

AHP を用いた Xfce デスクトップ環境の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、インターネット時代の到来とオープン系システム開発案件の増大、短期開発の要求の増加によって、従来の古典的スタイルでの開発形態では対応できない状況が生まれている。こうした状況から、ネットワークを基にしたソフトウェアの分散開発、およびソフトウェアそのものの分散化がさらに拡大してきた。現在、分散ソフトウェア共同開発は、同一企業内における開発形態から、複数のソフトウェアハウスや同一企業内、複数の企業間での遠隔地間共同開発、さらにはオープン・ソース・プロジェクトなどの様々な形態が存在する [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、UNIX系OS上で動作するXデスクトップ環境の1つであるXfceを一例にとり、意思決定手法の1つであるAHP (Analytic Hierarchy Process) 手法とソフトウェア信頼度成長モデル (software reliability growth model, 以下SRGMと略す) に基づい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手法として、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、実利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。このNHPPモデルは、所定の時間区間内に発見されるフォールト数や発生するソフトウェア故障数を観測して、これらの個数を数え上げる計数過程 $\{N(t), t \geq 0\}$ を導入し、以下の式で与えられる確率変数すなわちポアソン過程を仮定するSRGMである [2]。

$$\Pr\{N(t) = n\} = \frac{\{H(t)\}^n}{n!} \exp[-H(t)] \quad (1)$$

$(n = 0, 1, 2, \dots)$

ここで、 $\Pr\{\cdot\}$ は確率を表し、 $H(t)$ は時間区間 $(0, t)$ において発見される総期待フォールト数、すなわち $N(t)$ の期待値を表し、NHPPの平均値関数と呼ばれる。

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

さらに、モデルに含まれる未知パラメータの推定方法として最尤法を適用する。

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、各コンポーネントでのデバッグの状況やその良し悪しが、システム全体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、一対比較行列は、

$$A = \begin{bmatrix} \frac{w_1}{w_1} & \frac{w_1}{w_2} & \dots & \frac{w_1}{w_n} \\ \frac{w_2}{w_1} & \frac{w_2}{w_2} & \dots & \frac{w_2}{w_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{w_n}{w_1} & \frac{w_n}{w_2} & \dots & \frac{w_n}{w_n} \end{bmatrix}, \quad (2)$$

となる。この一対比較行列から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。

$$\alpha_i = \left(\prod_{i=1}^n w_i \right)^{\frac{1}{n}}. \quad (3)$$

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、

$$p_i = \frac{\alpha_i}{\sum_{i=1}^n \alpha_i}, \quad (4)$$

により与えられる。

3 システム全体に対する信頼性評価

フォールト発見の意味と同様に重要なソフトウェア信頼性要因としては、フォールト発見の独立性に関するものがある。ソフトウェア構造とプログラムパスの観点から考えて、発見されるフォールトには、潜在するフォールトが互いに独立で発見の容易なもの、独立ではなく発見にはその前に他のフォールトの発見を必要とするものとが混在すると仮定した習熟 S 字形 SRGM が提案されている。その

平均値関数 $H_c(t)$ は,

$$H_c(t) = \frac{a(1 - e^{-bt})}{(1 + C \cdot e^{-bt})} \quad (a > 0, b > 0, C > 0), \quad (5)$$

により与えられる。ここで、パラメータ a は最終的に発見される総期待フォールト数、パラメータ b はフォールト 1 個当りのソフトウェア故障発見率またはフォールト認知率を表す。本研究では、パラメータ C は各コンポーネントに対して推定されたパラメータ c_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,

$$C = \frac{\sum_{i=1}^n p_i \cdot c_i}{\sum_{i=1}^n p_i}, \quad (6)$$

により定義する。ここで、 n はソフトウェア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、 c_i は i 番目のコンポーネントに対する習熟 S 字形 SRGM に含まれる習熟係数を表す。

4 数値例

4.1 各コンポーネントに対する信頼性評価

Xfce のフォールトデータを適用した数値例を示す。本研究で用いたデータは、6 つのコンポーネントから構成された Xfce デスクトップ環境におけるバグトラッキングシステムから採取されたものである。

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、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。

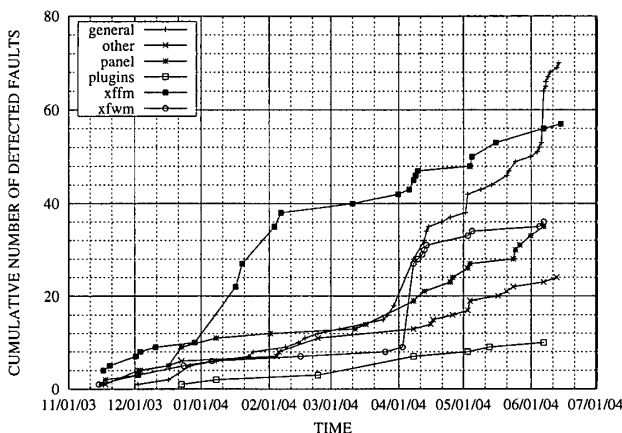


図 1: 各コンポーネントの累積フォールト発見数データ。

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 デスクトップ環境の信頼性評価の一例を示

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	0.20314
other	0.07975
panel	0.06936
plugins	0.01543
xffm	0.43629
xfwm	0.19604

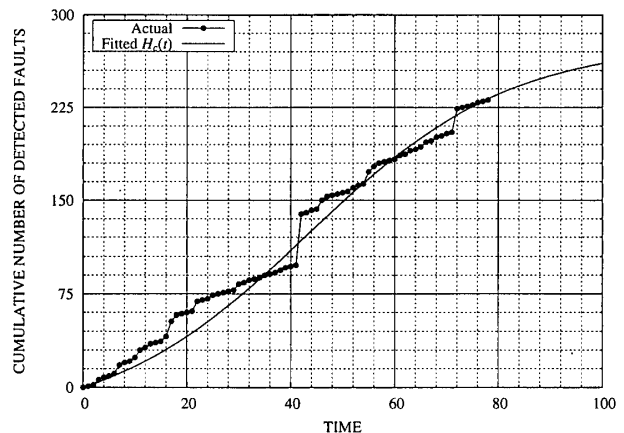


図 2: 推定された累積フォールト発見数の期待値, $\widehat{H}_c(t)$.

す。式 (5) における累積フォールト発見数の期待値の推定値 $\widehat{H}_c(t)$ を図 2 に示す。

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 UNIX 系 OS 上で動作するデスクトップ環境の 1 つである Xfce 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、習熟 S 字形 SRGM に基づいた信頼性評価法を提案した。また、実際の Xfce に基づいたフォールト発見数データに対する数値例を示した。

本研究で対象としている Xfce は、デスクトップ上で様々なアプリケーションソフトウェアが動作する環境であるために、従来のような単体で動作するソフトウェアシステムとは環境が大きく異なることが考えられる。今後は、こうしたアプリケーションソフトウェアのデスクトップ環境に及ぼす影響などについても考慮する必要がある。

参考文献

- [1] 松本 正雄, 小山田 正史, 松尾谷 徹, ソフトウェア開発検証技法, 電子情報通信学会, 東京, 1997.
- [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.