議論においては総期待費用 $C(T_R)$ のソフトウェア・リリース時刻（総テスト時間）T_R に関する最小化問題を取扱うが，必要条件 $q(T_R) = 0$ を満足する T_R を式(3.7)の $C(T_R)$ に代入すると，

$$\begin{aligned}C(T_R)&=k_R\left\{\int_0^{T_R}(\alpha_1 x+\beta_1)\,r(x)dx-(\alpha_1 T_R+\beta_1)\,T_R r(T_R)\right\}\\&\quad+k_G\left\{\int_{T_R}^{T_R+T_G}(\alpha_2 x+\beta_2)\,r(x)dx-(\alpha_2 T_G+\beta_2)T_R r(T_R+T_G)\right.\\&\quad\left.+\beta_2 T_R r(T_R)\right\}\end{aligned} \tag{3.12}$$

となる．

次にテスト段階および運用（保証）段階におけるソフトウェアの障害発生過程，すなわちエラー発見過程を，非定常ポアソン過程に基づいた典型的なソフトウェア信頼度成長モデル，すなわち指数形ソフトウェア信頼度成長モデルを導入することにより記述し，総期待費用を最小にする最適ソフトウェア・リリース政策を導出する．

4. 指数形ソフトウェア信頼度成長モデルを適用した場合の最適ソフトウェア・リリース政策

Goel and Okumoto[1)] は単位時間当りに発見される期待エラー数はソフトウェア内に残存する期待エラー数に比例すると仮定し以下の指数形ソフトウェア信頼度成長モデル

$$\frac{dR(x)}{dx}=b\{a-R(x)\},\ a>0,\ b>0,\ X(t)=x \tag{4.1}$$

を提案した．ここで a および b は以下のものである．

a　最終的に発見される期待エラー数.

b　残存エラー 1 単位当りのエラー発見率.

式(4.1)における微分方程式を初期条件 $R(0)=0$ の下で解くと,

$$R(x)=a(1-e^{-bx}) \tag{4.2}$$

となり，さらに,

$$r(x)=abe^{-bx} \tag{4.3}$$

を与える．すなわち指数形ソフトウェア信頼度成長モデルの平均値関数が式(4.2)で，強度関数が式(4.3)で与えられる．また,

$$R(\infty)=a \tag{4.4}$$

となり，最終的に発見される期待エラー数を与える．

このとき，総期待費用 $C(T_R)$ および関連諸関数は以下のように与えられる．

$$\begin{aligned}C(T_R)&=k_oT_R+k_R\int_0^{T_R}(\alpha_1x+\beta_1)abe^{-bx}dx\\&\quad+k_G\int_{T_R}^{T_R+T_G}\{\alpha_2(x-T_R)+\beta_2\}abe^{-bx}dx\\&=k_oT_R\\&\quad+k_Ra\left\{-\alpha_1T_Re^{-bT_R}+\left(\frac{\alpha_1}{b}+\beta_1\right)\left(1-e^{-bT_R}\right)\right\}\\&\quad+k_Gae^{-bT_R}\left\{-\alpha_2T_Ge^{-bT_G}+\left(\beta_2+\frac{\alpha_2}{b}\right)(1-e^{-bT_G})\right\},\\&\quad 0\le T_R<\infty.\end{aligned} \tag{4.5}$$

$$C(0)=k_G a\left\{-\alpha_2 T_G e^{-bT_G}+\left(\beta_2+\frac{\alpha_2}{b}\right)(1-e^{-bT_G})\right\}. \tag{4.6}$$

$$\begin{aligned}q(T_R)&=k_o+k_R(\alpha_1 T_R+\beta_1)abe^{-bT_R}\\&\quad+k_G\left\{(\alpha_2 T_G+\beta_2)abe^{-b(T_R+T_G)}-\beta_2 abe^{-bT_R}-\alpha_2 a(-e^{-b(T_R+T_G)}+e^{-bT_R})\right\}\\&=k_o+abe^{-bT_R}\left[k_R(\alpha_1 T_R+\beta_1)+k_G\left\{\alpha_2 T_G e^{-bT_G}-\left(\beta_2+\frac{\alpha_2}{b}\right)(1-e^{-bT_G})\right\}\right].\end{aligned} \tag{4.7}$$

$$q(0)=k_o+ab\left[(k_R\beta_1-k_G\beta_2)+k_G\left\{(\alpha_2 T_G+\beta_2)e^{-bT_G}-\frac{\alpha_2}{b}(1-e^{-bT_G})\right\}\right]. \tag{4.8}$$

$$\begin{aligned}q'(T_R)&=k_R\left\{\alpha_1 abe^{-bT_R}-(\alpha_1 T_R+\beta_1)ab^2 e^{-bT_R}\right\}\\&\quad+k_G\Big[-(\alpha_2 T_G+\beta_2)ab^2 e^{-b(T_R+T_G)}+\beta_2 ab^2 e^{-bT_R}\\&\quad-\alpha_2 ab\left\{e^{-b(T_R+T_G)}-e^{-bT_R}\right\}\Big]\\&=abe^{-bT_R}\Big[k_R\alpha_1-b(k_R\beta_1-k_G\beta_2)+k_G\left\{-(\alpha_2 T_G+\beta_2)be^{-bT_G}\right.\\&\quad\left.+\alpha_2(1-e^{-bT_G})\right\}-k_R\alpha_1 bT_R\Big].\end{aligned} \tag{4.9}$$

必要条件 $q(T_R^*)=0$ を満足する T_R^* を式(4.5)に代入すると $C(T_R^*)$ は以下のように求められる.

$$\begin{aligned}C(T_R^*)&=k_R\Big[\alpha_1 a\left\{-\left(T_R^*+\frac{1}{b}\right)e^{-bT_R^*}+\frac{1}{b}\right\}+\beta_1 a(1-e^{-bT_R^*})\\&\quad-(\alpha_1 T_R^*+\beta_1)T_R^* abe^{-bT_R^*}\Big]\\&\quad+k_G\Big[\alpha_2 a\left\{-\left(T_R^*+T_G+\frac{1}{b}\right)e^{-b(T_R^*+T_G)}+\left(T_R^*+\frac{1}{b}\right)e^{-bT_R^*}\right\}\\&\quad+\beta_2 a(-e^{-b(T_R^*+T_G)}+e^{-bT_R^*})-(\alpha_2 T_G+\beta_2)T_R^* abe^{-b(T_R^*+T_G)}\\&\quad+\beta_2 T_R^* abe^{-bT_R^*}\Big]\\&=a\Big[k_R\left[-\left\{\alpha_1+(\alpha_1 T_R^*+\beta_1)b\right\}T_R^* e^{-bT_R^*}+\left(\frac{\alpha_1}{b}+\beta_1\right)(1-e^{-bT_R^*})\right]\\&\quad+k_G e^{-bT_R^*}\Big[\left\{(\alpha_2+\beta_2 b)T_R^*+\frac{\alpha_2}{b}+\beta_2 a\right\}(1-e^{-bT_G})\\&\quad-\alpha_2 T_G(1+bT_R^*)e^{-bT_G}\Big]\Big].\end{aligned} \tag{4.10}$$

以上の諸関数を用いて解析を行うと結果的に以下の最適ソフトウェア・リリース政策を求めることができる（証明略）.

最適ソフトウェア・リリース政策

式(4.5)における総期待費用 $C(T_R)$ を最小にする T_R を最適ソフトウェア・リリース時刻 T_R^* とする．そのとき以下が成立する．

（1） $k_R\alpha_1 + b(k_G\beta_2 - k_R\beta_1) + k_G\{-(\alpha_2 T_G + \beta_2)be^{-bT_G} + \alpha_2(1-e^{-bT_G})\} > 0$ であるとき次のことが成立つ．

（i） もし式(4.8)における $q(0) < 0$ ならば式(4.7)における $q(T_R) = 0$ を満足する有限でただ1つの $T_R^*(0 < T_R^* < \infty)$ が存在し，そのときの総期待費用 $C(T_R^*)$ は式(4.10)で与えられる．

（ii） もし $q(0) \geq 0$ ならば $T_R^* = 0$ となる．すなわちテストは実施されない．そのときの総期待費用は式(4.6)で与えられる．

（2） $k_R\alpha_1 + b(k_G\beta_2 - k_R\beta_1) + k_G\{-(\alpha_2 T_G + \beta_2)be^{-bT_G} + \alpha_2(1-e^{-bT_G})\} \leq 0$ であるときには，$T_R^* = 0$ となる．||

本節では指数形ソフトウェア信頼度成長モデルを適用することにより最適ソフトウェア・リリース政策を求めた．

5. む　す　び

本稿ではエラー保全（修正）時間が実際のサービス時間（ソフトウェア運用時間）に依存し，ソフトウェア・リリース後の保証期間が既知である場合の，総期待費用を最小にする最適ソフトウェア・リリース政策を議論した．最初に一般的な場合を取扱い，次に期待エラー修正時間を線形関数とし，最後に指数形ソフトウェア信頼度成長モデルを適用することにより最適ソフトウェア・リリース政策を求めた．

最後に線形エラー修正時間の保証段階における変動部分（時間）で適用される比例定数 α_2 について言及する．保証（運用）段階においてユーザは到着率 $\lambda\ (>0)$ を伴うポアソン過程に従ってランダムにソフトウェアを使用する．その使用時間は平均 $1/\mu\ (\mu>0)$ を伴う指数分布に従い，各使用時間は独立とする．すなわち M/M/1 型の待ち行列モデルを仮定する．このとき，トランザクションの処理時間を基本としたエラー修正時間の変動部分に対する比例定数を $k\ (>0)$ とすると α_2 は

$$\alpha_2 = k\frac{\lambda}{\mu}\quad\left(\frac{\lambda}{\mu}\leq 1\right) \tag{5.1}$$

で与えられる[2)]．ここで λ/μ はよく知られたトラフィック密度である．

文　献

1) A. L. Goel and K. Okumoto, "Time-dependent error-detection rate model for software reliability and other performance measures," IEEE Trans. Reliability, vol. R-28, pp. 206–211, 1979.
2) E. Gelenbe, "On the optimum checkpoint interval," J. ACM, vol. 26, pp. 259–270, 1979.
3) P. K. Kapur and R. G. Garg, "Optimal software release policies for software reliability growth models under imperfect debugging, "RAIRO Recherche operationnelle/Operations Research, vol. 24, pp. 295–305, 1990.
4) P. K. Kapur and R. G. Garg, "Cost-reliability optimum release policies for a software system with testing effort," Opsearch, vol. 27, pp. 109–116, 1990.
5) P. K. Kapur and R. G. Garg, "Optimal release policies for software systems with testing effort," International J. Systems Science, vol. 22, pp. 1563–1571, 1991.
6) H. S. Koch and P. Kubat, "Optimal release time for computer software," IEEE Trans. Software Engineering, vol. SE-9, pp. 323–327, 1983.
7) K. Okumoto and A. L. Geol, "Optimum release time for software systems based on reliability and cost criteria," J. Systems and Software, vol.1, pp. 315–318, 1980.
8) H. Ohtera and S. Yamada, "Optimum software release time considering an error detection phenomenon during operation," IEEE Trans. Reliability, vol.R-39, pp. 596–599, 1990.
9) Y. Shinohara, T. Dohi and S. Osaki, "Comparisons of optimal release policies for software systems," in Proceedings of 20th International Conference on Computers & Industrial Engineering, pp. 493–496, 1996.
10) S. Yamada and S. Osaki, "Cost-reliability optimal release policies for software systems," IEEE Trans. Reliability, vol.R-34, pp. 422–424, 1985.
11) S. Yamada and S. Osaki, "Optimal software release policies with simultaneous cost and reliability requirements," European J. Operational Research, vol. 31, pp. 46–51, 1987.
12) W. Y. Yun and D. S. Bai, "Optimum software release policy with random life cycle," IEEE Trans. Reliability, vol. R-39, pp. 167–170, 1990.

Abstract

A Note on Optimal Software Release Policies Taking Account of Error Maintenance Time

Tadashi Dohi*, Tsuneo Tomioka and Naoto Kaio

In this paper, we discuss optimal software release policies minimizing the total expected cost. We treat the case that each error maintenance time depends on each effective service time, the operating (guarantee) time after the software release is known and the exponential software reliability growth model is applied.

* Department of Information Engineering, Hiroshima University, Higashi-Hiroshima 739-8527, Japan