

潜入ソフトウェアフォールトを考慮した 不完全デバッグモデルと適合性評価

01702425 鳥取大学工学部 山田茂 YAMADA Shigeru

1 はじめに

本研究では、ソフトウェア信頼度成長モデル [1] を構築する際の完全デバッグの仮定を非現実的なものと考え [2]、より実際的な不完全デバッグ環境を考慮したモデルを議論し、その適合性比較と妥当性を考察する。

2 モデルの記述 [3]

ソフトウェアの動的環境において発生するソフトウェア故障には、稼働前に潜在する固有フォールトに起因するものと、稼働中にランダムに潜入するフォールトに起因するものとの2種類があると仮定する。このソフトウェア故障発生現象に対して非同次ポアソン過程 (NHPP) を導入する。前者のソフトウェア故障発生率は、稼働時間 t の経過と共に検出される固有フォールト数の2種類の時間的傾向を考慮するため $a_i(t)$ ($i=1,2$) とする。また、後者のソフトウェア故障発生率は発生時間に関してランダムであり、一定の λ ($\lambda > 0$) により表す。したがって、これらのソフトウェア故障の発生現象を同時に考えるとき、稼働時間 t におけるソフトウェア故障発生率 $h_i(t)$ 、および時間区間 $(0,t]$ において発生する総期待ソフトウェア故障数 (または発見される総期待フォールト数) $H_i(t)$ は

$$h_i(t) = \lambda + a_i(t) \quad (i=1,2), \quad (1)$$

$$\left. \begin{aligned} H_i(t) &= \lambda t + A_i(t) \\ A_i(t) &= \int_0^t a_i(x) dx \end{aligned} \right\}, \quad (2)$$

となる。さらに、式 (2) の $a_i(t)$ に対して NHPP に基づく指数形および遅延 S 字形ソフトウェア信頼度成長モデル [2] を適用した

$$a_1(t) = abe^{-bt} \quad (a > 0, b > 0), \quad (3)$$

$$a_2(t) = ab^2te^{-bt} \quad (a > 0, b > 0), \quad (4)$$

を考察することにする。ここで、 a は初期内蔵固有フォールト数の期待値、 b は固有フォールトに起因する1個当りのソフトウェア故障率を表す。

上記のモデルに対して、稼働中に潜入するフォールトに起因するソフトウェア故障発生現象を暗黙のうちに考慮しているワイブル過程モデルおよび対数型ポアソン実行時間モデル [2] についても、NHPP に基づき本研究における不完全デバッグモデルとして考察する。ワイブル過程モデルにおける時間区間 $(0,t]$ において発生する総期待ソフトウェア故障数および稼働時間 t におけるソフトウェア故障率は、それぞれ

$$\left. \begin{aligned} \gamma(t) &= \alpha t^\beta \\ h_\gamma(t) &= \alpha \beta t^{\beta-1} \end{aligned} \right\}, \quad (5)$$

であり、対数型ポアソン実行時間モデルについては、それぞれ

$$\left. \begin{aligned} \mu(t) &= \ln(\lambda_0 \theta t + 1) / \theta \quad (\lambda_0 > 0, \theta > 0) \\ h_\mu(t) &= \lambda_0 / (\lambda_0 \theta t + 1) \end{aligned} \right\}, \quad (6)$$

となる。ここで、 α および β は定数パラメータ、パラメータ λ_0 は初期故障強度、 θ はソフトウェア故障1個当りの故障強度の減少率を表す。

3 信頼性評価尺度

2. で議論した NHPP モデルの平均値関数および強度関数をそれぞれ $H(t)$ および $h(t)$ とすると、以下のような定量的なソフトウェア信頼性評価尺度を導出できる。

総稼働時間が $t(t \geq 0)$ のとき、時間区間 $(t, t+x)$ ($x \geq 0$) においてソフトウェア故障の発生しない条件付き確率は、

$$R(x|t) \equiv \exp[H(t) - H(t+x)], \quad (7)$$

により与えられ、ソフトウェア信頼度と呼ばれる。ソフトウェア故障の発生率や発生頻度を表すハザードレートは、

$$z(x|t) \equiv -\frac{d}{dx} R(x|t) / R(x|t) = h(t+x), \quad (8)$$

となる。さらに、総稼働時間が t のときの平均ソフトウェア故障発生時間間隔は

$$E[X|t] = \int_0^\infty R(x|t) dx, \quad (9)$$

により与えられる。

4 パラメータの推定と適合性評価尺度

本研究では、モデルパラメータの推定方法として最尤法を適用する。そこで、ソフトウェアの動的環境で、一定の時間区間 $(0, t_k]$ において発見された総フォールト数 y_k に関する n 組のフォールト発見数データ (t_k, y_k) ($k=1, 2, \dots, n$) が観測されたものとする。このとき、平均値関数 $H(t)$ をもつ NHPP モデルの対数尤度関数を

$$\ln L = \sum_{k=1}^n (y_k - y_{k-1}) \cdot \ln[H(t_k) - H(t_{k-1})] - H(t_n) - \sum_{k=1}^n \ln[(y_k - y_{k-1})!], \quad (10)$$

として、最尤法によりモデルパラメータを推定する。

一方、いままで議論した4つのモデルと従来の指数形および遅延 S 字形ソフトウェア信頼度成長モデルの基本モデルを含めた6つのモデルの実測データに対する適合性比較を行い、各モデルの実測データに対する適合性と特徴を考察する。適

合性評価基準としては、次のものを用いる。

(1) 平均偏差二乗和

実測値と推定値との二乗誤差の総和をデータ数で平均化して評価したもの。

(2) 対数尤度値

実際データに対する同時確率密度関数(尤度関数)、すなわち実際データの観測される確率の対数値。

(3) AIC

最尤法によってパラメータ推定を行ういくつかのモデルのうちから、適合性のよい最適なモデルをモデルパラメータ数を考慮して選択する1つの基準。

(4) 予測相対誤差

テスト終了時刻における累積フォールト数の予測値と実測値の相対誤差。

(5) 相関係数

実際データと推定値との相関係数。

5 数値例

本研究では、不完全デバッグ環境下にあるデータと仮定し、実際のテスト工程で観測された12セットのフォールト発見数データに各モデルを適用し、上記の適合性評価基準を用いて適合性比較を行った。ここでは、データセット DS1 を取り上げて比較を行い、その結果を表1と図1に示す。表1より、全ての適合性評価基準に対して、モデル μ とモデル H_1 の適合性が良いことが分かる。なお、表2には各モデルパラメータの最尤推定値を示した。ここで、モデル名には各平均値関数の記号を用いている。次に、予測相対誤差を示した図1より、テスト進捗率40%以降、各モデルの推定値が安定していること、またモデル μ およびモデル H_1 の適合性の良さが分かる。図2には、各モデルによる式(7)のソフトウェア信頼度の推定結果も示した。

表1 DS1に対する適合性評価結果

	モデル m	モデル H_1	モデル γ	モデル μ
平均偏差二乗和	35.5183	13.8601	39.1075	6.8729
対数尤度値	-57.219	-53.766	-55.358	-52.024
AIC	118.438	113.532	114.715	108.047
相関係数	0.9983	0.9995	0.9994	0.9997

表2 DS1に対するモデルパラメータの最尤推定値

	\hat{a}	\hat{b}	$\hat{\lambda}$	$\hat{\alpha}$	$\hat{\beta}$	$\hat{\lambda}_0$	$\hat{\theta}$
m	142.315	0.1246	/	/	/	/	/
H_1	76.1759	0.3302	2.3938	/	/	/	/
γ	/	/	/	31.987	0.4496	/	/
μ	/	/	/	/	/	36.661	0.0226

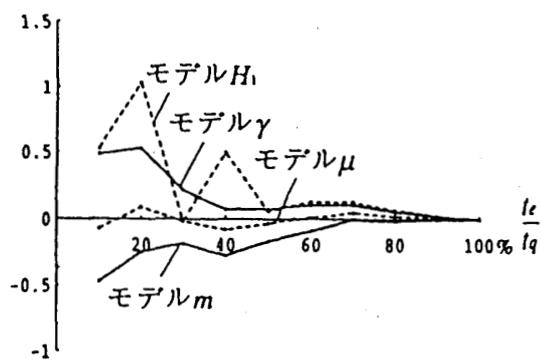


図1 DS1に対する予測相対誤差

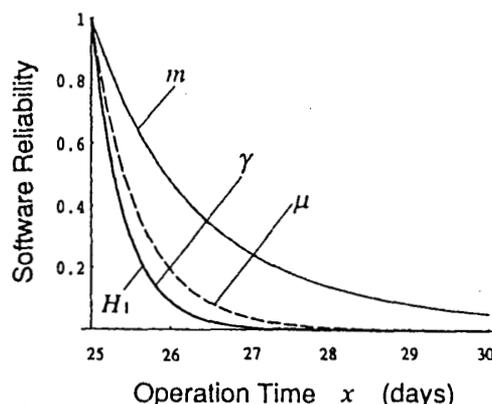


図2 DS1に対する推定されたソフトウェア信頼度

このような適合性比較結果から考えると、従来のモデルに比べて、不完全デバッグ環境を考慮したモデルのうち平均値関数 $H_i(t)(i=1,2)$ および $\mu(t)$ をもつNHPPモデルの実測データに対する適合性がかなり良いことが分かった。特に、不完全デバッグ環境を前提とすれば、陽に潜入フォールトの存在を明示したモデル $H_i(i=1,2)$ は、その定量化が可能であることから開発管理者にとっては有効であろう。

謝辞

本研究を進めるにあたり協力頂いた本学大学院生の三木貴史君に感謝致します。

参考文献

- [1] 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)。
- [2] M. L. Shooman: *Software Engineering: Design, Reliability, and Management*, McGraw-Hill, New York (1983)。
- [3] 山田茂:“潜入フォールトによる不完全デバッグを考慮したソフトウェア信頼度成長モデル”, 電子情報通信学会論文誌, Vol. J80-A, No. 2, pp. 363-370 (1997)。