

ソフトウェア信頼度成長曲線に関する統合モデルと有効性の検証

古山 恒夫

中川 豊

NTTソフトウェア研究所

NTT技術情報センター

これまで提案されたソフトウェア信頼度成長モデル (Software Reliability Growth Model: SRGM) を統一して扱えるモデル—"統合モデル"を提案する。このモデルは、これまでのSRGMだけでなく、これまでのSRGMが扱えなかった領域のデータも統一して扱えるため、複雑なフォールト成長曲線の残存フォールトをこれまでより高い精度で推定できる。本論文では、最初に、統合モデルをひとつの微分方程式で表し、その一般解を示す。次に、これまでの代表的なSRGMで用いられている曲線が、その微分方程式の特殊解であることを示し、従来提案されたモデルの間を関係を明確にする。また、これまでのモデルでは最適な推定が行えなかったデータ領域を明らかにする。最後に、理論データと実データを用いて、統合モデルが従来モデルより高い精度で残存フォールトを推定できることを示す。

A MANIFOLD GROWTH MODEL THAT UNIFIES SOFTWARE RELIABILITY GROWTH MODELS

Tsuneo Furuyama

Yutaka Nakagawa

NTT Software Laboratories

NTT R&D Information and Patent Center

Nippon Telegraph and Telephone Corporation

NTT Shinagawa Twins, 1-9-1 Kohnan, Minato-ku, Tokyo 108-19, Japan

This paper presents a manifold model that unifies existing software reliability growth models (SRGMs). This model, in addition to making it easy to select the most suitable model for each growth curve, makes it possible to predict the number of remaining faults for complicated faults growth curves with higher accuracy than previously. Using the manifold model, this paper clarifies the relationships between the existing SRGMs. Then using actual data, it compares the fault estimation accuracy of the manifold model and representative SRGMs, showing the usefulness of this model.

1. はじめに

信頼性の高いソフトウェアを予定期間内に開発するためには、残存フォールト数を高い精度で推定することが不可欠である。この課題を解決するために、これまで多くの方法が提案されてきたが、最も広く利用されているのは、試験工程で検出されるフォールトの累積値の傾向から残存フォールト数を推定する方法である。すなわち、時間対検出フォールト累積数の特性をもとにしたソフトウェア信頼度成長モデル (SRGM) を利用する方法である。

SRGMはすでに Jelinski & Moranda[1], Musa[2], Littlewood[3], Goel & Okumoto[4], Yamada & et al. [5], Ohba[6], Tohma & et al.[7], Kanoun & et al. [8], Karunanithi & et al. [9] など多くの研究者によって論じられている。しかし、これまで論じられているモデルは、指数形やS字形などの各モデル固有の定型的な特性を持った曲線をもとにしたものであり、利用できる範囲もそれぞれ限られている。そのため、あるモデルを固定的に使用すると、対象データの特性によっては残存フォールト数の推定精度が低くなることがある。実際に第一線のソフトウェア開発管理者は、適切なモデルを選択することが煩雑で難しいことから、特定のモデルを固定して利用することが多く、データによっては信頼性の予測が大きすぎるというリスクをつねに抱えている。

モデルのパラメータの値によって曲線の特性が変化するモデルもある[6]が、モデルのパラメータの決定は難しく、またその値が決定されると定型的特性曲線となってしまうので、上記の問題が解決されたとは言えない。

汎用モデルとしてNHPPモデルや超幾何モデルが提案されている。しかし、これらのモデルは考え方の枠組みを与えるものであり、具体的な関数型は新たに仮定を設けたり、プロジェクトの特性や条件を考慮して個々に選択・決定しなければならない。すなわち、関数型を決定しなければならないという問題は依然として残されている。

これまで、モデルの提案だけでなく、それらの

モデル間の優劣比較も議論されてきた。しかし、これまでの議論は、あるデータ群に対する各モデルの有効性と優劣を議論するものが多く、モデル間の相対的位置づけについての議論は行われなかった。すなわち、モデルは個々に独立したものであり、モデル間の数学的な関係については論じられていない。このように、SRGMについて解決すべき多くの課題が残されていた。

本論文では、これらの問題を解決することのできるモデル、すなわち指数形からS字形までの広い範囲の特性曲線を形成することができる統合モデルについて述べる。このモデルは、これまで提案された多くのモデルをカバーできるだけでなく、これまでモデル化されていなかった領域もカバーする。そのため、既存のSRGMより高い精度で残存フォールト数を推定することができる。また、開発管理者にとっては、適用モデルの選定作業の省力化が図れる。

2. 統合モデル

2.1 微分方程式

当該モデルを次の微分方程式で定義する：

$$\frac{d(y+\delta)}{dt} \cdot (y+\delta)^{\gamma-1} = \alpha \cdot \exp(-\beta t) \quad (1)$$

ここで、 $\alpha > 0$, $\beta \geq 0$, $\delta \geq 0$ である。 δ は、 $\gamma \leq 0$ のときにNHPPモデルの平均値関数の条件 ($t=0$ で $y=0$) を満たす解を持つために導入したパラメータで、この微分方程式の解となる関数のy軸上の移動を意味する。ただし、厳密には曲線の形にもわずかではあるが影響を与える。

式(1)の微分方程式は次のように解釈できる：

ある時点でのバグ検出率は、基本的には残存バグ数に比例する。(1)式右辺の指数項がそれを表す。ただし、プロジェクトの特性に応じてバグの発見過程が変わるので、この特性を、時間tまでに発見したバグ数yに依存した補正係数yのべき乗 ($\gamma-1$) で表す。補正係数の絶対値が大きいモデルほど、バグがなだれ現象的に発見される。すなわち、最初はバグがなかなか発見されないが、あるところから急激に増加する。時間が経過した

試験工程の最後では、指数関数で表される残存バグ数の減少ファクタ $(-\beta t)$ がきいて飽和する。

指数形モデルは補正係数がない場合である。

3. 2で述べるように習熟S字形モデル、ロジスティック曲線はべき乗が (-2) で強いなだけ現象がある場合である。

2. 2 一般解 ($\beta > 0$ の場合)

$\beta > 0$ の場合、式(1)の一般解は次のようになる。

(A) $\gamma \neq 0$ の場合

$$\frac{1}{\gamma} (y + \delta)^{\gamma} = -\frac{\alpha}{\beta} \exp(-\beta t) + m \quad (2)$$

ただし、 m は初期条件により定まる定数である。 $t=0$ のときの y の値を y_0 とする (以下同様) と

$$m = \frac{\alpha}{\beta} + \frac{(y_0 + \delta)^{\gamma}}{\gamma} \quad (3)$$

となる。これを (2) 式に代入して y について解くと、(2)式は次のようになる：

$$\begin{aligned} y &= \left[\frac{\alpha \gamma}{\beta} \{1 - \exp(-\beta t)\} + (y_0 + \delta)^{\gamma} \right]^{\frac{1}{\gamma}} - \delta \\ &= \left\{ \frac{\alpha \cdot \gamma + \beta (y_0 + \delta)^{\gamma}}{\beta} \right\}^{\frac{1}{\gamma}} \\ &\quad \times \left\{ 1 - \frac{\alpha \cdot \gamma}{\alpha \cdot \gamma + \beta (y_0 + \delta)^{\gamma}} \cdot \exp(-\beta t) \right\}^{\frac{1}{\gamma}} - \delta \quad (4) \end{aligned}$$

ここで、

$$\left\{ \frac{\alpha \cdot \gamma + \beta (y_0 + \delta)^{\gamma}}{\beta} \right\}^{\frac{1}{\gamma}} = N \quad (5)$$

$$\frac{\alpha \cdot \gamma}{\alpha \cdot \gamma + \beta (y_0 + \delta)^{\gamma}} = a \quad (6)$$

$$\beta = b \quad (7)$$

$$\frac{1}{\gamma} = c \quad (8)$$

とおくと、(4) 式は次のようになる：

$$y = N \{1 - a \cdot \exp(-b t)\}^c - \delta \quad (9)$$

(9)式は(5)式と(6)式より次式としても表せる：

$$y = N [(1 - a \cdot \exp(-b t))^c - (1 - a)] + y_0 \quad (10)$$

ここで、 N は(定数分 ($a=1$ の場合は 0) を差し引いて) 試験開始時点に含まれるバグ総数、 b はフォールト1個あたりの検出率を表す。 a と c は

成長曲線の基本特性を決定するプロジェクト毎のフォールト検出特性と解釈できる。

NHPPモデルに適用するためには、 $y_0=0$ であることが必要であるので、(10)式は、

$$y = N [(1 - a \cdot \exp(-b t))^c - (1 - a)] \quad (11)$$

となる。

3. 2で示すように、 $c > 0$ の場合は $a = 1$ としても近似を含めれば従来のモデルをカバーする。むやみにパラメータを増やさないために、 $c > 0$ の場合は(11)式で $a = 1$ とした次の式を用いる。

$$y = N \{1 - \exp(-b t)\}^c \quad (12)$$

(B) $\gamma = 0$ の場合

(1)式は次式となる：

$$\frac{d(y + \delta)}{dt} \cdot \frac{1}{y + \delta} = \alpha \cdot \exp(-\beta t) \quad (13)$$

(13)式の一般解は次のように表せる：

$$\ln(y + \delta) = -\frac{\alpha}{\beta} \exp(-\beta t) + m \quad (14)$$

m は初期条件によって定まる定数で、

$$m = \ln(y_0 + \delta) + \frac{\alpha}{\beta} \quad (15)$$

となる。これを(14)式に代入して y について解くと、

$$y = (y_0 + \delta) \exp\left(\frac{\alpha}{\beta}\right) \cdot \exp\left\{-\frac{\alpha}{\beta} \cdot \exp(-\beta t)\right\} - \delta \quad (16)$$

となる。ここで、

$$(y_0 + \delta) \exp\left(\frac{\alpha}{\beta}\right) = N \quad (17)$$

$$\frac{\alpha}{\beta} = k \quad (18)$$

$$\beta = b \quad (19)$$

とおけば、(16) 式は次のように表せる：

$$y = N \exp\{-k \exp(-b t)\} - \delta \quad (20)$$

(20)式は(17)式と(18)式より次式としても表せる：

$$y = N [\exp\{-k \exp(-b t)\} - \exp(-k)] + y_0 \quad (21)$$

ここで、 N と b はそれぞれ(10)式の N と b と、 k は(10)式の a および c と同じ意味の特性である。

(A)のときと同様な議論により、NHPPモデルを対象とする場合[10]には(21)式は、

$$y = N [\exp\{-k \exp(-b t)\} - \exp(-k)] \quad (22)$$

と表せる。なお、 k は(6)式、(8)式および(18)式か

ら、 $\gamma \rightarrow 0$ ($c \rightarrow \infty$, $a \rightarrow 0$) のときの c と a の積と等しいことに注意しておく。

2.3 一般解 ($\beta = 0$ の場合)

この節では(1)式において $\beta = 0$ の場合の一般解を求める。(1)式において、 $\beta = 0$, $\delta = 0$ とおくと、次の式が得られる。

$$\frac{dy}{dt} \cdot y^{\gamma-1} = \alpha \quad (23)$$

この方程式の一般解は、次式のようになる：

$$y = (\alpha \gamma t + m)^{\frac{1}{\gamma}} \quad (24)$$

ただし、 m は初期条件によって定まる定数で、

$$m = y_0^\gamma \quad (25)$$

となる。これを(24)式に代入すれば、次の式が得られる。

$$y = (\alpha \gamma t + y_0^\gamma)^{\frac{1}{\gamma}} \quad (26)$$

(26)式は $t \rightarrow \infty$ で $y \rightarrow \infty$ となる。すなわち、非飽和型のモデルに対応した曲線である。

3. 従来SRGMとの関係

3.1 従来モデルと関数

これまで提案された代表的なSRGMは表1のように分類できる。ここで、成長曲線が $t \rightarrow \infty$ のときに有限のある一定値になるモデルを飽和型と呼び、無限大に発散するモデルを非飽和型と呼ぶ。

3.2 飽和型モデルとの関係

表1の飽和型モデルのうち、指数形モデル、ゴンベルツ曲線、ロジスティック曲線および習熟S字形モデルは、統合モデルの特殊解となる。また、遅延S字形モデルおよび超指数形モデルでは近似解となる。

(1) 指数形モデル (表1 項番1) [1],[2],[4]

指数型モデルは、表1で示すように、

$$y = N \{1 - \exp(-bt)\} \quad (27)$$

と表せる。この式を t で微分すると、

$$\frac{dy}{dt} = N b \exp(-bt) \quad (28)$$

となる。これは(1)式において、 $\alpha = N b$, $\beta = b$, $\gamma = 1$, $\delta = 0$ とおいた微分方程式である。

表1 これまでに提案された主なモデル

No	成長曲線の特性		SRGM	
			モデル(曲線)名	関数形(*1)
1	指数形		指数形モデル	$y = N \{1 - \exp(-bt)\}$
2			超指数形モデル	$y = \sum_{i=1}^n N_i \{1 - \exp(-b_i t)\}$
3	S字形	飽和型	習熟S字形モデル	$y = N \frac{1 - \exp(-bt)}{1 + \psi \exp(-bt)}$
4			ロジスティック曲線	$y = \frac{N}{1 + \psi \exp(-bt)}$
5			ゴンベルツ曲線	$y = N p^{p^t}$
6			遅延S字形モデル	$y = N \{1 - (1+bt) \exp(-bt)\}$
7	疑似指数形	非飽和型	ワイブル過程モデル	$y = \lambda t^n$
8			対数ポアソン実行時間モデル	$y = \frac{1}{\theta} \ln(\lambda \theta t + 1)$

(*1) すべてのパラメータは正の値をとる。特にゴンベルツ曲線の p と q 、およびワイブル過程モデルの m は1より大きい値をとる。

(2) ゴンベルツ曲線 (表1 項番5)

ゴンベルツ曲線

$$y = N \cdot p^{p^t} \quad (29)$$

は、 $p^x = \exp\{x \ln(p)\}$ という関係を用いて、次式のような指数関数を用いた式に変形できる：

$$y = N \exp\{-k \exp(-bt)\} \quad (30)$$

ここで、 $k = -\ln(p) > 0$, $b = -\ln(q) > 0$ である。(30)式は(21)式で $y_0 = N \exp(-k) > 0$ とおいた場合に等しい。

(30)式を t で微分すると、

$$\frac{dy}{dt} = N k b \exp\{-k \exp(-bt)\} \exp(-bt) \quad (31)$$

となる。(30)式と(31)式より

$$\frac{dy}{dt} \cdot \frac{1}{y} = k b \exp(-bt) \quad (32)$$

が得られるが、これは(1)式で、 $\alpha = k b$, $\beta = b$, $\gamma = 0$, $\delta = 0$ のときの微分方程式である。

(3) ロジスティック曲線 (表1 項番4)

ロジスティック曲線の一般形およびそれを t で微分した式は、

$$y = \frac{N}{1 + \psi \exp(-bt)} \quad (33)$$

$$\frac{dy}{dt} = \frac{N b \psi \exp(-bt)}{\{1 + \psi \exp(-bt)\}^2} \quad (34)$$

となる。この2つの式から、

$$\frac{dy}{dt} \cdot \frac{1}{y^2} = \frac{b\psi}{N} \exp(-bt) \quad (35)$$

が得られるが、この式は(1)式で、 $\alpha = b\psi/N$ 、 $\beta = b$ 、 $\gamma = -1$ 、 $\delta = 0$ とおいた微分方程式である。なお、(10)式で $c = -1$ 、 $a = \psi < 0$ 、 $y_0 = N/(1 + \psi)$ の場合がロジスティック曲線である。

(4) 習熟S字形モデル[6]

習熟S字形モデルは、次のように変形できる：

$$y = N \frac{1 - \exp(-bt)}{1 + \psi \exp(-bt)}$$

$$= N \frac{\psi + 1}{\psi} \left(\frac{1}{1 + \psi \exp(-bt)} - \frac{1}{\psi + 1} \right) \quad (36)$$

(36)式は、ロジスティック曲線を表わす(33)式を $t=0$ のときに y が 0 になるように、 y 軸方向に並行移動した式と定数倍を除いて等しい。すなわち、(33)式において、 y を $y + N/\psi$ 、 N を $N(\psi + 1)/\psi$ と置き換えると(36)式が得られる。したがって、(3)と同様な計算で、習熟S字形モデルが、(1)式と同型の次の微分方程式を満たすことを示すことができる：

$$\frac{d\left(y + \frac{N}{\psi}\right)}{dt} \cdot \frac{1}{\left(y + \frac{N}{\psi}\right)^2} = \frac{b\psi^2}{N(\psi + 1)} \exp(-bt) \quad (37)$$

この式は、(1)式で、 $\alpha = b\psi^2/\{N(\psi + 1)\}$ 、 $\beta = b$ 、 $\gamma = -1$ 、 $\delta = N/\psi > 0$ とおいたものと等しい。

(5) 遅延S字形モデル (表1項番6) [5]

遅延S字形モデルは(1)式の解ではない。しかし、(10)式で表される(1)式の解のひとつが、よい近似を与える。近似の程度を議論するのに、遅延S字形モデルの関数で、 $N = 1$ 、 $b = 1$ とした

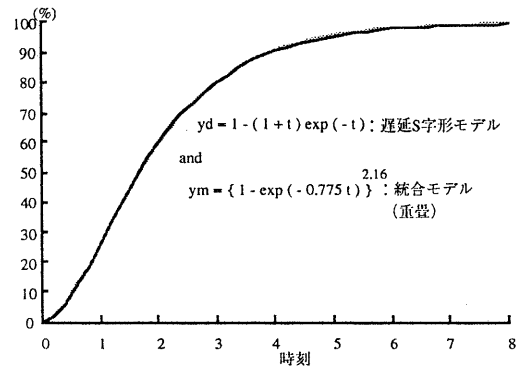
$$y = 1 - (1 + t) \exp(-t) \quad (38)$$

を対象としても一般性を失わない。(38)式に対しては(11)式で $b = 0.775$ 、 $c = 2.16$ とおいた次式がよい近似を与える：

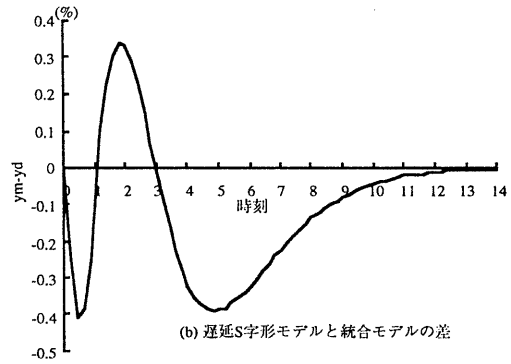
$$y = \{1 - \exp(-0.775t)\}^{2.16} \quad (39)$$

図1に、 t が $[0, 8]$ までの区間のそれぞれの曲線と、 t が $[0, 14]$ までの区間の2つの曲線の差のグラフを示す。2つの曲線の差は最大0.4%程度であり、時間経過と共にその差が小さくなる。

これより、統合モデルが遅延S字形モデルを良く近似することが分かる。



(a) 遅延S字形モデルと統合モデル



(b) 遅延S字形モデルと統合モデルの差

図1 統合モデルによる遅延S字形モデルの近似

(6) 超指数形モデル[6]

遅延S字形モデルと同様に、超指数形モデルも(1)式の解ではない。しかし、ここでも(11)式のひとつが、よい近似を与える。超指数形モデルは多くの組み合わせがあり、それらすべてについて近似解を論ずることは難しい。ここでは、異なる2つの指数曲線からなる次の関数について近似する。

$$y = 0.5 \{1 - \exp(-t)\} + 0.5 \{1 - \exp(-2t)\} \quad (40)$$

この関数に対しては、統合モデルの解のひとつである次の式がよい近似を与える。

$$y = \{1 - \exp(-1.24t)\}^{0.886} \quad (41)$$

図2に t が $[0, 5]$ までの区間のそれぞれの曲線と、 t が $[0, 6]$ までの区間の2つの曲線の差のグラフを示す。2つの曲線の差は最大1%程度である。

以上の結果をまとめて表2に示す。空欄はこれ

まで論議されていない関数型を持つSRGMの領域である。

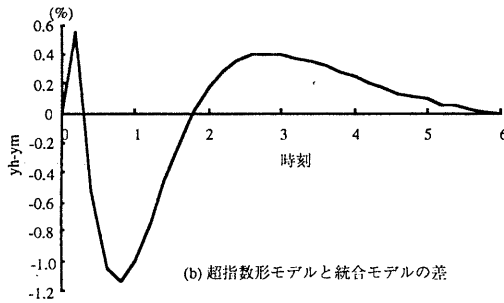
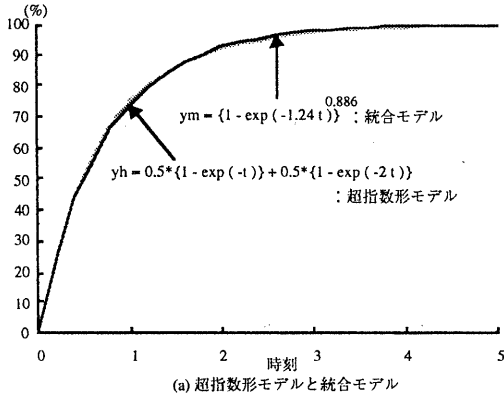


図2 統合モデルによる超指数形モデルの近似

表2 従来モデル(飽和型)と統合モデルの関係

c	a	a = 1	1 > a > 0	a → 0	a < 0
c = 0 (γ → ∞)		(y = 1)			
1 > c > 0 (1 < γ)		超指数形モデル		(y = 1)	(単調減少)
c = 1 (1 = γ)		指数形モデル			
2.16 > c > 1 (0.463 < γ < 1)					
c = 2.16 (γ = 0.463)		遅延S字形モデル			
c > 2.16 (0 < γ < 0.463)					
c → +∞ (γ = 0+)		(y = 0)		(a · c = k) ゴンベルツ 曲線	(発散)
c → -∞ (γ = 0-)		(発散)			(y = 0)
-1 > c (-1 < γ < 0)		(単調減少)			
c = -1 (γ = -1)		(y = 0で発散)		(y = 1)	ロジスティック 曲線 遅延S字形モデル
0 > c > -1 (γ < -1)		(y = 0で発散)			
c = 0 (γ → -∞)		(y = 1)			

(注) ハッチング部分は非単調増加で、SRGMの対象外である。

3. 3 非飽和型モデルとの関係

表1の非飽和型モデルのうちワイブル過程モデル(表1項番7)が(26)式の特解となる。(26)式で $y_0 = 0$ の場合は、次の式となる:

$$y = (\alpha \gamma)^{\frac{1}{\gamma}} \quad (42)$$

これは、ワイブル過程モデル

$$y = \lambda t^a \quad (43)$$

と同型である。

4. 統合モデルの評価

ここでは、リリース時点の抽出累計フォールト数を試験工程の途中でどの程度予測できるか(これをEnd point推定と呼ぶことにする)を、既存モデルと統合モデルで比較する。比較モデルとして、成長曲線特性の異なる指数型モデルと遅延S字型モデルを用いる。いずれもNHPPモデルに従うものとし、モデルのパラメータ推定には、確率論的に意味付けが明確である最尤推定法を用いる。

4. 1 理論データによる検証

実際のプロジェクトから収集したデータで検証を行う前に、指数型モデルと遅延S字型モデルの間に位置するモデルから得られる理論上のフォールトデータをそれぞれのモデルで推定した場合に、どの程度の推定誤差が生じるかを調べ、統合モデルの推定精度を確かめる。間に位置するモデルとしては、(1)式でパラメータ γ を0.75($c = 4/3$)としたものを用いる。

図3に、中間的なモデルから得られる曲線と、その曲線上の理論上のフォールトデータを統合モデル、指数形モデル、遅延S字形モデルで推定した結果を示す。当然のことながら統合モデルでの推定誤差はない。End point推定値は指数形モデルでは、大きくなりがちであり、遅延S字形モデルでは小さくなりがちである。

4. 2 実データによる検証

大規模ソフトウェア開発のフォールトデータである中川のデータ[11]を用いて分析した結果を以下に示す。

表3に中川データ、統合モデルによるEndpoint推定値、コルモゴロフスミルノフ検定(K-S検定)による適合度検定の結果、および各パラメータの推定値を示す。図4に統合モデル、指数形モデル、遅延S字形モデルによる推定曲線の適合

度を示す。ただし、推定時点は $t=42$ 、すなわち End point で推定したものである。図5にEnd point 推定誤差の変化を示す。特徴をつぎに要約する。
 (a)全体の推定誤差の平均は、統合モデルで16.4%、遅延S字形モデルでは17.0%、指数形モデルでは21.2%である。

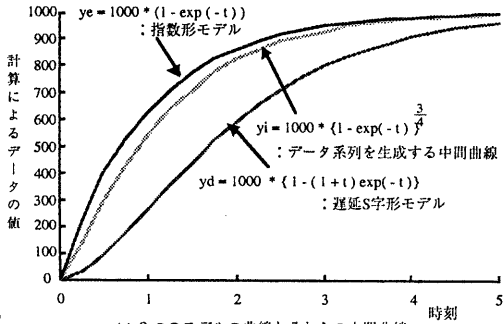


図3 理論的データに対する3つのモデルの推定誤差の比較
 (a) 2つのモデルの曲線とそれらの中間曲線
 (b) 中間曲線に対するEndpoint推定誤差の推移

表3 中川データに対する統合モデルのEnd point推定

t	累積フォールト数	Endpoint推定値	推定誤差 (%)	K-S 検定値 (*1)	パラメータの推定値		
					N	b	c
3	677	-	-	-	-	-	-
6	1769	-	-	-	-	-	-
9	2243	2536.5	-38.8	0.1857	2536.5	0.2682	2.462
12	2556	2742.3	-33.8	0.1571	2742.4	0.3061	2.735
15	2767	2892.6	-30.1	0.1422	2892.7	0.2682	2.462
18	2864	2933.1	-29.2	0.1370	2933.2	0.2554	2.360
21	3047	3132.6	-24.4	0.1283	3133.9	0.2025	1.961
24	3310	3479.0	-16.0	0.1198	3493.9	0.1402	1.537
27	3687	4062.1	-1.9	0.1099	4213.8	0.0836	1.207
30	3809	4074.4	-1.6	0.1065	4232.7	0.0825	1.201
33	3985	4207.4	1.6	0.1052	4453.8	0.0720	1.139
36	4074	4204.8	1.5	0.1027	4446.5	0.0724	1.143
39	4126	4181.0	0.9	0.1004	4400.6	0.0748	1.158
42	4142	4142.0	0	0.0980	4325.5	0.0793	1.189

(*1)すべての段階で推定曲線は適合 (信頼度95%)

(b) $t=9$ から24までは、遅延S字形モデルの推定誤差の絶対値の平均は27.8%、次いで統合モデルの平均が28.7%、指数型モデルの平均が31.1%である。ただし、指数形モデルは極端に悪い地

点とよい地点 ($t=9, 12$) を除くと、この区間の平均は15.9%となり他の2つのモデルよりよくなる。

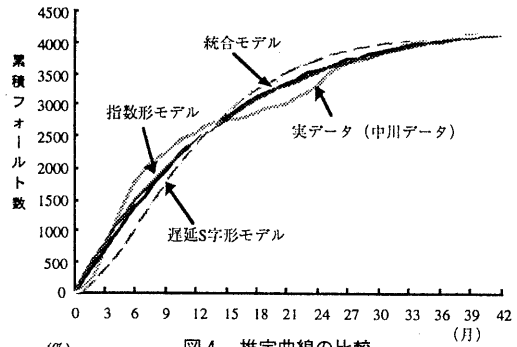


図4 推定曲線の比較

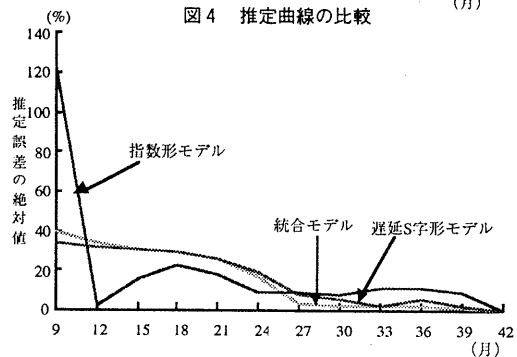


図5 中川データに対するEnd point推定誤差の推移比較

(c)統合モデルと遅延S字形モデルでは、 $t=24$ と27の間を境にして、推定誤差の絶対値の大きさが急激に減少する。これは、図4からわかるように、累積フォールト曲線が最終安定期に入った時点と一致する。ただし、指数形モデルではこのような顕著な変化は見られない。

(d) $t=27$ から39までは統合モデルの推定誤差の絶対値の平均は1.5%で最もよく、次いで遅延S字形モデルが平均3.7%である。指数形モデルは9.4%で他の2つのモデルに比較してかなり悪い。

指数形モデルによる推定精度が統合モデルによるものよりよい場合 ($t=18$) の実測データとそれぞれの推定曲線を図6に示す。明らかに統合モデルの方が実測データによく適合している。それにも拘らず統合モデルの推定精度の方が低いのは、統合モデルの適合度が高すぎて、次に来る急激な変化 (上昇) をそのまま推定誤差としてしまったことによる。このように統合モデルでは実際の成

長曲線に良く適合するため、フォールト検出率の変動が大きい開発の初期や中期では、End point 推定誤差が大きくなることがあるが、平均的に最も推定精度が高い。特にリリースが近付いて、累積フォールトデータが安定してくると、推定精度が高くなる。

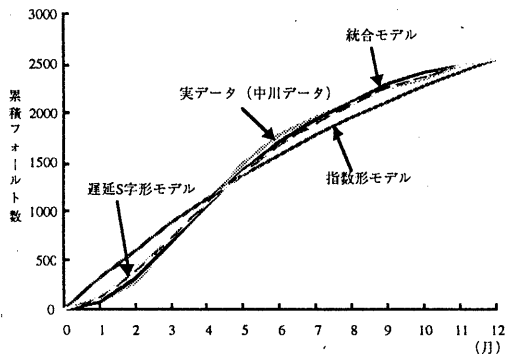


図6 3つのモデルの適合度比較

5. おわりに

本論文ではソフトウェア信頼度成長曲線に関する新しいモデル—統合モデルを提案した。このモデルは、

- (1) これまで提案された代表的なSRGMをカバーする、
 - (2) 従来のモデルの相対的位置関係を明確にする、
 - (3) 従来のモデルでは適合しにくかった形状の信頼度成長曲線にも適用できる、
- などの特徴を持つ。したがって、これまでのSRGMの利用者は、成長曲線特性に関わりなく統合モデルを使用するだけで残存フォールト数推定やリリース時までの検出フォールト数予測をより高い精度で行うことができるようになる。

参考文献

[1] Jelinski, Z. and Moranda, P.: *Software reliability reseach*, Statistical Computer Performance Evaluation, W. Freiberger, Ed. New York: Academic, pp. 465-484 (1972).

[2] Musa, J.D.: *A Theory of Software Reliability and Its Application*, *IEEE Trans. Software Eng.*, Vol. SE-1, No. 3, pp. 312-327 (1975).

[3] Littlewood, B.: *Theories of Software Reliability*

: *How Good Are They and How Can They Be Improved?*, *IEEE Trans. Software Eng.*, Vol. SE-6, No.5, pp. 489-500, (1980).

[4] Goel, A.L. and Okumoto, K.: *Time-Dependent Error-Detection Rate Model for Software Reliability and Other Performance Measures*, *IEEE Trans. Rel.*, Vol. R-28, No 3, pp. 206-211 (1979).

[5] Yamada, S., Ohba, M. and Osaki, S.: *S-S shaped Reliability Growth Modeling for Software Error Detection*, *IEEE Trans. Rel.*, Vol. R-32, No. 5, pp. 475-478 (1983).

[6] Ohba, M.: *Software reliability analysis models*, *IBM J. RES. DEVELOP.* Vol. 28, No. 4, pp. 428-443 (1984).

[7] Tohma, M., Tokunaga, K., Nagase, S. and Murata, Y.: *Structural Approach to the Estimation of the Number of Residual Software Faults Based on the Hyper-Geometric Distribution*, *IEEE Trans. Software Eng.*, Vol. SE-15, No.3, pp. 345-355 (1989).

[8] Kanoun, K., Martini, M.R.B. and Souza, J.M.: *A Method for Software Reliability Analysis and Prediction Application to the TROPICO-R Switching System*, *IEEE Trans. Software Eng.*, Vol. SE-17, No. 4, pp. 334-344 (1991).

[9] Karunanithi, N., Whitley, D. and Malaiya, Y.K.: *Prediction of Software Reliability Using Connectionist Models*, *IEEE Trans. Software Eng.*, Vol. SE-18, No. 7, pp. 563-574 (1992).

[10] 山田茂: *ゴンベルツ曲線を用いた確率的ソフトウェア信頼度成長モデル*, *情処論文誌*, Vol. 33, No. 7, pp. 964-969 (1992)

[11] Nakagawa, Y. and Hanata, S.: *An error complexity model for software reliability measurement*, *Proceedings of 11th ICSE*, pp. 230-236, 1989.

[12] Goel, A.L.: *Software Reliability Models: Assumptions, Limitations, and Applicability*, *IEEE Trans. Software Eng.*, Vol. SE-11, No. 12, pp. 1411-1423 (1985).