ソフトウェア工学 64－1
（１９８９．２．２）

フィールドにおけるソフトウェア信頼度に関する現象とそのモデル

松尾谷　徹
日本電気㈱

本報告ではフィールドにおけるソフトウェアの信頼度に関して観測される３つの現象を示し、その現象を説明するモデルを提案する。　３つの現象とは『フィールドにおける信頼度の成長』、『量販ソフトウェアにおける出荷数と障害件数』、及び『複雑な機能をもつ量販ソフトウェアの障害件数』に関するものである。

ソフトウェアの障害発生メカニズムをソフトウェアの持つ機能の集合に対し、その利用者がランダム探索を行うとしたモデルを提案し、３つの現象を説明する。その結果、フィールドにおいて発生する障害はソフトウェアに残留する欠陥数よりも利用者がどの程度ソフトウェアを使うかを測る『使用の程度』が大きな要因であることが分かった。

A model and some phenomena of software reliability within user sites

Toru Matsuodani

NEC Corp.
Shibaura 4-14-22, Minato-ku,
Tokyo, 108, Japan

This paper presents three phenomena of software reliabilty within user sits and proposes a model which describes these phenomena. Those phenomena are "growth of reliability", "number of shipments and number of failures" and "complicated software's number of failures".

Software failures developments mechanizum is explained by a model where failures are randomly searched by users within a set of software functions through three phenomena. As a result, "degree of use" which measures user's use of the software is more important factor than remaining errors in the software to explain number of failures within user sits.

1．はじめに

ソフトウェアの信頼性に関する研究は、主にソフトウェア開発過程において欠陥除去を行うテスト工程を対象として進められてきた。　そこでは発見された累積欠陥数とテストに費やされたリソース（テスト項目やカレンダー時間）の関係に着目し、欠陥の発見過程をソフトウェアの信頼度成長モデルとして記述している。　信頼度成長曲線として指数型の関数からLogistic，遅延S字型、さらにそれらを変形した関数等その数は約４０にも達している。1)2)3)

ソフトウェアの信頼度が重要な役割を果たすのは、そのソフトウェアが実際に運用される過程である。にもかかわらずこの分野の研究は信頼度の計測、信頼度に関する諸現象の整理すら不足しているのが現状である。

本報告はソフトウェア開発過程における試験の信頼度成長に関するものではなく、ソフトウェア運用段階における信頼度について観測を基にモデル化を試みた。　運用段階における信頼度は利用者のソフトウェア『使用の程度』とソフトウェアの欠陥数により決まるとし、ここでは『使用の程度』を主に考察した。

2．信頼度とモデルの考え方

ソフトウェア品質尺度としての信頼性4)5)とは別に、故障率の補集合として知られてい信頼度の定義は『規定の条件のもとで意図する期間中、規定の機能を遂行する確率』とされている。　ソフトウェアは明らかに故障による障害を起こすことは無いが、欠陥によって障害を引き起こす。

ソフトウェアを運用した場合どの位の期間でどの位の障害件数に遭遇するかを確率として示す信頼度の定義が必要と考える。（ソフトウェア品質尺度としての信頼性と区別するため信頼度と呼ぶ）

ソフトウェアの信頼度についても『規定の条件』、『意図する期間』、『規定の機能』を定義する必要がある。ソフトウェアにおける『規定の機能を遂行する』ことは、定義されたソフトウェアの機能が障害なく動作するとする。

障害とはソフトウェアの欠陥もしくは機能そのものの欠落により定義された機能が遂行されない現象と定義し、ここでは欠落も欠陥の一種として欠陥だけを取り扱う。

『規定の条件』、『意図する期間』は本来故障に対するストレスを規定している。　ソフトウェアの障害は欠陥の探索過程であるから、このストレスに対応するものは探索量である。（どんなに欠陥の多いソフトウェアでも使わなければ障害は発生しない）　探索量として利用者における『使われ方』や『期間』を定義することが必要となる。

この考え方を次に示すモデルで定義する。

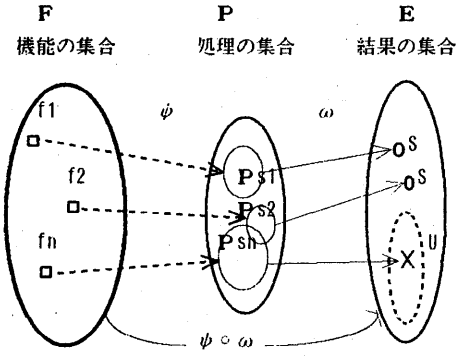


図1　ソフトウェアの機能と処理の対応モデル

ソフトウェアの持つ機能の集合を**F**とし、その要素をf1,f2,・・・・,fn とする。同じくその機能に対応して作られたソフトウェアの処理実体の集合を**P**とし、その要素をp1,p2,・・・・,pm とする。

$$\mathbf{F} = \{f1, f2, \cdots\cdots, fn\} \qquad (1)$$

$$\mathbf{P} = \{p1, p2, \cdots\cdots, pm\} \qquad (2)$$

機能と処理の対応は一対一にはならない、簡単な例を考えると共用されるサブルーチンなどは複数の機能と対応することがある。　そこで機能との対応を**P**の部分集合**Psi**とし**F**と対応すると定義する。

$$\mathbf{Psi} \subset \mathbf{P}$$

$$\mathbf{Psi} = \{pk ; pk \in \mathbf{P}\} \qquad (3)$$

$$1 \leqq i \leqq n$$

そして、機能fiに対する処理の部分集合**Psi**との関係を写像ψとする。

$$\psi : \mathbf{F} \rightarrow \mathbf{P} \quad fi \rightarrow \mathbf{Psi} \qquad (4)$$

処理の部分集合**Psi**が正しく動作するか、又は障害を含むかを示すために結果の集合**E**を考える。

$$\mathbf{E} = \{e1, e2, \cdots\cdots, en\} \qquad (5)$$

但しeiはs（成功・正常）｜u（失敗・障害）

$1 \leqq i \leqq n$ においてu の数をｐ個とする

処理の部分集合**Psi**と結果**E**との関係を写像ω、機能→処理→結果の関係を写像ψとωの合成とする。

$$\omega : \mathbf{P} \rightarrow \mathbf{E} \qquad (6)$$

$$\psi \circ \omega : \mathbf{F} \rightarrow \mathbf{E} \qquad (7)$$

式(1) から式(7) までで定義したモデルにおいて、利用者がソフトウェアを使う行為を、機能集合**F**の中から部分集合を取り出すと考え、その部分集合を**Fu** とする。

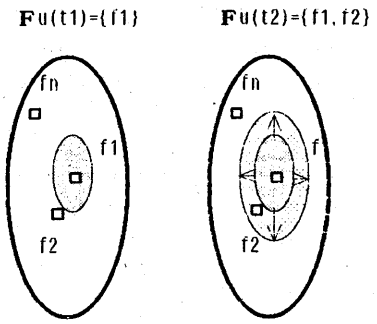


図2　機能の部分集合Fu(t)の成長

$$\mathbf{Fu} \subset \mathbf{F}$$
$$\mathbf{Fu} = \{fk; fk \in \mathbf{F}\} \qquad (8)$$

このモデルにおいて利用者が障害に遭遇する現象は、取り出した機能の部分集合Fu に対応する処理の部分集合Pu に欠陥が含まれる事象として表現する。

また利用者が使用する機能の部分集合Fu は時間と共に成長すると考えられる、時間t における部分集合をFu(t)とする。　利用者における障害の時間変化（信頼度成長）はFu(t)のPへの写像　ψ（Fu(t)）によって決まるPの部分集合Pu(t)に含まれる欠陥数により決まる。

このことは、便宜的に結果の集合Eにに対する写像$\psi \circ \omega$で定義される部分集合をEu(t)により、遭遇する累積件数をBu(t)を現すことができる。

$$\mathbf{Eu}(t) = \psi \circ \omega\ (\mathbf{Fu}(t))$$
$$Bu(t) := \mathbf{Eu}(t)\text{の要素ejのうちej=0の総数} \qquad (9)$$

故障率に対応する障害率F(t) を考える、これは時刻tまでに選択された機能単位あたりの障害の数と定義する。

$$F(t) = \frac{Bu(t)}{\mathbf{Fu}(t)} \quad \text{注1} \qquad (10)$$

信頼度R(t) は、

$$R(t) = 1 - F(t) = 1 - \frac{Bu(t)}{\mathbf{Fu}(t)} \qquad (11)$$

t=∞について考える、仮説としてFu(t)がt=∞においてFと等しくなるならば、次の式が成立する。

$$F(t=\infty) = \frac{\text{障害のある機能の数}}{\text{全機能の数}} = \frac{p}{n} \qquad (12)$$

$$R(t=\infty) = 1 - F(t=\infty) = \frac{n-p}{n} \qquad (13)$$

式(12)は欠陥の機能に対する率を示しており、その率をμとする。　**処理集合の中で欠陥の分布が一様**であると仮定すれば、式(12)，(13)は次の定数となる。

$$F(t) = \frac{p}{n} = \mu \qquad (12)'$$

$$R(t) = 1 - F(t) = \frac{n-p}{n} = 1 - \mu \qquad (13)'$$

このように定義すると、ソフトウェアの信頼度自身は時間の関数では無く一定の値となる。　利用者が遭遇する障害の累積件数Bu(t)は利用者が選択した機能の部分集合Fu(t)の大きさによって決まることになる。

$$Bu(t) = \mu \cdot \mathbf{Fu}(t) \qquad (14)$$

ハードの故障率はt=0 で0 でありt=∞で1になる、その時間関数の形（たとえばバスタブ曲線）が重要な問題である。　それに対しソフトウェアの障害率は一定である、もちろんμの大小が影響を与えるが、それ以上にFu(t)，つまり使用する機能の大きさ（使用の程度）が障害件数に大きく影響する。

Fu(t)は明らかにt=0 において0 である、よってBu(t)もやはりt=0 において0 である。　またFu(t)は単調増加でありその微係数をΔFu(t)とすると、

$$\Delta\mathbf{Fu}(t) = 0 \quad \text{ならば} \quad \Delta Bu(t) = 0 \qquad (15)$$

式(15)は興味深い結果である、すなわちμのいかんにかかわらず、Fu(t)の成長が飽和すれば障害件数も飽和することを示している。このモデルを前提とし、以下に観測された現象とそれを説明するサブモデルを示す。

3．保守工程におけるソフトウェアの改版・修正

まずフィールドにおけるソフトウェア保守工程とその用語を定義する。

①機能追加・version-up

ソフトウェアの環境変化に適応するため、あるいは競合製品に対応して製品品質を向上するため等の目的でソフトウェアに新しい機能を追加する。　その時出荷するソフトウェアをversion-upと呼ぶ。

②予防保守・revison-up

ソフトウェアの欠陥、不具合の修正や、性能、効率の改善等、外部機能仕様に大きな変化を伴わない改版。　この時の出荷をrevison-upと呼ぶ、便宜上最初に出荷する場合はversion-up（version-1）とする。

③事後保守・個別対処

障害の発生した該当利用者に対して、障害の復旧、原因の除去、回避手段の提供等を行う。　ここで発見された欠陥は②の予防保守により他の利用者に対しても処置が取られる。

注1：Fu(t)は集合を表現する記号であるが、ここでは便宜上その要素の数を表す、以下同様

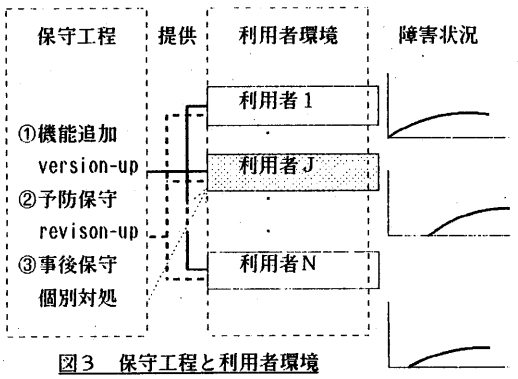


図３　保守工程と利用者環境

利用者環境には個別注文ソフトウェアのように一つの利用者しか存在しない場合もある、この場合は複数の利用者に継続的に保守を行う事を前提とした図３の定義において、Ｊ番目の利用者が他の利用者で発見された欠陥に対する予防保守を受けない状態、つまり同一version/revison を使用した状態として取り扱うことができる。

４．現象Ｉ　信頼度成長（ランダム探索モデル）

図３の一つの利用者Ｊについて、障害の発生状況を観測する。　表１に規模約200ks のソフトウェアに対する観測値を示す。

表１　利用者における月別障害件数

月数 / 利用者	1	2	3	4	5	6	7	8	9	10	11	12	13	期間
U1	6	2	1	0	1	0	0	0	0	0	0	0	0	15
U2	4	0	0	3	0	0	1	0	0	0	0	0	0	14
U3	1	1	1	0	0	0	0	0	0	0	0	1	0	21
U4	7	2	0	2	0	0	0	0	0	0	0	0	0	16

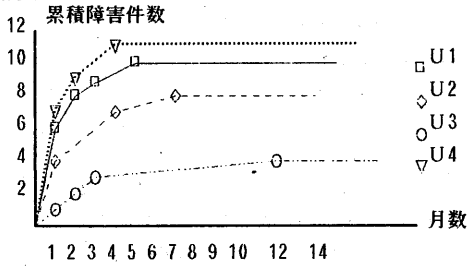


図４　表１の累積値

表１における利用者U1, U2, U3, U4 それぞれの観測月数は15, 14, 21, 16 であり表に示した13ケ月以降の障害件数は０であった。（それ以降はversion-upが行われた）

注２：正確には$\mathbf{Fu}(t)$に含まれる要素の数。

図４に表１の４つの観測値に対する累積値のグラフを示す。　図４から明らかなように障害件数は時間と共に減少（累積値は飽和する）しており、その振舞いは指数型に見える。

指数型に適合している事を確かめるために指数型の関数式(16)を変形し両辺にlog をとった式(17)を考える。

$$Y=\alpha\ (1-e^{-\gamma t}) \qquad (16)$$

$$-\gamma t = \log\left(1-\frac{Y}{\alpha}\right) \qquad (17)$$

式(17)のＹを月数、αを観測期間における全累積障害件数として表１の観測値をプロットしたものを図５に示す。

また式(17)から各観測値の平均値としてγを求め、求めたγ、αを使って累積障害件数を予測したものを表２に示す。　これらのことをまとめると、

現象Ｉ.

利用者において発生する故障件数の累積値Bu(t)は時間と共に飽和し、十分長い期間後に発見された累積障害件数をαとすると、Bu(t)は次の式に当てはめられる。

$$Bu(t)=\alpha\ (1-e^{-\gamma t}) \qquad (18)$$

式(14)と現象Ｉの式(18)から$\mathbf{Fu}(t)$を求める。　注２

$$\mu\cdot\mathbf{Fu}(t)=\alpha\ (1-e^{-\gamma t}) \qquad (19)$$

αがそのままμに対応していないので別の係数κを用いて式(19)を変形する。

$\alpha=\mu\cdot\kappa$とすると

$$\mathbf{Fu}(t)=\kappa\ (1-e^{-\gamma t}) \qquad (19)'$$

よって式(14)は

$$Bu(t)=\mu\cdot\kappa\ (1-e^{-\gamma t}) \qquad (14)'$$

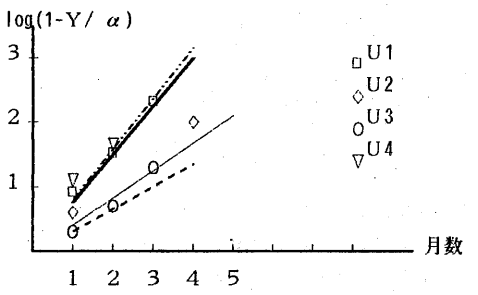


図５　故障件数の時間変化

表２　モデルによる予測と残差

利用者名	α	γ	月別の予測値（上）　残差（下）							
			1	2	3	4	5	6	7	7
U1	10	0.77	5.4	7.8	9.0	9.5	9.8	9.9	10	10
			-0.6	-0.2	0	.5	-.2	-.1	0	0
U2	8	0.43	2.7	4.5	5.8	6.5	7.0	7.3	7.6	7.7
			-1.2	.6	1.7	-0.5	0	.3	-.4	-.3
U3	4	0.34	1.2	2	2.6	3	3.2	3.5	3.6	3.7
			.2	0	-0.4	0	.2	-.5	-.4	-.3
U4	11	0.81	6.1	8.8	10	10.6	10.8	10.9	11	11
			-0.9	-0.2	1	-.4	-.2	-.1	0	0

式(19)' のμは欠陥の密度を、またκは機能集合における部分集合$\mathbf{F}u(t)$の大きさをそれぞれ示す係数である。

この式(19)' を説明するために、図６に示す探索モデルを考える。　このモデルは探索の対象となる空間の大きさを探索空間Ａとし、ある瞬間における探索幅を有効探索幅Ｗとする。　探索空間内には目標物が一様に分布しているとし、探索速度Ｖでｔ時間のランダム探索を行うと、目標物の発見確率Ｐ（ｔ）は式(20)で与えられる。9)

$$P(t)=1-e^{-WVt/A} \qquad (20)$$

式(20)を複数の目標物α個に対する探索として考えると式(18)になる。　そうすると探索モデルにおける探索の要素と係数の関係は次式で与えられる。

$$\alpha=\mu\cdot\kappa=\mu\cdot A \qquad \kappa=A$$
$$\gamma=\frac{W\cdot V}{A} \qquad (21)$$

κは探索空間の大きさを現しておりこれは機能集合$\mathbf{F}$の部分集合$\mathbf{F}u(t)$の大きさに対応する。γは時間当たりのκに対する探索幅の比に対応している。　表１に示したデータから利用者U1, U2, U3, U4 におけるそれぞれの値を添え字により表すと次のようになる。

$$\mu\cdot A1=\alpha1=10 \qquad \mu\cdot A2=\alpha2=8$$
$$\mu\cdot A3=\alpha3=4 \qquad \mu\cdot A4=\alpha4=11 \qquad (22)$$
$$\mu\cdot A1:\mu\cdot A2:\mu\cdot A3:\mu\cdot A4=A1:A2:A3:A4=10:8:4:11 \qquad (23)$$

Ｗ・Ｖ／ＡをＱとすると

$$Q1=\gamma1=0.77 \qquad Q2=\gamma2=0.43$$
$$Q3=\gamma3=0.34 \qquad Q4=\gamma4=0.81 \qquad (24)$$

式(23)の比はU1, U2, U3, U4 の利用者における利用した機能の大きさ$\mathbf{F}u(t)$の比に相当する。　この事例では２から３倍の差がある事を示している。　式(24)は単位時間（ここでは１ケ月）における探索幅を探索空間Ａとの比として示したものであり、例えばU1においては最初の１ケ月で全利用機能の77% を使用したと読み取ることが出来る。

なお興味深いことにこの二つの係数間にはなんらかの関係がある、それを図７に示す。

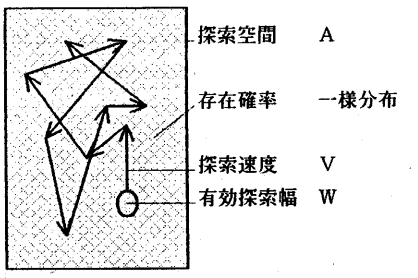


図６　探索モデルとその要素

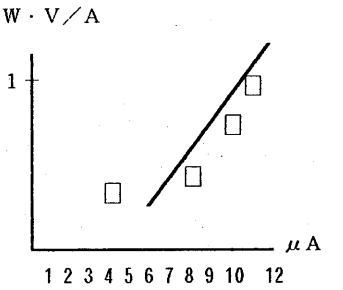


図７　探索空間の大きさと探索量の関係

５．現象Ⅱ　量販ソフトウェア出荷数と障害件数

量販されるソフトウェア製品は同じversion を多数の利用者が使うことになる。　この場合において障害件数はどの様な振舞いをするかについて観測を行い考察する。

同じversion の利用者をu1, u2, u3, ‥‥, uN とすとそれぞれの利用者で発生する障害の累積件数は式(14)によりＢu1(t), Bu2(t), ‥‥, BuN(t) となる。

つまりソフトウェアの出荷数をＮ本とした場合である。このときの全ての障害件数をBall(t)　とすると、

$$Ball(t)=Bu1(t)+BU2(t)+\cdots+BuN(t)=\sum_{J=1}^{N}BuJ(t) \qquad (25)$$

ところがソフトウェアの出荷は同時に行われないので、式(25)における時間はそれぞれの利用者によって異なることになる。　しかし式(24)で示したように実測されたγの実測値は大きく、月数ｔに対し3-4 ケ月程度で式(19)は飽和するとしてさしつかえない。

よって式(25)は観測期間を長くとった場合ｔの関数ではなくなり次式として扱える。

$$Ball=BU1+BU2+\cdots+BUN=\sum_{j=1}^{N}BUj \qquad (26)$$
$$BUj=\alpha j=\mu\cdot\kappa j$$

表３に３つの異なる製品についての障害件数を観測した値を示す、この表は障害件数の発生を利用者の度数として示している。　製品Ｘ１は観測期間中に障害件数=0の利用者数が154,障害件数=1が55と読み取る。　表３からもこの度数分布がポアソン分布に近いことが推測される、図8,9,10に度数分布とポアソン分布から求めた計算値を示す。そしてこの現象は次のように記述できる。

現象Ⅱ

同一ソフトウェアを複数の利用者が使う場合、それぞれの利用者で発生する障害件数の度数分布はポアソン分布に従う。

この時、ポアソン分布の係数λは利用者総数(出荷数)をN、比較的長い期間に発生した障害総件数をBall とすると次式で与えられる。

$$\lambda=\frac{Ball}{N} \quad または \quad Ball=\lambda\cdot N \qquad (27)$$

現象Ⅱは直感とは異なる結果である、ソフトウェアの障害件数はその製品の質(残留欠陥数) により支配されると考えられるが、実際の観測値はそうではなくユーザの数(製品の出荷数)に比例するのである。

この事例の場合、該当製品は十分テストされた後に出荷されている、そうした条件下での現象である。

この現象をモデルの面から検討するために、図11に示す機能の広がりにおけるFu の重複部分と処女地の関係に着目する。　機能集合Fの上で部分集合Fujがj=1 から順に増加して行く事を考える、図11のJ=1 から順にJ=2, J=3,・・・・、J=n へと進んで行くと、使われた機能の部分集合はFu1∪Fu2∪Fu3∪・・・・∪Funとして膨らんで行く。

表３　障害件数の度数表

製品種	利用者の度数						
	Tu=0	Tu=1	Tu=2	Tu=3	Tu>3	利用計	λ
Ｘ１	154	55	10	3	6	228	.47
Ｘ２	130	42	8	1	3	184	.32
Ｘ３	26	11	2	1	3	43	.70

製品の規模　Ｘ１　350ks　　Ｘ２　93ks
　　Ｘ３　230ks
観測期間　Ｘ１　３年　　Ｘ２　３．５年
　　Ｘ３　３年
機種　　汎用・中大型機

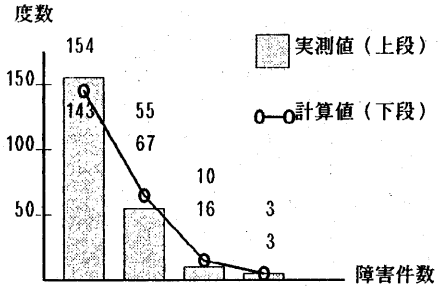


図８　障害件数の度数分布　製品＝Ｘ１

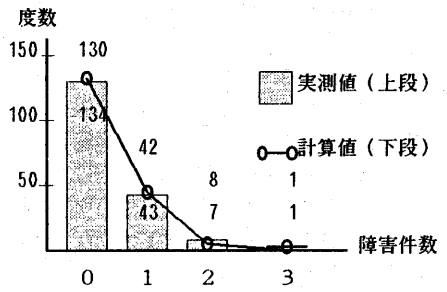


図９　障害件数の度数分布　製品＝Ｘ２

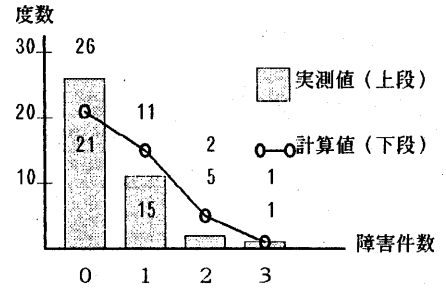


図10　障害件数の度数分布　製品＝Ｘ３

n 番目の利用者が使用した機能の集合FUnを考えると、そのほとんどはn-1 番目までの機能集合に含まれるが、図11の網かけで示す重複しない部分が考えられる。この部分がn 番目の利用者における処女地であり、新たな障害はこの部分から発生すると考える。　J=n における機能集合の中の処女地をFbnとすると、

$$Fbn=Fun-(Fu1\cup Fu2\cup Fu3\cup\cdots\cup Fun-1) \qquad (28)$$

そうするとn 番目の利用者における障害件数BUnは、

$$BUn=\mu\cdot Fbn \qquad (29)$$

式(29)において、μが一定でFBnの大きさの変動があまり無いとすれば、μ・FbnにEbnに含まれる失敗（障害）の数はベルヌーイの試みとなり、ポアソン近似により表現することが出来る。　よって式(27)が成立すると考える。

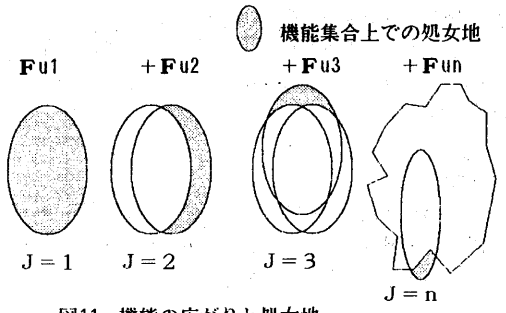


図11　機能の広がりと処女地

式(27)のλを使って、個々の利用者における障害件数を式(18),(19) から求められる。

$$Bu(t)=\mu\cdot FUb(t)=\lambda\cdot(1-e^{-\gamma t}) \qquad (30)$$

$\lambda=\mu\cdot\kappa b$ とすると

$$Fub(t)=\kappa b\cdot(1-e^{-\gamma t}) \qquad 注3$$

5．現象Ⅲ　複雑な機能を持つソフトウェアの場合

表３をもう少し詳しく見るとtu>3の度数がポアソン分布で考えられる度数より多いことが読み取れる。特にＸ１の場合などtu>3を除いてλを求め、それからtu>3が発生する確率を求めると。

$$P(k>3)=(1-(P0+P1+P2+P3))=0.00119 \qquad (31)$$

この値は実際に観測されている６件(0.03)と隔たりがある。そこで多くの製品について障害状況を調査したところ表３とは少し異なる分布を持つ例があることがわかった。

その観測値を表４に、そして度数分布を図12に示す。事例の製品Ｙの規模は150Ks であるが、言語仕様を持ちマクロ処理を行う製品であるため、他と比較し複雑な構造を持っている。

表４の値は現象Ⅱのように一つのポアソン分布では説明出来ない、説明出来ない部分とは表４の度数８から１３にかけての値と、４から６にかけての値がそれに該当する。

この現象は、現象面からみると３つの異なるポアソン分布の合成として捕らえることが出来る。　それぞれを分布a,分布b,分布c その数をNa,Nb,Nc その平均値をλa,λb,λc とすると、

$$Ball=\lambda a\times Na+\lambda b\times Nb+\lambda b\times Nb+\cdots\cdots \qquad (32)$$
$$N=Na+Nb+Nc$$

製品Ｙの場合は、

$$Ball=0.54\times 360+4.7\times 7+10.6\times 7 \qquad (33)$$

これをまとめると、

注３：正確には連続値と離散値の関係を考慮する必要あり

表４　障害件数の度数分布２（製品Ｙ）

ﾚﾝｼﾞ	0	1	2	3	4	5	6	7	8	9	10	11	12	13	>13
度数	210	117	22	10	4	3	1	0	1	1	1	2	1	1	0

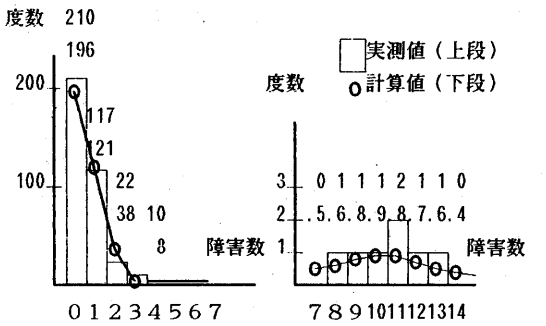


図１１　障害件数の度数分布（製品Ｙ）

表５　三つ度数分布　（製品Ｙ）

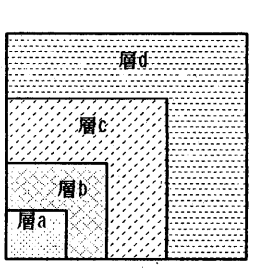
	利用者数		Σ障害件数		λ
分布a	360	96%	195	64%	0.54
分布b	7	2%	33	11%	4.7
分布c	7	2%	74	25%	10.6
合計	374	100%	302	100%	---

現象Ⅲ

複雑な機能を持つソフトウェアの場合、障害件数の度数分布は必ずしも一つのポアソン分布だけで構成されるとは限らず、複数の分布に層別される場合がある。　全障害件数Ball は式(32)で与えられる。

この現象を説明するために図６に示した探索空間を幾つかの層に分けた層別モデルを考え図13に示す。　式(23)からも利用者間におけるκに差が存在することがわかる、この差を幾つかに区分したものが図13の層に相当する。

このモデルにおいて、利用者は該当する層の中をランダム探索し、そして層の広さは層a<層b<層････とする。



全探索空間＝F

層a における利用者はこの層の中でランダム探索を行う。

図１３　探索空間の層別モデル

そうすると、この探索空間・層別モデルが現象Ⅲを説明することが出来る。　式(33)から$\lambda a<\lambda b<\lambda c$と成っているが、この値がそのまま層別された空間の広さに対応しているかどうかは分からない、つまりそれぞれの層において式(30)を考えたとき、

$$\mu a \cdot \mathbf{F}Ub(t) = \lambda a \cdot (1 - e^{-\gamma t}) = \mu a \cdot \kappa a \ (1 - e^{-\gamma t}) \quad (34)$$

$$\lambda a = \mu a \cdot \kappa a \quad (35)$$

式(35)において別の層、たとえば層b と比較したときμa とμb が同じになるとは考えにくい。何故ならば層bの方が高度で複雑な機能に対応していると考えられるからである。

現象Ⅲを説明する層別モデルが正しいとするならば、ここで考えている機能集合$\mathbf{F}$の大きさは非常に大きな規模と考えられる。ソフトウェアの障害件数が見かけ上飽和するのは利用者側の利用する機能の部分集合が飽和するのが原因と考えられる。少なくとも一般に機能仕様書と呼ばれる設計段階で明示的に定義される機能数や、検査工程において原因－結果グラフ技法を用いて機能間の組み合わせまで網羅した機能数（多くともks当たり200 から400 程度）よりも大きな数と思われる。

これは推測であるが、ここで考えた機能集合はソフトウェアの内部状態、たとえば内部変数とその組み合わせに比例するのではないかと考える。　この考えはハードの論理回路試験における順序回路の役割を基にしている。

6．むすび

ソフトウェアのフィールドにおける障害件数に関する３つの現象を報告し、その現象を信頼度と利用者の『使用の程度』によって説明することを試みた。　そして障害件数を決定するのは利用者の『使用の程度』が大きな要因であり、それは探索行為としてとらえることが出来る事を示した。　従来よりソフトウェアの信頼度の問題はソフトウェアに含まれる欠陥除去を中心に進んできたが、ここで提案することは、利用者の使う機能の部分集合$\mathbf{F}u(t)$に関する諸特性が大きな要素となっていることである。

今後の課題として次の項目がある。

Ａ．応用面

①運用中のソフトウェアに対する障害件数の予測

個別システム、量販システム、複雑な機能を持つ量販システムを区別し、μとκの推測から可能と考えれれる。

②出荷製品の保守量の見積もり

ソフトウェア供給者側から見ると、出荷後の量販ソフトウェアの保守量は予防保守によって決まる。最適な予防保守を行う。

③計測問題

出荷後の製品の信頼度を、製品相互間で比較できる計測単位、計測方法としてκ、μ、Ｎと障害件数を用いて使われ方の量を考慮した比較が可能となる。

Ｂ．研究面

④検査工程における信頼度成長との関係

妥当な検査量、検査工程における信頼度成長、それらと実際に製品が出荷された後のμ、κとの関係。

⑤検査工程でのモデル応用

こで提案した機能集合における探索行為によるモデルはソフトウェアのテストにおける信頼度成長の飽和問題を説明出来る。特に探索上の効率を大きく左右するランダム探索と平行探索の問題への応用。

⑤利用者の利用量に対応した設計・検証の方法論

同じμであっても、障害件数の少ないソフトウェア設計、検証方法‥‥つまりκの利用方法。

参考文献

１）山内　慎二他”電子情報通信学会信頼性小委員会Ａ活動報告”　信学会Ｒ研究会R88-5 1988年 5月

２）Ａ A.Abdel-Ghaly"Evaluation of Competing Software Reliability Predictions"　IEEE VOL.SE-12 1986-N09

３）山田　茂”ｿﾌﾄｳｪｱの信頼性評価法”　ＳＲＣ　1985年11月

４）管　忠義編”ｿﾌﾄｳｪｱ開発の実際　標準とその活用”　日本規格協会　1988年 2月

５）東　基衛”品質評価尺度と計測技術”　標準化と品質管理、Vol.40,No.8,PP.63-75　1987年 8月

６）松尾谷　”ｿﾌﾄｳｪｱ誤りの統計的解析”　信学会Ｒ研究会R83-11　1983年5 月

７）松尾谷　”予防保守のモデル化”　第36回情処学会全国大会　5L-3　1988年3 月

８）松尾谷　”層別モデルによるﾌｨｰﾙﾄﾞ信頼性の考察”　第38回情処学会全国大会　1989年3 月

９）多田和夫”探索理論”　日科技連　ＯＲライブラリー　1973年8 月