

## 対数型ポアソン実行時間モデルに基づく オープンソースソフトウェアに対する信頼性評価に関する一考察

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

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

### 1 はじめに

現在のソフトウェア開発は、ネットワークを基にした分散開発が主流となっている。中でも、世界中の誰もが開発に参加できるという特徴をもつオープン・ソース・プロジェクトは分散型ソフトウェア開発形態の成功例として特に注目されている。こうしたLinux<sup>1</sup>やApache, Sambaなど<sup>2</sup>に代表されるオープンソースソフトウェア（Open Source Software, 以下OSSと略す）は、サーバ用途を主として社会の中でも基幹的で重要な役割を担っている[1]。

本研究では、オープン・ソース・プロジェクトの下で分散共同開発されたソフトウェアシステムに対する信頼性評価法を提案する。特に、オープン・ソース・プロジェクトの下で開発されたUNIX系OS上で動作するXデスクトップ環境を1例にとり、意思決定手法の1つであるAHP（Analytic Hierarchy Process）手法と、検出可能フォールト数が無限であるという仮定に基づい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）を使用した信頼性評価法について考察する。さらに、実際のフォールト発見数データに基づいた数値例を示す。

### 2 各コンポーネントに対する信頼性評価

#### 2.1 SRGMに基づく信頼性評価

従来から、ソフトウェアの信頼性を定量的に評価する方法として、SRGMによる方法がとられている。中でも非同次ポアソン過程（nonhomogeneous Poisson process, 以下NHPPと略す）モデルは、実用上極めて有効でありモデルの簡潔性が高いゆえにその適用性も高く、実際のソフトウェア信頼性評価に広く応用されている。

本研究では、各コンポーネント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NHPPモデル[2]のうち最適なモデルを適用する。

- 指数形SRGM
- 習熟S字形SRGM

#### 2.2 AHPに基づく各コンポーネントに対する重み係数の推定

ソフトウェアの信頼性評価手法の開発において、各コンポーネントのシステム全体への信頼性に与える影響を考慮しようとする場合、プログラムパス、コンポーネントの規模、フォールト報告者のスキルなどの、様々に絡み合った要因を捉える必要があると考えられる。しかしながら、これは困難であることが予想される。したがって本研究では、こうした複雑な状況下でシステム全体の信頼性に対する各

コンポーネントの影響度合いを推定するために、一般には主観的判断の合理的合成方法として知られているAHPを利用し、システム全体の信頼性に対する各コンポーネントの重要度を表す重み係数の推定を行う[3]。特に、適用される評価基準としては、各コンポーネントに対して発見されたフォールトの重要度、コンポーネントの規模、フォールト報告者のスキルといった要因が挙げられる。各コンポーネントにおけるAHPの評価基準に対するウエイトをそれぞれ $w_i (i = 1, 2, \dots, n)$ とすれば、各評価基準に対するウエイトを次式の幾何平均により求めることができる。

$$\alpha_i = \sqrt[n]{\prod_{j=1}^n x_{ij}},$$

$$x_{ij} = \frac{w_i}{w_j} \quad (1)$$

したがって、各ソフトウェアコンポーネントに対するウエイトは、

$$p_i = \frac{\alpha_i}{\sum_{i=1}^n \alpha_i} \quad (2)$$

により与えられる。

### 3 システム全体に対する信頼性評価

本研究で取り上げるOSSの動作環境は、様々なアプリケーションソフトウェアから影響を受け易く、従来のような同一組織内で開発され、単体で動作するソフトウェアシステムとは大きく環境が異なる。こうしたソフトウェア間の相互作用により、発見されるフォールト数も一定の値に収束することなく、将来的には増加し続けるものと考えられる。

本研究では、検出可能フォールト数が無限であると仮定されたNHPPに基づく対数型ポアソン実行時間モデルを適用する。時間区間 $(0, t]$ で発見される総期待フォールト数を表す平均値関数 $\mu(t)$ は、

$$\mu(t) = \frac{1}{\theta - P} \ln[\lambda_0(\theta - P)t + 1]$$

$$(0 < \theta, 0 < \lambda_0, 0 < P < 1), \quad (3)$$

により与えられる。ここで、パラメータ $\lambda_0$ は初期故障強度、パラメータ $\theta$ 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。また、パラメータ $P$ はシステム全体に及ぼすコンポーネントの影響率を表す。これは、各コンポーネントに対して推定されたパラメータ $b_i$ と2.2のAHP手法により推定された重みパラメータ $p_i$ との重み付き平均により表されるものとし、

$$P = \sum_{i=1}^n p_i \cdot b_i \quad (4)$$

<sup>1</sup>Linuxは、Linus Torvaldsの米国およびその他の国における登録商標あるいは商標です。

<sup>2</sup>その他記載している会社名、商品名は一般に各社の商標または登録商標です。

により定義する。ここで、 $n$ はソフトウェアシステムのコンポーネント数を表す。さらに、 $p_i$ は各コンポーネントに対する重みパラメータを表し、システム全体に対する各コンポーネントの重要度を表す。また、 $b_i$ は*i*番目のコンポーネントに対する指数形SRGMおよび習熟S字形SRGMに含まれるフォールト発見率を表す定数パラメータを表す。

#### 4 数値例

##### 4.1 各コンポーネントに対する信頼性評価

実際のオープン・ソース・プロジェクトにおけるバグトラッキングシステムから採取されたフォールトデータを適用した数値例を示す。本研究では、6つの主要コンポーネントから構成されるXfceと呼ばれるOSSを取り上げる。

各コンポーネントの累積フォールト発見数データを図1に示す。次に、2.2のAHPに基づく各コンポーネントに対する重みパラメータ $p_i$ ( $i = 1, 2, \dots, 6$ )の推定結果を表1に示す。特に、AHPの評価基準としては、各コンポーネントに対するフォールトの重要度、フォールト報告者の重要度、フォールト修正者の重要度を取り上げた。

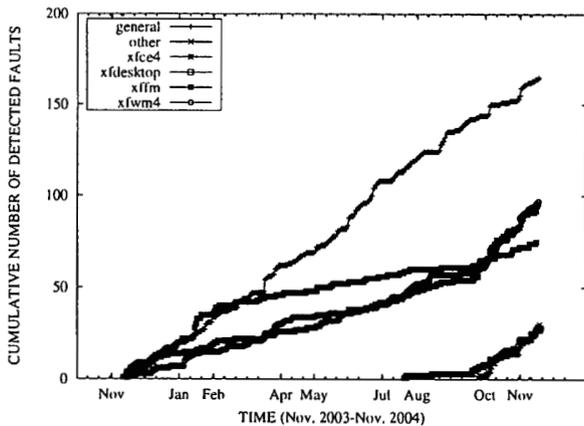


図1：各コンポーネントの累積フォールト発見数データ。

表1：AHPに基づく各コンポーネントに対する重み係数の推定結果。

Component	Weight parameter $p_i$
general	0.055219
other	0.44820
xfce4	0.091727
xfdesktop	0.18165
xffm	0.11970
xfwm	0.10351

##### 4.2 システム全体に対する信頼性評価

次に、各コンポーネントに対して適用されたSRGMに含まれる未知パラメータを最尤法により推定された結果を踏まえて、Xfceの信頼性評価の1例を示す。式(3)における累積フォールト発見数の期待値の推定値 $\hat{\mu}(t)$ を図2に示す。

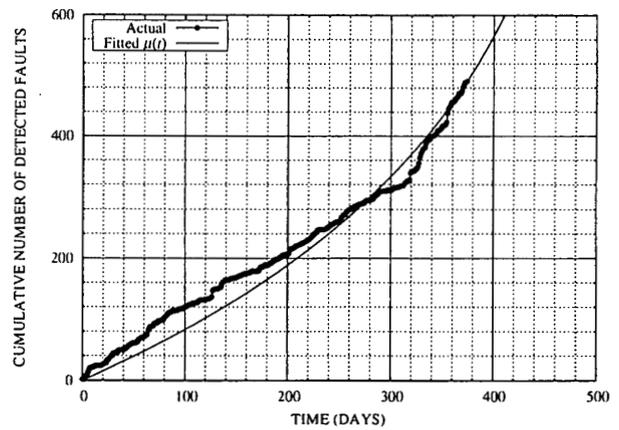


図2：推定された累積フォールト発見数の期待値、 $\hat{\mu}(t)$ 。

##### 4.3 適合性評価

システム全体に対する信頼性評価法として、発見フォールト数が有限である場合を仮定した習熟S字形SRGMを適用した信頼性評価法が提案されている[1]。ここでは、本研究で提案されたモデルと過去に提案されたモデルとの適合性比較を行う。適合性評価基準としては、AIC(Akaike's Information Criterion, 赤池情報量規準)および平均偏差2乗和(mean squared errors, 以下MSEと略す)を採用した。

表2：適合性比較結果。

Compared models	AIC	MSE
Proposed model	1258.7	586.45
Conventional model	1271.4	607.16

表2から、本研究で提案されたモデルの実測データに対する適合性が良いことが確認できる。

#### 5 おわりに

本研究では、オープン・ソース・プロジェクトの下で分散共同開発されているソフトウェアシステムに対する信頼性評価法について議論した。特に、意思決定手法の1つであるAHP手法に基づき各コンポーネントに対する重み係数を推定し、発見フォールト数が無限であると仮定された対数型ポアソン実行時間モデルに基づいた信頼性評価法を提案した。また、実際のフォールト発見数データに対する数値例を示した。

#### 参考文献

- [1] 田村慶信, 山田茂, 木村光宏, “オープンソース共同開発環境におけるソフトウェア信頼性評価法,” ソフトウェアジャパン2004論文集(情報処理学会), pp. 1-8, 2004.
- [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.
- [3] 加藤豊, 小沢正典, ORの基礎—AHPから最適化まで—, 実教出版株式会社, 東京, 1998.