2002年日本オペレーションズ・リサーチ学会
春季研究発表会

累積ベルヌーイ過程による離散型ソフトウェア信頼性モデルの統一化とパラメータ推定手法に関する考察

岡村寛之 (01013754)†, 村山篤史 ‡, 土肥正 (01307065)†
† 広島大学大学院工学研究科情報工学専攻
‡ 広島大学工学部第二類（電気系）

1. はじめに

現在までに，ソフトウェアの信頼性を定量的に評価する方法として様々なソフトウェア信頼性モデルが提案されている．一般的に知られている手法は，テスト段階あるいは運用段階におけるフォールト発見事象を確率過程で表現する手法である．特に，フォールト発生過程が非同次ポアソン過程 (NHPP) によって表現する NHPP モデルはモデルの柔軟さや数理的な取扱いが容易であるため，これまでに多数のモデルが提案されている [1].

一般的に NHPP モデルは時刻 t までに発見されるフォールトの期待値を表す平均値関数 $\Lambda(t)$ に関する微分方程式から構築される．例えば NHPP モデルの代表である指数形信頼度成長モデル [2] は，単位時間当たりに発見されるフォールト数が残存フォールト数に比例するという仮定の下で構築される．つまり，平均値関数が次の微分方程式を満たす NHPP で表現される．

$$\frac{d\Lambda(t)}{dt} = b[a - \Lambda(t)],$$

$$\Lambda(0) = 0.$$

ここで，パラメータ a, b はそれぞれ総期待エラー数と残存エラー 1 個当たりのエラー発見率である．NHPP に基づいたソフトウェア信頼性モデルは連続時間経過に対するフォールト発生事象を取り扱ってる．しかしながら，実際のソフトウェア開発工程におけるソフトウェア信頼性モデルの利用に対する観点から，離散時間に基づいたソフトウェア信頼性モデル（離散型ソフトウェア信頼性モデル）も提案されている．離散型ソフトウェア信頼性モデルに対して，文献 [3] では NHPP モデルを構築する際に用いられる微分方程式を直接離散化し，離散化後の差分方程式に基づいた離散型ソフトウェア信頼性モデルについて議論を行っている．

本研究では，離散型ソフトウェア信頼性モデルを統一的に扱うため，累積ベルヌーイ過程を用いたソフトウェア信頼性モデル [4] について考察を行う．また，そのパラメータ推定手法として，EM (Expectation-Maximization) [5] による推定アルゴリズムを提案する．

2. 累積ベルヌーイ過程によるソフトウェア信頼性モデル

累積ベルヌーイ過程に基づいたソフトウェア信頼性モデルは，ある時間区間における累積フォールト数に対する分布関数を各時間区間により異ったフォールト発見確率を持つ二項分布によって表現することで得られる．各時間区間 $i = 1, 2, \ldots$ に対するフォールト発見確率をそれぞれ $\pi = (p_1, p_2, \ldots)$ とし，時間区間 i で発見されるフォールト数を確率変数 Y_i $(i = 1, 2, \ldots)$ で表す．また，ソフトウェア内に潜在する総フォールト数を N_0，時間区間 i までに発見される累積フォールト数を $N(i) = Y_1 + Y_2 + \cdots + Y_i$ としたとき，$i = 2$ における確率関数は次のように表される．

$$\begin{aligned}
&\Pr\{N(2) = k\} \\
&= \sum_{m=0}^{k} \binom{N_0}{m} {p_1}^m (1 - p_1)^{N_0 - m} \\
&\quad \times \binom{N_0}{k - m} {p_2}^{k-m} (1 - p_2)^{N_0 - (k - m)} \\
&= \binom{N_0}{k} \{1 - (1 - p_1)(1 - p_2)\}^k \\
&\quad \times \{(1 - p_1)(1 - p_2)\}^{N - k}.
\end{aligned}$$

同様にして時間区間 n までに発見される累積フォールト数は次の 2 項分布で表現される．

$$\Pr\{N(n) = k\} = \binom{N_0}{k} \left\{1 - \prod_{i=1}^{n}(1 - p_i)\right\}^k \left\{\prod_{i=1}^{n}(1 - p_i)\right\}^{N-k}.$$

特に総フォールト数 N_0 がパラメータ ω のポアソン分布に従うとき上式は

$$\Pr\{N(n) = k\} = \frac{(\omega\{1 - \prod_{i=1}^{n}(1 - p_i)\})^k}{k!} \times \exp\left\{-\omega\left(1 - \prod_{i=1}^{n}(1 - p_i)\right)\right\}$$

となり，連続型 NHPP モデルの時間に関する離散化を行ったモデル（離散型ソフトウェア信頼性モデル）と等価となる．

3. EM アルゴリズムによるパラメータ推定

累積ベルヌーイ過程に基づいた離散型ソフトウェア信頼性モデルに対するパラメータ推定手法について考察する．ソフトウェア信頼性モデルのパラメータ推定では，尤度方程式を

数値的に解いてパラメータを決定する最尤法が用いられている．しかしながら，尤度方程式を数値的に解く方法では，推定すべきパラメータが多数ある場合に計算効率の観点から必ずしも有効であるとは言えない．そこで本論文では累積ベルヌーイ過程に基づいた離散型ソフトウェア信頼性モデルに対して，EM アルゴリズムに基づいたパラメータ推定手法を提案する．

初期時刻においてソフトウェアに潜在するフォールトの総数を N_0 とし，時間区間 $i=1,2,\ldots$ で観測されるフォールト発見個数を確率変数 $Y_1, Y_2, \ldots$ で表現する．いま，全ての確率変数 $Y_1, Y_2, \ldots$ の観測値（完全データ）

$$\mathcal{D}_\infty = (y_1, y_2, \ldots)$$

が得られたとする．このとき，完全データに対する対数尤度関数は

$$\begin{aligned}\log L(\omega, p_1, \ldots, p_\infty | \mathcal{D}_\infty) &= n \log\omega - \omega - \sum_{i=1}^{\infty} \log(y_i!) \\ &\quad + \sum_{i=1}^{\infty} \left\{ y_i \log p_i + \sum_{k=i+1}^{\infty} y_k \log(1-p_i) \right\}\end{aligned}$$

となる．従って完全データが与えられた下でパラメータ ω, $\boldsymbol{\pi} = (p_1, \ldots, p_\infty)$ に対する推定値は

$$\hat{\omega} = \sum_{i=1}^{\infty} y_i$$

$$\hat{\boldsymbol{\pi}} = \underset{\boldsymbol{\pi}}{\operatorname{argmax}} \left\{ \sum_{i=1}^{\infty} g_i(y_i, p_i) \right\}$$

として与えられる．ここで

$$g_i(y_i, p_i) = y_i \log p_i + \sum_{k=i+1}^{\infty} y_k \log(1-p_i) \quad \text{for } i = 1, 2, \ldots$$

である．

EM アルゴリズムは不完全なデータが観測された下で構築される．いま，フォールト発見に対する不完全データ

$$\mathcal{D}_m = (y_1, y_2, \ldots, y_m) \quad (m < \infty)$$

が得られた時，EM アルゴリズムによる推定値は不完全データ $\mathcal{D}_m$ の下での期待対数尤度を最大にするパラメータを用いて，以下の更新式 $k=0$ から繰り返し評価することで算出される．

$$\hat{\omega}^{(k+1)} = \mathrm{E}\left[\sum_{i=1}^{\infty} Y_i \,\middle|\, \mathcal{D}_m; \hat{\omega}^{(k)}, \hat{\boldsymbol{\pi}}^{(k)} \right],$$

$$\hat{\boldsymbol{\pi}}^{(k+1)} = \underset{\boldsymbol{\pi}}{\operatorname{argmax}} \left\{ \mathrm{E}\left[\sum_{i=1}^{\infty} g_i(Y_i, p_i) \,\middle|\, \mathcal{D}_m; \hat{\omega}^{(k)}, \hat{\boldsymbol{\pi}}^{(k)} \right] \right\},$$

ここで，$\mathrm{E}[\cdot; \omega, \boldsymbol{\pi}]$ はパラメータ ω と $\boldsymbol{\pi}$ の下での期待演算子を表す．上記の期待値演算に対して次の結果を与える．任意の可測関数 $h(\cdot)$ に対して

$$\mathrm{E}\left[\sum_{i=1}^{\infty} h(i, Y_i) \,\middle|\, \mathcal{D}_m; \omega, \boldsymbol{\pi} \right] = \sum_{i=1}^{m} h(i, y_i) + \sum_{i=m+1}^{\infty} \mathrm{E}[h(i, X_i)]$$

が成立する．ここで X_i はパラメータ $\omega p_i \prod_{k=1}^{i-1}(1-p_k)$ のポアソン分布に従う確率変数である．

特にパラメータ列 $\boldsymbol{\pi}$ に対する制約を特定することで，より具体的なモデルおよびその推定アルゴリズムを導出することができる．例として，パラメータ列 $p_1, p_2, \ldots$ に対して

$$p_i = c \quad (0 < c < 1) \quad \text{for } i = 1, 2, \ldots$$

とすると，対応する離散型ソフトウェア信頼性モデルは文献[3]における離散化指数形ソフトウェア信頼度成長モデルになる．上式の制約のもと期待対数尤度を最大にする最適化問題を考えることで，次の離散化指数形ソフトウェア信頼度成長モデルに対する推定アルゴリズムを得る．

$$\hat{\omega}^{(k+1)} = \sum_{i=1}^{m} y_i + \hat{\omega}^{(k)}(1-c)^m,$$

$$\hat{c}^{(k+1)} = \left\{ \sum_{i=1}^{m} y_i + \hat{\omega}^{(k)}(1-c)^m \right\} \Big/ \left\{ \sum_{i=1}^{m} (i+1) y_i + \hat{\omega}^{(k)}(1-c)^m (1 + m + 1/c) \right\}.$$

参考文献

[1] 山田，ソフトウェア信頼度モデル — 基礎と応用，日科技連出版社，東京，(1994).

[2] Goel, A. and Okumoto, K., Time-dependent error-detection rate model for software reliability and other performance measures, *IEEE Trans. Reliab.*, Vol. 39, pp. 206–211 (1979).

[3] 井上，山田，佐藤，差分方程式に基づくソフトウェア信頼度成長モデルに考察，2001 年度日本オペレーションズ・リサーチ学会秋季研究アブストラクト集，pp. 66–67 (2001).

[4] Wakana, N., Dohi, T. and Osaki, S., Bivariate cumlative binomial trial process and its application to reliability assessment, *Proc. 6th ISSAT Int'l Conf. on Reliab. and Qualty in Design*, pp. 122–126 (2000).

[5] 岡村，渡部，土肥，尾崎，EM アルゴリズムに基づいたソフトウェア信頼性モデルの推定，信学技法（信頼性），Vol. 101, pp. 35–40 (2001).