Chapter - II

REVIEW OF SOME SOFTWARE RELIABILITY MODELS

§ 2.1 Introduction

Many models for software reliability exist in the literature. In this chapter we present some of the more popular software reliability models. These models have been broadly categorized in three classes- the first category, where the failure rate is constant and is proportional to the residual number of bugs. Second category is of Bayesian nature, where the failure rate is a random variable with a specified prior distribution and third category relates the occurrence of failures as a Stochastic Process.

According to Musa, Iannino and Okumoto (1987) the first study of software reliability appears to have been conducted by Hudson (1967). He viewed software development as a birth and death process (a type of Markov process), in which it was assumed that the rate of detection of fault was proportional to the number of faults remaining and a positive power of the time. In other words, the rate of detection was assumed to increase with time.

However the model of Jelinski and Moranda (1972), popularly known as JM model, was the first software reliability model to be used and has formed the basis for many models developed after. In this model, it is assumed that the rate of failure is proportional to the residual number of bugs. In this model the failure rate changes at each fault correction by a constant amount, but is constant between correction. Details of the model is reported in section (2.2). Further Goel and Okumoto (1978) have made an attempt to improve upon the JM model by altering its assumption that a perfect fix of a bug always occurs. In section (2.3), we represent this model, in which it is assumed that the failure rate is a porbabilistic function instead of a linear function. Keeping all the other assumptions of JM model as it is and assuming that the failure rate is proportional to the elapsed time since last failure, Schink and Wolverton (1978) gives a different model from those of JM (1972) and Goel and Okumoto (1978) model. This is reported in section (2.4). In section (2.5), we present the De-eutrophication model by Moranda (1975), in which the failure rate is assumed constant between failures, but it is decreasing geometrically.

A Bayesian approach to software reliability is presented in sections (2.6), (2.7) and (2.8). In section (2.6) we give a Bayesian Reliability Growth model by Littlewood and Verrall (1973). In this model software reliability is viewed as a measure of strength of belief that a program will operate successfully.

The failure rate is assumed to be a random variable. Mazzuchi and Soyer (1988) have given an extension to the model of Littlewood and Verrall (1973), which is introduced in section (2.7) in which an additional assumption of inter occurrence time to be exponential random variable with scale failure rate between failures, is made. Again Littlewood and Verrall (1973) model is extended by assuming failure rate as a function of the residual number of bugs. This model is reported in section (2.8).

Goel and Okumoto (1979), described failure detection as a non-homogeneous Poisson process (NHPP), with an exponential decaying rate function. It is assumed that, the number of failures in the software is a random variable, which is fixed in JM model and the time between failures are assumed dependent, which is independent in JM model. This is given in section (2.9). In section (2.10), we give a Logarithmic Poisson model by Musa and Okumoto (1984); which is based on a NHPP with an intensity function that decreases exponentially with failures experienced.

Now we give a table which shows various software reliability models in accordance with their category. In the dissertation we limit our discussion to only *Type I-i* and *Type II* software reliability models. In the subsequent section these models occur in a sequence as given in the table (see table 2.1).

Table 2.1

Sr. No.	Category	Name of the model & year	Abbrivation
		Type I-1	
1	I	Jelinski & Moranda (1972)	JM
2	I	Imperfect Debugging model by Goel & Okumoto (1978)	-
3	I	A model by Schink & Wolverton (1978)	-
4	I	The De-eutrophication model by Moranda (1975)	-
5	II	Bayesian Reliability Growth model by Littlewood & Verrall (1973)	LV73
6	II	Bayes Empirical Bayes or Hierarchical model by Mazzuchi & Soyar (1988)	-
7	II	Bayesian Differential Debugging model by Littlewood (1980)	-
		Type II	
8	III	Time-dependent Error Detection model by Goel & Okumoto (1979b)	GO79b
9	III	Logarithmic Poisson Execution Time model by Musa & Okumoto (1984)	-

§ 2.2 The JM Model

Jelinski and Moranda (1972) proposed a simple model for describing failures in computer software. It models time between failures by considering their failure rates. The model that we shall consider here is one that has been widely discussed in the literature.

Suppose that the total number of bugs in the software is N (which is unknown); and suppose that each time the software fails, one bug is "perfectly" corrected. Let us assume that the failure rate at any point of time is proportional to the residual number of bugs in the program. That is, for $t > 0$ we have,

$$r_T(t) \propto N_t \ ,$$

which implies that,

$$r_T(t) = \Lambda \ N_t \ , \tag{2.2.1}$$

where Λ is the constant of proportionality and N_t denote the number of faults remained in the software, at time t.

Thus, the failure rate of T_i $(i = 1,2,\dots,N)$, the i^{th} time between failure, is

$$r_{T_i}(t|N,\Lambda) = \Lambda \ (N-i+1) \tag{2.2.2}$$

for some unknown constant Λ and $t > 0$.

Since constant failure rate is characterization of exponential random variable, we have, for fixed Λ and N; T_i is an exponential random variable with mean $\{\Lambda(N-i+1)\}^{-1}$. Therefore the cumulative distribution function (cdf) of T_i ($i = 1,2,\dots,N$) is given by,

$$F_{T_i}(t) = P[\ T_i \le t\ ]$$

$$= 1 - \exp\{-(N-i+1)\Lambda t\} \qquad \text{for } t>0 \qquad (2.2.3)$$

For JM model, underlying assumptions are as follows,

i) there are N bugs in the software (N is unknown),

ii) failure rate at any point of time is proportional to the residual number of bugs in the program,

iii) each time a failure occurs, the fault that causes is immediately and perfectly removed and no new error is

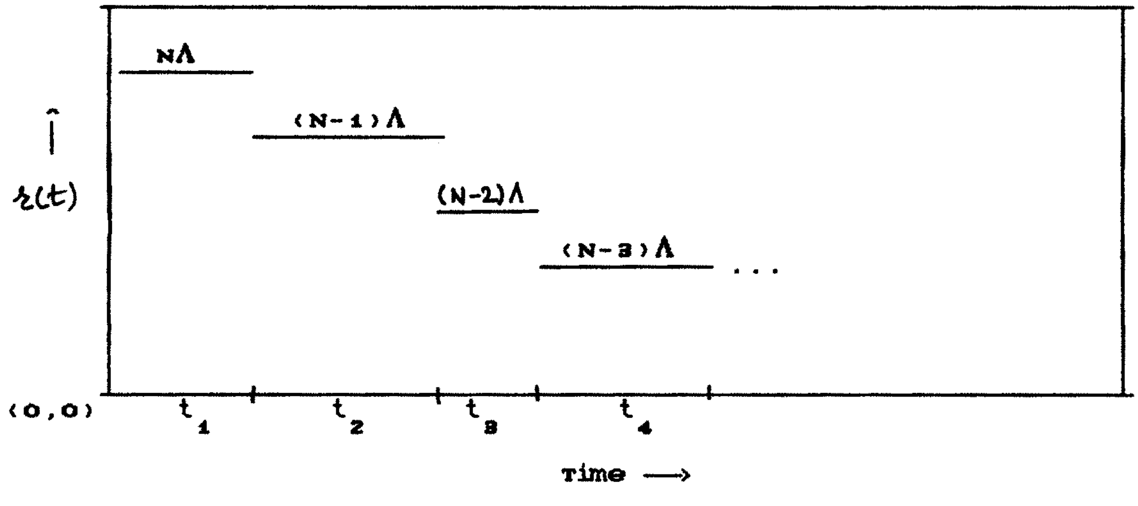

Fig.(2.1): The failure rate for JM model.

created. Thus it assumes that successive failure rates are decreasing, and

iv) each error is equal in the sense that it contributes the same amount Λ to the failure rate.

The figure (2.1) will illustrate the failure rate generated by JM model debugging process.

We now give, an alternative interpretation for JM model which is reported by Langberg and Singpurwalla (1985). Let N^* represents the total number of distinct "input types" to the software. N^* is assumed to large or conceptually infinite. Let $N << N^*$ be the number of input types which results in the inability of the software system to perform its desired function (N is assumed unknown).

Suppose that the input arrives at the software system according to the postulates of a Poisson Process with rate ω.

Then given N^*, N and ω, the probability that the software encounters no failures in a time interval $[0, t)$ is given by,

$$\overline{F}(t|N,N^*,\omega) = \sum_{i=1}^{\infty} \left[e^{-\omega t} (\omega t)^j / j!\right] \left[(N^* - N)/N^*\right]^j \qquad (2.2.4)$$

where the first term inside the summation sign denotes the probability that j inputs (shocks) are received in time t, and the second term inside the summation sign denotes the probability that all j input do not lead to a failure of the software.

Argument of this type form the basis of the theory of shock models and wear processes, which have played an important role in reliability theory (Barlow and Proschan, 1975, p.52).

From equation (2.2.4) we have,

$$\bar{F}(t|N,N^*,\omega) = \sum_{j=1}^{\infty} e^{-\omega t} \left[(\omega t)\ (1 - N/N^*)\right]^j / j!$$

$$\bar{F}(t|N,N^*,\omega) = e^{-\omega t} \exp\left[(\omega t)\ (1 - N/N^*)\right]$$

$$= \exp[\ -\ (\omega t)\ N/N^*\ ] \qquad ;\ \text{for } t \geq 0 \qquad (2.2.5)$$

Implying that the time to first failure of the software, say T_1, has an exponential distribution with a scale parameter $(\omega N/N^*)^{-1}$.

Following the error correction policy, T_i ($i = 1,2,\ldots,N$), the time between the $(i-1)^{th}$ and the i^{th} failure of the software, has survival functions,

$$\bar{F}_{T_i}(t|N,N^*,\omega) = P[\ T \geq t|N^*,N,\omega\ ]$$

$$= \exp[\ -\ (\omega t)\ (N-i+1)/N^*\ ] \qquad \text{for } t \geq 0 \qquad (2.2.6)$$

Hence the failure rate at time t, for the i^{th} time between failure is,

$$r_{T_i}(t|N,\Lambda) = \omega\ (N-i+1)/N^* \qquad (2.2.7)$$

Thus if we assign $\omega/N^* = \Lambda$ then we get failure rate for the

JM model. This implies that the JM model is a special case of the model described above.

Inference related to the parameters of JM model namely N and Λ will be reported in Chapter-III. In the following section we give Imperfect Debugging model by Goel and Okumoto (1978).

§ 2.3 Imperfect Debugging model

This model is like JM model, but it assumes that there is a probability p ($0 \le p \le 1$) of fixing a bug when it is encountered. This means that, after i faults have been found, we expect i*p faults to have been corrected instead of i. Thus the failure rate of T_i ($i = 1,2,\ldots,N$) is,

$$r_{T_i}(t|N,\Lambda,p) = \Lambda\ (N-p(i-1)) \qquad \text{for } t \ge 0 \qquad (2.3.1)$$

This is an attempt to improve upon the JM model by altering its assumption that a perfect fix of a bug always occurs. JM model can be obtained from this model by simply putting $p = 1$.

§ 2.4 A model by Schink and Wolverton

In this model, the failure rate is assumed proportional to the product of the number of faults remaining and the elapsed time since last failure. All other assumptions are as it is in JM model.

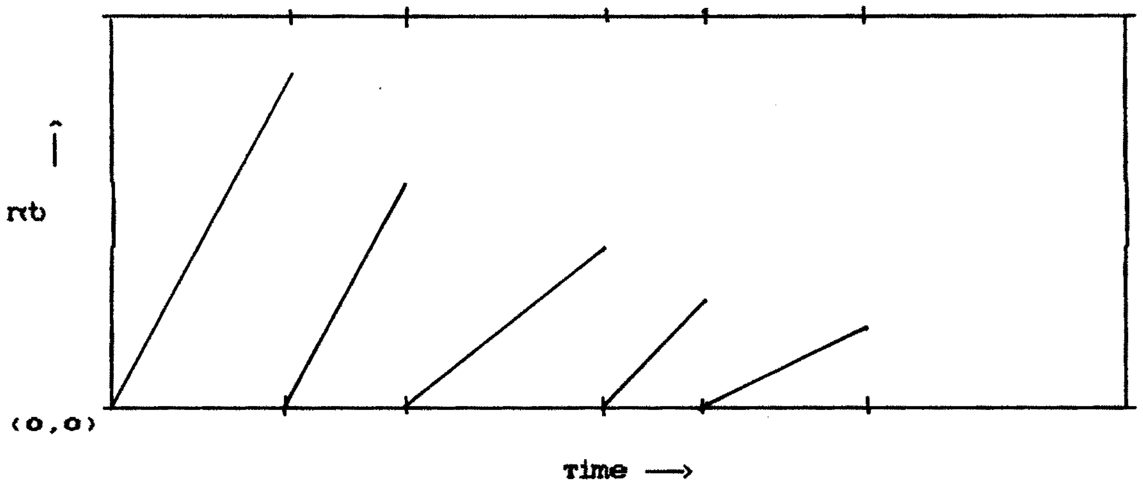

Fig.(2.2): The failure rate for the model of Schink and Wolveton.

Thus failure rate of T_i (i = 1,2,...,N) is,

$$r_{T_i}(t|N,\Lambda) = \Lambda\ (N-i+1)t \qquad \text{for } t \geq 0 \qquad (2.4.1)$$

This model differs from those of JM model and Goel & Okumoto (1978) Model, in that the failure rate does not decreases monotonically. Immediately after the i^{th} failure, the failure rate drops to zero, and then increases linearly with slope (N-i) until the $(i+1)^{th}$ failure occurs. The figure (2.2) will illustrate this rate.

§ 2.5 The De-eutrophication model of Moranda

In this model it is assumed that, the fixing of bugs that causes early failures in the system reduces the failure rate more than the fixing of bugs that occur later, because these early bugs are likely to be the bigger ones. Keeping this in mind, the author proposed the failure rate should remain constant for each T_i, but that it should be made decreasing geometrically in i after each failure, that is, the failure rate for T_i (i = 1,2,...,N) is given by,

$$r_{T_i}(t|D,k) = D\,k^{i-1} \quad ; \text{ for } t \geq 0,\ D > 0\ \&\ 0 < k < 1 \qquad (2.5.1)$$

where D is a constant which represents the initial failure rate. This relationship is illustrated in the figure (2.3).

In this model, the drop in failure rate after i^{th} failure is $D(1-k)k^{i-1}$, while it is Λ in the JM model. The assumption of perfect fix, with no introduction of new bugs during the fix is retained. Since the failure rate given in equation (2.5.1) is

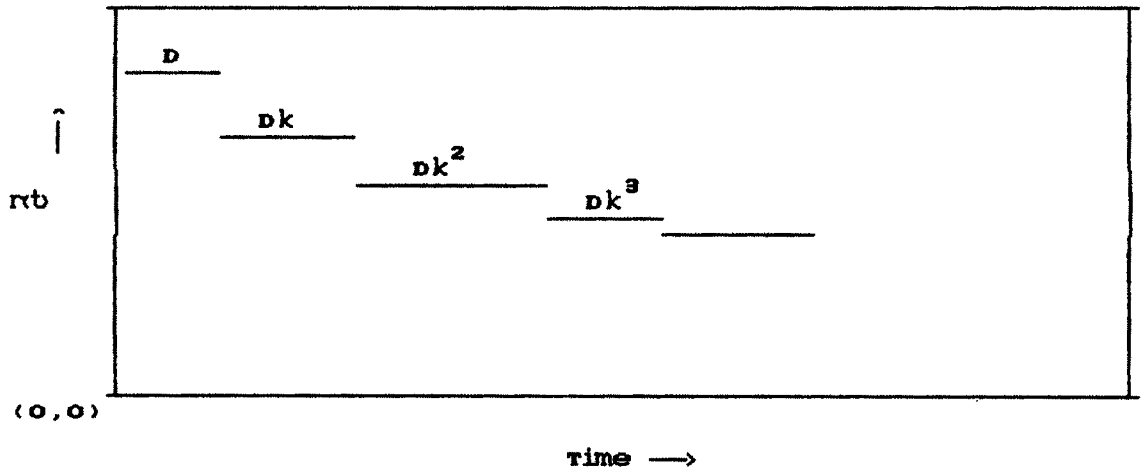

Fig.(2.3): The failure rate of the De-eutrophication model.

independent of t, T_i follows an exponential distribution with mean $(Dk^{i-1})^{-1}$, that is,

$$P[\ T_i \geq t\ ] = \exp[-\ Dk^{i-1}\ t] \qquad \text{for } t \geq 0 \qquad (2.5.2)$$

§ 2.6 Bayesian Reliability Growth model

A Bayesian approch to software reliability was taken by Littlewood and Verrall (1993). They viewed software reliability as a measure of strength of belief that a program will operate successfully. Littlewood and Verrall modeled failure rate as a random variable.

Specifically, they declared that the i^{th} interoccurence time T_i between failures to be exponential with failure rate Λ_i (i=1,2,...,N). The probability density function (pdf) of T_i, for i = 1,2,...,N is given by,

$$f_{T_i}(t|\Lambda_i) = \Lambda_i \exp[-\Lambda_i\ t] \qquad (2.6.1)$$

$$\text{for } t \geq 0 \text{ and } \Lambda_i \geq 0.$$

Also instead of Λ_i decreasing, as is assumed in JM model, they would merely require that the sequence of Λ_i's be stochastically decreasing, that is, for i = 1,2,...,N and $\lambda \geq 0$

$$P[\ \Lambda_{i+1} < \lambda\ ] \geq P[\ \Lambda_i < \lambda\ ] \qquad (2.6.2)$$

Let us assumes a Gamma distribution for Λ_i with shape parameter α and scale parameter $\Psi(i)$, where $\Psi(i)$ is monotonically increasing function of i, that is, if $\Pi(.)$ denote the pdf corresponding to the failure rate Λ_i, then we have for $i=1,2,\ldots$ and $\lambda \geq 0$,

$$\Pi_{\Lambda_i}(\lambda|\alpha,\Psi(i)) = \frac{\Psi(i)^{\alpha}}{\Gamma\alpha} \lambda^{\alpha-1} e^{-\Psi(i)\lambda} \qquad (2.6.3)$$

The function $\Psi(i)$ is supposed to describe the quality of the programmer and the programming task. The concept of failure rate random variable reflects uncertainty in the effectiveness of the fault correction process. Therefore the marginal distribution of T_i $(i = 1,2,\ldots,N)$ is given by,

$$\begin{aligned}
f_{T_i}(t) &= {}_0\int^{\infty} f(t, \lambda)\, d\lambda \\
&= {}_0\int^{\infty} f(t|\lambda)\, \Pi_{\Lambda_i}(\lambda|\alpha,\Psi(i))\, d\lambda \\
&= {}_0\int^{\infty} \Big[\lambda \exp(-\lambda\ t)\Big] \left[\frac{\Psi(i)^{\alpha}}{\Gamma\alpha} \lambda^{\alpha-1} e^{-\Psi(i)\lambda} \right] d\lambda \\
&= \frac{\Psi(i)^{\alpha}}{\Gamma\alpha}\, {}_0\int^{\infty} \lambda^{(\alpha+1)-1} e^{-(\Psi(i)+t)\lambda}\, d\lambda \\
&= \frac{\Psi(i)^{\alpha}}{\Gamma\alpha} \frac{\Gamma(\alpha+1)}{(t+\Psi(i))^{(\alpha+1)}} \\
&= \alpha \left[\frac{\Psi(i)}{t+\Psi(i)} \right]^{\alpha} (t+\Psi(i))^{-1} \qquad (2.6.4)
\end{aligned}$$

The mean time to failure (MTTF) between $(i-1)^{th}$ and i^{th} failures, denoted by $\Theta(i)$ and is given by,

$$\Theta(i) = E(T_i)$$

$$= {}_0\int^{\infty} t \, f_{T_i}(t) \;\; dt$$

$$= {}_0\int^{\infty} t \, \alpha \left[\frac{\Psi(i)}{t + \Psi(i)} \right]^{\alpha} (t+\Psi(i))^{-1} \, dt$$

$$= \frac{\alpha}{\Psi(i)} \, {}_0\int^{\infty} t \left[\frac{1}{[t / \Psi(i)] + 1} \right]^{(\alpha+1)} dt$$

Substituting $[t/\Psi(i)] = v$, we get,

$$\Theta(i) = \alpha \, \Psi(i) \, {}_0\int^{\infty} [\, v^{2-1} / (1+v)^{\alpha+1} \,] \, dv$$

Using $B(m,n) = {}_0\int^{\infty} [\, v^{m-1} / (1+v)^{n-1} \,] \, dv$

$$= \frac{\Gamma m \, \Gamma n}{\Gamma(m+n)}$$ we can write,

$$\Theta(i) = \alpha \, \Psi(i) \, B(2, (\alpha-1))$$

$$= \alpha \, \Psi(i) \, \frac{\Gamma 2 \, \Gamma(\alpha-1)}{\Gamma(\alpha+1)}$$

$$= \Psi(i) / (\alpha-1) \qquad (2.6.5)$$

From equation (2.6.4) we have,

$$F_{T_i}(x) = {}_0\int^{x} \alpha \left[\frac{\Psi(i)}{x + \Psi(i)} \right]^{\alpha} (x + \Psi(i))^{-1} \, dx$$

$$F_{T_i}(x) = \frac{\alpha}{\Psi(i)} \, {}_0\int^{x} \left[\frac{1}{[x/\Psi(i)] + 1} \right]^{(\alpha+1)} dx$$

Substituting $[x/\Psi(i)] = u$, we get,

$$F_{T_i}(x) = \alpha \, {}_0\int^{x/\Psi(i)} [\, 1 \,/\, (1+u)^{\alpha+1} \,] \, du$$

$$= \left[-\,1/\,[1+u]^{\alpha} \right]_0^{[x/\Psi(i)]}$$

$$= 1 - \{1+(x/\Psi(i)]\}^{-\alpha} \qquad (2.6.6)$$

which implies that,

$$\bar{F}_{T_i}(x) = 1 - F_{T_i}(x)$$

$$= \left[1 + \frac{x}{\Psi(i)} \right]^{-\alpha}$$

$$= \left[\frac{[x+\Psi(i)]}{\Psi(i)} \right]^{-\alpha}$$

$$= \left[\frac{\Psi(i)}{[\Psi(i)+x]} \right]^{\alpha} \qquad (2.6.7)$$

Using equations (2.6.4) and (2.6.7), the failure rate for resulting program is,

$$\Lambda_i(t) = f_{T_i}(t) \,/\, \bar{F}_{T_i}(t)$$

$$= \alpha \,/\, [t_i + \Psi(i)] \qquad (2.6.8)$$

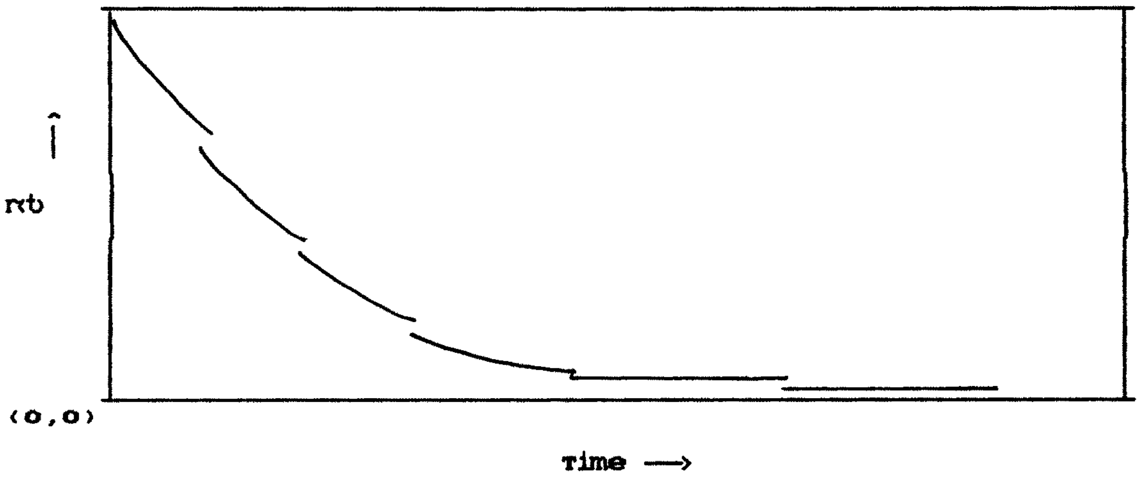

Fig.(2.4): The failure rate for the LV73 model

Figure (2.4) will illustrate this relationship. The concept of failure rate random variable reflects uncertainty in the effectiveness of the fault correction process. The failure rate decreases continuously with t and experiences discontinuities of various heights at each failure. The heights usually but not necessarily decreases with failure experienced.

Since we can choose the reliability growth function $\Psi(i)$ arbitrarily, the model is general and flexible. The model can fall in different classifications depending upon the form of the reliability growth function $\Psi(i)$.

Littlewood and Verrall (1973) have suggested two forms for $\Psi(i)$; which are given by,

$$\Psi_1(i) = \beta_0 + i\,\beta_1 \qquad (2.6.9)$$

and

$$\Psi_2(i) = \beta_o + i^2 \beta_1 \tag{2.6.10}$$

For the both cases there is no restriction on i. Hence infinite failures can be experienced. The families are inverse linear and inverse polynomial (2^{nd} degree), respectively. Values of parameters for a growth function and comparisons determining growth function is best established by testing goodness of fit to the data.

§ 2.7 Bayes Empirical Bayes or Hierarchical model

This is an extension to the LV73 model which is discussed in the previous section. As with the original model, they assumed T_i to be exponentially distributed with scale Λ_i. Then they proposed two ideas for describing Λ_i here called model A and B respectively. These are given as follows.

2.7.1 Model A

Assume that Λ_i is a Gamma variate with parameters α and β and also assume that α and β are described by Uniform and another Gamma distribution respectively. In other words,

$$\Pi_{\Lambda_i}(\lambda|\alpha,\beta) = \frac{\beta^{\alpha}}{\Gamma\alpha} \lambda^{\alpha-1} e^{-\beta\lambda} \qquad \lambda \geq 0$$

$$\pi(\alpha|v) = 1/v \qquad o \leq \alpha \leq v \tag{2.7.1}$$

$$\pi(\beta|a,b) = b^{a} \ \beta^{a-1} \ e^{-b\beta}/\Gamma a \qquad \beta \geq 0 \ , \ a, \ b > 0 \qquad (2.7.1)$$

where v, a and b are known.

2.7.2 Model B

Assume that Λ_i is described described as in LV73, that is,

$$\Pi_{\Lambda_i}(\lambda|\alpha,\Psi(i)) = \Psi(i)^{\alpha} \ \frac{\lambda^{\alpha-1}}{\Gamma\alpha} \ e^{-\Psi(i)\lambda} \qquad \lambda \geq 0 \qquad (2.7.2)$$

and that $\Psi(i) = \beta_o + i\ \beta_1$ except α, β_o and β_1 are described by probability distribution as follows,

$$\pi(\alpha|v) = 1/v \qquad o \leq \alpha \leq v$$

$$\pi(\beta_o|a,b,\beta_1) = b^{a} \ (\beta_o+\beta_1)^{a-1} \ \exp(-b(\beta_o+\beta_1) \ / \ \Gamma a \qquad (2.7.3)$$

$$-\beta_o \geq \beta_1 \ , \ a, \ b > 0$$

$$\pi(\beta_1|c,d) = d^{c} \ \beta_1^{c-1} \ \exp(-d\beta_1)/ \ \Gamma c \qquad \beta_1 \geq 0 \ , \ c, \ d > 0$$

So α is described by a Uniform distribution, β_o by a shifted Gamma and β_1 by another Gamma and there is dependence between β_o and β_1. By assuming β_1 to be degenerate at 0, model A is obtained from model B.

§ 2.8 Bayesian Differential Debugging model

This model can be considered as an elaboration of the model LV73. In LV73 model it is assumed that Λ_i, the failure rate of the i^{th} time between failure, is described as a Gamma random variable. In addition to this, in this model it is assumed that there are N bugs in the system and Λ_i is specified as a function of the remaining bugs.

In particular, if τ time has been elapsed and (i-1) bugs have been removed then the failure rate of the program is ,

$$\Lambda_i = \phi_1 + \phi_2 + \ldots + \phi_{N-i+1} \qquad (2.8.1)$$

where ϕ_i's are independent and identically distributed Gamma variants with shape parameter α and scale parameter β.

Thus, Λ_i has a Gamma distribution with shape $[\alpha\ (N-i+1)]$ and scale β. Hence the pdf of Λ_i is given by,

$$\Pi_{\Lambda_i}(\lambda|\alpha,\beta,N) = \frac{\beta^{\alpha(N-i+1)}}{\Gamma\alpha(N-i+1)} \lambda^{\alpha(N-i+1)-1} e^{-\beta\lambda} \qquad ; \lambda \geq 0 \qquad (2.8.2)$$

Thus,

$$E(\Lambda_i|\ \alpha,\beta,N\ ) = \alpha\ (N-i+1)/\beta \qquad (2.8.3)$$

which is linearly decreasing function of i. Other assumptions are identical as that of original LV73 model.

§ 2.9 Time-dependent Error detection model

In this model, Goal and Ockumoto reasoning from assumptions similar to those of JM model, describing failure detection as a NHPP with an exponential decaying rate function. The assumptions for the model are as follow.

i) there are no failure experienced at time 0, that is, $P[\ M(0) = 0\ ] = 1$,

ii) the counting process $\{M(t);\ t \geq 0\}$ has independent increments. This implies that, the number of failure experienced during $(t+\delta t)$ is independent of its past history. That is, the future $M(t+\delta t)$ of the process depends only on the present state $M(t)$ and is independent of its past $M(x)$ for $x < t$.

iii) the probability that a failure will occur during $(t,\ t+\delta t)$ is $\lambda(t)\delta t + o(\delta t)$, where $\lambda(t)$ is the failure intensity of the process and $o(\delta t)$ is a function such that, $o(\delta t)/\delta t$ tends to zero as δt tends to zero. (In practice the second or higher effects of δt are negligible).

iv) the probability that more than one failure will occur during $(t,\ t+\delta t)$ is $o(\delta t).\delta t)$.

Also expected number of failures $\mu(t)$, is assumed to be non-decreasing and bounded above function of t with following boundary conditions,

$$\mu(t) = E\ [M(t)] = \begin{cases} o & t = o \\ a & t \longrightarrow \infty \end{cases} \qquad (2.9.1)$$

where 'a' denotes expected numbers of errors in the software.

And the expected failures in the the time interval $(t,\ t+\delta t)$

is proportional to the number of undetected errors in the software; or

$$\mu(t+\delta t) - \mu(t) = b\ (\ a - \mu(t))\ \delta t + o(\delta t) \qquad (2.9.2)$$

where b is a constant of proportionality (fault detection rate).

Thus from equation (2.9.2) we get,

$$\mu'(t) = \lim_{\delta t \longrightarrow o} \frac{\mu(t+\delta t)-\mu(t)}{\delta t}$$

$$= \lim_{\delta t \longrightarrow o} \frac{b(a-\mu(t))\delta t+o(\delta t)}{\delta t}$$

$$= b\ (a - \mu(t)) \qquad (2.9.3)$$

This implies that,

$$\mu'(t) + b\ \mu(t) = ab$$

Multiplying both sides of above equation by e^{bt}, we get

$$e^{bt}\ \mu'(t) + e^{bt}\ b\ \mu(t) = e^{bt}\ ab$$

which equivalent to,

$$\frac{d}{dt}\left\{\ e^{bt}\ \ \mu(t)\ \right\} = e^{bt}\ ab \qquad (2.9.4)$$

Integrating both sides of equation (2.9.4) w.r.t. t, we have,

$$e^{bt}\ \ \mu(t) = e^{bt}\ a + c \qquad (2.9.5)$$

where c is constant of integration.

Since from (2.9.1), $\mu(0) = 0$; we get $c = -a$.

Thus equation (2.9.5) reduces to,

$e^{bt}\ \mu(t) = e^{bt}\ a - a$

Hence,

$\mu(t) = a\ (1-e^{-bt})$ (2.9.6)

Therefore, $\lambda(t) = \mu'(t)$

gives, $\lambda(t) = a\ b\ e^{-bt}$ (2.9.7)

(2.9.7)

We illustrate this relation by using figure (2.5). The function $\mu(t)$ completely specifies a particular Poisson process, and the distribution of M(t) is given by,

$$P[\ M(t) = n\ ] = \frac{[\mu(t)]^n}{n!}\ e^{-\mu(t)} \qquad n = 0,1,... \qquad (2.9.8)$$

Two assumptions of JM model are modifided here. First, the

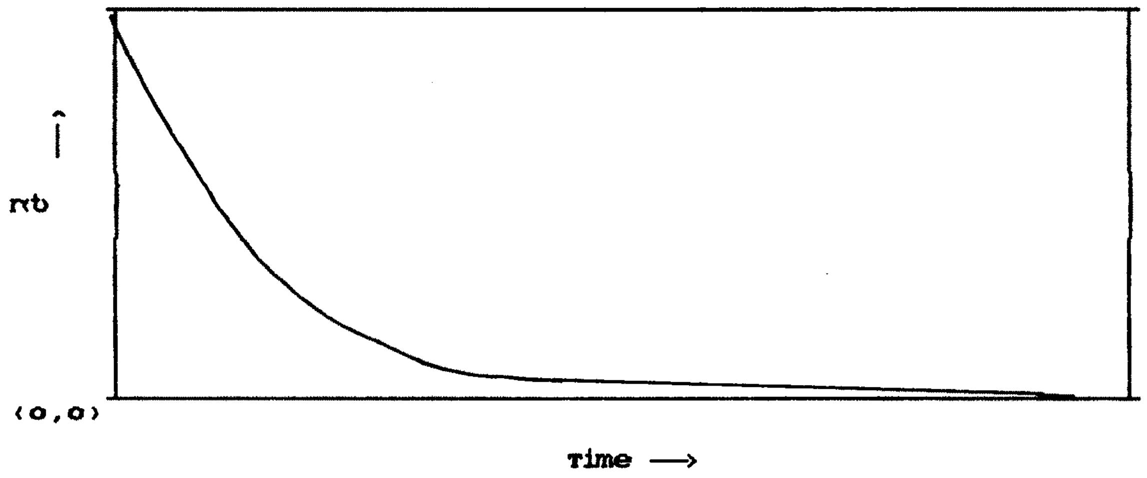

Fig.(2.5): The intensity function for the GO79b model.

total number of errors in the software is a random variable with mean ´a´, which is fixed in the JM model. Secondly, the times between successive failures are assumed dependent, while the JM model assumes independence. Authors claim that these modifications are a better description of the actual occurrence of failure in software. The inference regarding parameters of this model is studied in Chapter - IV.

§ 2.10 Logarithmic Poisson Execution Time model

In this model, the failure rate $\lambda(t)$ is expressed in terms of expected number of failures $\mu(t)$, in time $[0,t)$ as,

$$\lambda(t) = \lambda_o \exp(-\theta\, \mu(t)) \qquad ; \theta > 0 \qquad (2.10.1)$$

where λ_o denote the initial failure rate and θ the failure rate decay parameter.

From above equation we see that $\lambda(t)$ decreases exponentially with $\mu(t)$. Observe that the fixing of earlier failures will reduce $\lambda(t)$ more than the fixing of later once, because these earlier failures are assumed to be those that occur more frequently.

Since we are modeling the number of failures by Poisson Process, we have another relationship between $\lambda(t)$ and $\mu(t)$ as,

$$\mu(t) = \int_0^t \lambda(s)\, ds \qquad (2.10.2)$$

That is,

$$\lambda(t) = \frac{d}{dt}\mu(t) \qquad (2.10.3)$$

From equation (2.10.1) and (2.10.3) we get,

$$\frac{d}{dt}\mu(t) = \lambda_o \exp(-\theta\, \mu(t)) \qquad ; \theta > 0$$

or

$$\left\{\frac{d}{dt}\mu(t)\right\} \exp(\theta\, \mu(t)) = \lambda_o \qquad ; \theta > 0$$

Multiplying by θ to both sides of above equation gives,

$$\left\{\frac{d}{dt}\mu(t)\right\} \theta \exp(\theta\, \mu(t)) = \theta\, \lambda_o \qquad ; \theta > 0$$

which is equivalent to,

$$\left\{\frac{d}{dt}\exp(\theta\, \mu(t))\right\} = \theta\, \lambda_o \qquad ; \theta > 0 \qquad (2.10.4)$$

Integrating equation (2.10.4) w.r.t. 't', we get,

$$\exp(\theta\, \mu(t)) = \theta\, \lambda_o\, t + c_1 \qquad ; \theta > 0$$

where c_1 is the constant of integration.

Since $\mu(0) = 0$, $c_1 = 1$;

$$\exp(\theta\, \mu(t)) = \theta\, \lambda_o\, t + 1 \qquad ; \theta > 0$$

Thus the mean value function is obtained as,

$$\mu(t) = \{\log_e(\theta \lambda_o t + 1)\} / \theta \qquad (2.10.5)$$

From equation (2.10.1) and (2.10.5), we obtain $\lambda(t)$ as,

$$\lambda(t) = \lambda_o / (\theta \lambda_o t + 1) \qquad ; \theta > 0 \qquad (2.10.6)$$

which is the inverse linear function of t. This relationship is illustrated in the figure (2.6).

It is similar to the plot of intensity function for G079b model, except that the tail is thicker. By using equation (2.9.8), we can write for n = 0,1,2,...

$$P[\,M(t) = n\,] = \left\{ \frac{\log_e(\lambda_o \theta t + 1)}{\theta} \right\}^n (\lambda_o \theta\, t + 1)^{-1/\theta} / \, n! \qquad (2.10.7)$$

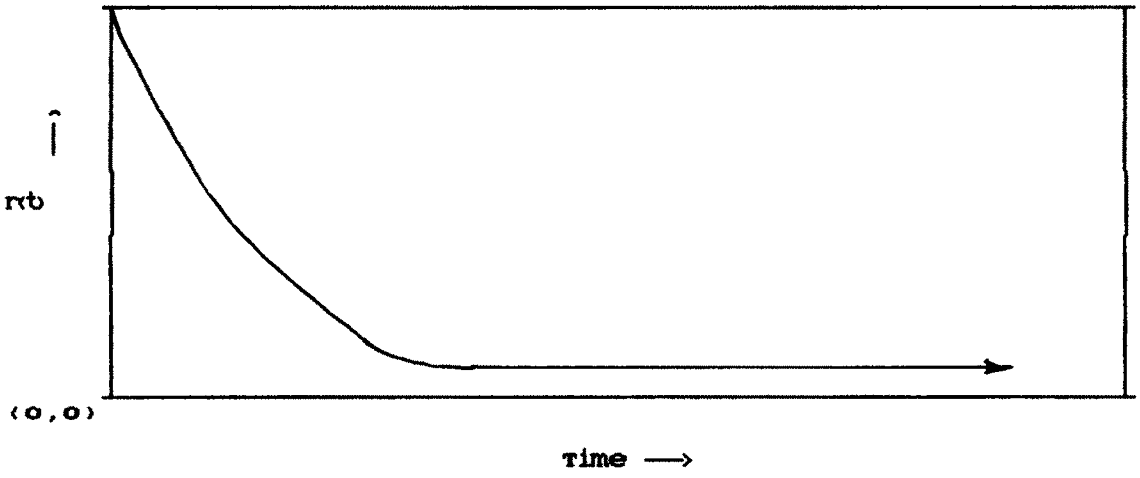

Fig.(2.6): The failure rate for Musa & Okumoto (1984) model.

* * * * *