ソフトウェア最適リリース問題のノンパラメトリック解法

土肥正，篠原康秀，西尾泰彦，尾崎俊治
広島大学工学部

1　はじめに

ソフトウェアのフォールト発見個数が非同次ポアソン過程(NHPP)に従う場合のソフトウェア信頼度成長モデルに対して，総期待ソフトウェア費用を最小にする目的で，様々な形態のソフトウェア最適リリース問題が議論されている [1]. 特に，通常のソフトウェア信頼度成長モデルはパラメトリックモデルであるため，その平均値関数の具体的な形状を事前に仮定しなければならない．これに対して本稿では，ソフトウェアのフォールト発見時間データを用いて，ソフトウェアのテスト工程から運用段階に移行するための最適停止時期を決定するノンパラメトリックな手法を提案する．

具体的には，ソフトウェア最適リリース問題を二次元平面上での幾何学的問題に変換し，フォールト発見時間データから総期待ソフトウェア費用を最小にするソフトウェアの最適リリースタイミングを推定する．このような幾何学的解法の特徴として，(i) 実務家のソフトウェア最適リリース問題への理解を助ける，(ii) フォールト発見時刻データに忠実な意思決定を可能とする等が挙げられる．

2　NHPPに基づくソフトウェア信頼度成長モデル

計数過程 $\{N(t), t \geq 0\}$ を時刻 t までに発見される累積ソフトウェアフォールト数とする．$N(t)$ が以下の条件：

(i)　$N(0)=0$

(ii) $\{N(t), t \geq 0\}$ は独立増分をもつ

(iii) $\mathrm{P_r}\{N(t+h)-N(t)=1\}=\lambda(t)h+o(h)$

(iv) $\mathrm{P_r}\{N(t+h)-N(t)\geq 2\}=o(h)$

を満たすならば，ソフトウェアの累積フォールト数は確率関数

$$\mathrm{P_r}\{N(t)=n|N(0)=n\}=\frac{\{\Lambda(t)\}^n}{n!}e^{-\Lambda(t)} \tag{1}$$

を持つ非同次ポアソン過程(NHPP)に従うという．ここで，

$\lambda(t)$：時刻 t における瞬間ソフトウェア故障率，

$\Lambda(t)$：時刻 t における総期待ソフトウェアフォールト数，

であり，特に

$$\Lambda(t)=\int_0^t \lambda(x)\,dx \tag{2}$$

はNHPPの平均値関数と呼ばれる．

NHPPに基づく代表的なソフトウェア信頼度成長モデルとして，Goel and Okumoto [2] は，テスト工程において単位時間当たりに発見されるソフトウェアフォールト数がソフトウェア内に残存するフォールトの数に比例するという仮定の下で，指数型信頼度成長モデルを提案している．一方，Yamada and Osaki [3] は，複雑なソフトウェアにおけるフォールト発見過程とその原因解析を行うフォールト認知過程を導入することによって，遅延S字型信頼度成長モデルを提案している．

3　ソフトウェア最適リリース問題

ソフトウェア製品をテスト段階から運用段階に移行させるために，経済性（および信頼性）の観点からソフトウェアを出荷する時期を理論的に決定することは極めて重要な管理技術となる．通常，このような問題はソフトウェア最適リリース問題と呼ばれ，様々な角度から解析が行われている（例えば [4] を参照）．

いま，T $(0 \leq T \leq T_{LC})$ をソフトウェアリリース時刻とし，以下のような費用構造を仮定する．

$c_1(>0)$：テスト工程において発見されるフォールト一個当たりの修正費用

$c_2(>0)$：運用段階において発見されるフォールト一個当たりの修正費用

$c_3(>0)$：ソフトウェアテストに要する単位時間当たりの費用

ここで，T_{LC} (>0) はソフトウェア寿命であり，既知の定数とする．このとき，NHPPに基づいたソフトウェア信頼度成長モデルに対して，総期待ソフトウェア費用は以下のように定式化される．

$$C(T)=c_1\Lambda(T)+c_2\{\Lambda(T_{LC})-\Lambda(T)\}+c_3T. \tag{3}$$

これより，問題は総期待ソフトウェア費用を最小にする最適ソフトウェアリリース時刻 T^* を求めることである．従って，

$$\min_{0\leq T\leq T_{LC}} C(T). \tag{4}$$

通常，上述のような意志決定を実際のテスト工程で行う場合，ソフトウェアの累積フォールト発見時間データ

© 日本オペレーションズ・リサーチ学会．無断複写・複製・転載を禁ず．

$$0 = x_0 < x_1 < \cdots < x_n, \quad (n: \text{データ数}), \tag{5}$$

が採集された後に，仮定された平均値関数 $\Lambda(T)$ に基づいて最尤推定を行い，$\Lambda(T)$ のパラメトリック推定量 $\hat{\Lambda}(T)$ を求める．続いて，式 (3) に $\hat{\Lambda}(T)$ を代入することによって，式 (4) で与えられた最小化問題を解く．

4 幾何学的問題としての解釈

ここでは，ソフトウェア最適リリース問題に対する幾何学的解法について述べる．

補題 1: 式 (4) によって与えられる最小化問題は

$$\max_{0 \le T \le T_{LC}} \left\{ \Lambda(T) - \frac{c_3}{c_2 - c_1} T \right\} \tag{6}$$

のような双対問題と等価である．

これより，平均値関数が既知の場合，(x, y) 座標平面上に $(T, \Lambda(T))$ をプロットし，曲線 $y = \Lambda(T)$ と直線 $y = \{c_3/(c_2 - c_1)\}T$ の距離を最大にする T^* を求めればよい．

以下の結果は，指数型信頼度成長モデルのように，単調非減少な平均値関数が有界でかつ変極点をもたない場合に成立する．

命題 1: (i) 原点における曲線 $y = \Lambda(T)$ の傾きが $c_3/(c_2 - c_1)$ より大きく，かつ曲線 $y = \Lambda(T)$ と直線 $y = \{c_3/(c_2 - c_1)\}T$ の交点の x 座標が T_{LC} より小さいならば，有限で唯一のソフトウェアリリース時刻 $T^*(0 < T^* < T_{LC})$ が存在する．

(ii) (i) と同様な条件の下で，曲線 $y = \Lambda(T)$ と直線 $y = \{c_3/(c_2 - c_1)\}T$ の交点が T_{LC} よりも大きいならば，最適解は $0 < T^* \le T_{LC}$ の範囲で存在する．

(iii) 原点における曲線 $y = \Lambda(T)$ の傾きが $c_3/(c_2 - c_1)$ より小さい場合，最適ソフトウェアリリース時刻は $T^* = 0$ となり，ソフトウェアテストを全く行わないことが経済的に最適となる．

5 ノンパラメトリック解法

いま，式 (5) のように n 個の累積フォールト発見時間データが得られているものとし，ソフトウェアに含まれる初期フォールト数 N(定数) は十分大きく，テスト工程で観測される他の要因から推定可能であるとする．時刻 x_n において，将来の $i\ (i = n+1, \cdots, N)$ 番目のフォールト発見時刻の推定値 $\tilde{x}_i$ が得られているものとすると，N 個の標本に基づく平均値関数のノンパラメトリック推定量は

$$\tilde{\Lambda}_N(x) = -\log \overline{F_N}(x) \tag{7}$$

によって与えられる．ここで，$\overline{F_N}(\cdot) = 1 - F_N(\cdot)$ であり，

$$F_N(x) = \begin{cases} 0 & (0 \le x \le x_1) \\ j/N & (x_j < x \le x_{j+1}, j = 1, \cdots, N-1) \\ 1 & (x \ge x_N) \end{cases} \tag{8}$$

はデータ x と推定値 $\tilde{x}$ に対する経験分布を表わす．

命題 2: (i) データ $x_k - x_{k-1}\ (k = 1, \cdots, n)$ および推定値 $\tilde{x}_i - \tilde{x}_{i-1}\ (i = n+1, \cdots, N)$ 間における平均値関数の変化量の推定値は

$$\tilde{d}_j = \log\left(1 + \frac{1}{N - j}\right), \quad (j = 1, \cdots, N) \tag{9}$$

となり，初期フォールト数の減少列かつフォールト発見個数の増加列となる．

(ii) 十分大きい N に対して，ソフトウェアの最適リリース時期を求める問題は

$$\max\{\tilde{\Lambda}(x_n) - c_3 \tilde{x}_n/(c_2 - c_1), \tilde{V}_n\} \tag{10}$$

となる．ここで，

$$\tilde{V}_n = \max_{n+1 \le i < N} \left\{ \tilde{\Lambda}_N(\tilde{x}_i) - \frac{c_3}{c_2 - c_1} \tilde{x}_i \right\} \tag{11}$$

であり，最適解の推定値はフォールト発見時刻のデータ x_n もしくは $\tilde{x}_i\ (i = n+1, \cdots, N-1)$ の中に必ず存在する．

注意: 命題 2 (i) より，もし $i = N$ ならば $\tilde{d}_N$ は発散し，$\tilde{x}_N$ でソフトウェアテストを停止することが常に最適となる．すなわち，上述の命題 (ii) は，$T_{LC} \ll \tilde{x}_N$ であり，ソフトウェアに内在するすべてのフォールトを除去することは不可能であることを仮定している．

具体的に，将来のフォールト発見時間 $\tilde{x}_i\ (i = n+1, \cdots, N-1)$ を推定する方法として，回帰モデルや階層型ニューラルネットワークを適用することが有効であると考えられる．また，本稿では未知のデータを含む N 個の標本から構成される経験分布から，経済的に最適なリリース時期を推定する方法について述べたが，(i) 初期フォールト数 N がフォールト発見時刻と独立な確率変数である場合や，(ii) 経験分布に Kaplan and Meier 推定量 [5] を用いた場合，(iii) 最適停止ルールが look-ahead policy となる場合等に対して，さらなる考察が必要であろう．

参考文献

[1] 山田茂，ソフトウェア信頼性評価技術，HBJ 出版局，東京，1989.

[2] A. L. Goel and K. Okumoto, "Time-dependent error-detection rate model for software reliability and other performance measures ", *IEEE Trans. Reli.*, Vol. R-28, pp. 206-211, 1979.

[3] S. Yamada and S. Osaki, "Software reliability growth modeling : models and applications", *IEEE Trans. Soft. Eng.*, Vol. SE-11, pp. 1431-1437, 1985.

[4] Y. Shinohara, T. Dohi and S. Osaki, "Conparisons of optimal release policies for software systems", *Proc. 20th International Conference on Computers and Industrial Engineering*, pp. 493-496, 1996.

[5] E. L. Kaplan and P. Meier, "Nonparametric estimation from incomplete observations", *J. Amer. Stat. Asso.*, Vol. 53, pp. 457-481, 1958.

© 日本オペレーションズ・リサーチ学会．無断複写・複製・転載を禁ず．