文章编号: 1001-506X(2009)06-1496-04

考虑右删失数据的改进 Jelinski-Moranda 软件可靠性模型

李海峰, 陆民燕, 王学成
(北京航空航天大学工程系统工程系, 北京 100083)

摘 要: 针对软件可靠性测试停止时间不对应失效的右删失问题,给出考虑完全右删失失效数据的改进 Jelinski-Moranda (J-M) 模型。定量分析了右删失失效数据对可靠性参数评估值的影响,在此基础上分别给出基于残存缺陷比率、失效率以及可靠度函数的软件可靠性测试停止准则。最后将改进模型应用于一组失效数据集,计算结果表明,改进后的模型可充分挖掘右删失数据中蕴含的时间信息,得到更准确的参数评估值。并且所提出的停止准则可有效指导软件可靠性测试活动的开展,具有较好的工程应用价值。

关键词: Jelinski-Moranda 模型; 右删失数据; 软件可靠性测试; 测试停止准则
中图分类号: TP 311.5 **文献标志码**: A

Modified Jelinski-Moranda model with right-censored data

LI Hai-feng, LU Min-yan, WANG Xue-cheng
(Dept. of Engineering and System Engineering, Beihang Univ., Beijing 100083, China)

Abstract: The non-failure stops (viz, the stopping time of the testing doesn't correspond to a failure) of software reliability testing can be viewed as a type of right-censored failure data. However, the traditional J-M model can't deal with this type of right-censored data directly. To resolve this problem, a modified J-M model with respect to the right-censored data is presented. Based on this modified model, the effect of the right-censored data on the estimations of the model parameters is analyzed, and then three separate stopping criteria for software reliability testing with regard to the ratio of residual defects, reliability and failure density are proposed. Finally, the modified model is applied on a classical failure data set. The results of the application show that the proposed model can deal with the right-censored data well since it can fully utilize the time information contained in the right-censored data.

Keywords: J-M model; right-censored data; software reliability testing; stopping criteria

0 引 言

在软件可靠性增长测试中,若发现 n 个失效后,迟迟不能出现第 $n+1$ 个失效,假设时刻 t_e 停止测试,此时第 $n+1$ 个失效时间 t_{n+1} 未知,只知其大于 t_e,称 t_{n+1} 在 t_e 处是右删失的[1]。由于 t_e 不对应一个失效,传统的软件可靠性增长模型(SRGMs)是无法直接将其和前 n 个失效数据一起处理的,这就是失效数据右删失问题。(Jelinski-Moranda, J-M)模型是一个经典 SRGM,至今还受到软件可靠性工程界的重视。本文仅就 J-M 模型在完全失效数据下的右删失问题展开研究。

J-M 模型暗含着这样一个假设:软件可靠性评估值仅由软件的失效数据所决定,在第 n 个失效时间 t_n 至 t_e 这段时间间隔内由于没有发生新的失效,软件可靠性指标评估值不会变化,也即 t_e 所蕴含的时间信息与 t_n 没有区别。这个假设与实际情况是不相符的。文献[2]指出软件可靠性的行为描述应由软件内部特征和外部环境共同决定,在 t_n 至 t_e 这段时间内软件内部特征确实没有发生变化,但这并不意味着测试用例数等对软件可靠性评估可能产生影响的因素没有发生变化,广义上 t_e 时的可靠性指标评估值与 t_n 时完全一致是不准确的。所以在软件可靠性评估中考虑 t_e 所蕴含的时间信息是十分有意义且必要的。

原始 J-M 模型并未考虑失效数据的右删失问题[3],通常会舍弃数据 t_e。而利用可处理成组数据的改进 J-M 模型[4-7]来处理右删失数据又不能充分利用前 n 个完全失效数据的信息,这些都与最大化信息准则相违背[2]。文献[2]阐述了如何改进四个传统 SRGMs 以适应因各种原因所引

收稿日期: 2008-03-16; 修回日期: 2008-07-10。
作者简介: 李海峰(1981-),男,博士研究生,主要研究方向为软件可靠性度量,软件可靠性测试等。E-mail: lihaifeng@dse.buaa.edu.cn

发的非失效停止(non-stop)数据集,但其没有就本文开始所提到的具体问题(完全数据的右删失)展开针对性的研究。文献[8-9]从非参数角度出发,提出了针对删失数据的软件可靠性模型。这两篇文献中的删失数据指的是软件残存缺陷的可能失效时间,与本文讨论的问题并不一致。

本文首先给出可以处理右删失数据的改进 J-M 模型,在此基础上就右删失数据对模型参数和可靠性参数评估值的影响进行了严格的理论分析,由此给出右删失情况下,基于残存缺陷率、软件可靠度以及失效率的软件可靠性测试停止准则。

1　考虑右删失数据的改进 J-M 模型

1.1　模型假设与符号约定

(1) 初始缺陷数为 N,测试中共发现 n 个缺陷,测试在 t_e 时刻截止,t_e 不对应一个失效;

(2) 程序失效率与剩余缺陷个数成正比,比例常数记为ϕ;

(3) 累积失效时间序列$\{t_1,t_2,\cdots,t_n,t_e\}$,失效间隔时间序列$\{t'_1,t'_2,\cdots,t'_n,t'_e\}$。

1.2　模型参数估计

J-M 模型无条件最大似然函数的一般形式为[10]

$$L(N,\phi;\ t_1,t_2,\cdots,t_n,t_e)=$$

$$\left[\prod_{i=1}^{n} f(t_i\mid t_1,\cdots,t_{i-1})\right]P[t_{n+1}>t_e\mid t_1,t_2,\cdots,t_n] \tag{1}$$

对于二项式指数类模型,t_i 的条件密度函数仅依赖于 t_{i-1},因此,式(1)可简化为

$$L=\left[\prod_{i=1}^{n} f(t_i\mid t_{i-1})\right]P[t_{n+1}>t_e\mid t_n] \tag{2}$$

式中

$$f(t_i\mid t_{i-1})=-\frac{\mathrm{d}}{\mathrm{d}t_i}P[T_i>t_i\mid T_{i-1}=t_{i-1}] \tag{3}$$

给定 $T_{i-1}=t_{i-1}$时,有 T_i 的条件分布为

$$P[T_i>t_i\mid T_{i-1}=t_{i-1}]=[1-F(t_i\mid t_{i-1})]^{N-i+1} \tag{4}$$

其中 $F(t_i|t_{i-1})=1-\exp[-\phi(t_i-t_{i-1})]$,带入式(4)

$$P[T_i>t_i\mid T_{i-1}=t_{i-1}]=\exp\{-(N-i+1)\phi t'_i\} \tag{5}$$

联合式(3)、(5)可得

$$f(t_i\mid t_{i-1})=(N-i+1)\phi\exp\{-(N-i+1)\phi t'_i\} \tag{6}$$

整理式(6)可得

$$\left[\prod_{i=1}^{n} f(t_i\mid t_{i-1})\right]=$$

$$\exp\{-(N-n)\phi t_n\}\prod_{i=1}^{n}(N-i+1)\phi\exp(-\phi t_i) \tag{7}$$

其次由式(5)可得

$$P[t_{n+1}>t_n\mid t_n]=\exp\{-(N-n)\phi(t_e-t_n)\} \tag{8}$$

联合式(2)、(6)、(8),可得最大似然函数为

$$L=\exp(-\phi(N-n)t_e)\prod_{i=1}^{n}(N-i+1)\phi\exp(-\phi t_i) \tag{9}$$

将式(9)取对数并分别对 N 和ϕ取偏导,最终可得 N 和ϕ的最大似然估计方程为

$$\phi=\frac{n}{\sum_{i=1}^{n}t_i+t_e(N-n)}- \tag{10}$$

$$\frac{nt_e}{\sum_{i=1}^{n}t_i+t_e(N-n)}+\sum_{i=1}^{n}\frac{1}{N-i+1}=0 \tag{11}$$

特殊地,若 $t_e=t_n$,可证明式(10)和(11)与文献[3]中给出的原始 J-M 模型参数的最大似然估计是完全一致的,证明过程略。

2　改进 J-M 模型在软件可靠性测试中的应用

Littlewood 指出,软件无失效运行时间越长,则对于"软件的可靠程度是高的"的信任程度就越高[3]。因此,若给定 $t_1,t_2,\cdots,t_n$,则 t_e 越大,测试停止时的软件可靠度评估值就应该越高,失效率与残存缺陷的比例的评估值越低。本节将对以上 3 点结论进行严格的理论证明,并由此分别建立基于软件残存缺陷比率、失效率函数和可靠度的软件可靠性测试停止准则。

2.1　基于残存缺陷比率的测试停止准则

记 t 时刻软件的残存缺陷数为 $Q(t)$,发现的缺陷数记为 $M(t)$。则 $Q(t_e)$的条件分布为[10]

$$P[Q(t_e)=q\mid M(t_n)=n]=$$

$$C_{N-n}^{q}[1-\exp[(t_n-t_e)\phi]]^{N-n-q}\exp\{(t_n-t_e)\phi q\} \tag{12}$$

式(12)是二项式分布,则 $Q(t)$的期望值为

$$\sum_{q=0}^{N-n}P[Q(t_e)=q\mid M(t_n)=n]\cdot q=$$

$$(N-n)\exp\{-\phi(t_e-t_n)\} \tag{13}$$

记残存缺陷比率为

$\Omega_{t_e}=(N-n)\exp\{-\phi(t_e-t_n)\}/N$,则有

结论 1　Ω_{t_e} 的评估值关于 t_e 单调递减。

证明　若利用 N 和ϕ的无条件最大似然估计(式(10)、(11))来证明结论 1 是非常繁复的,故本文将从 J-M 模型的条件最大似然估计入手来证明结论 1(若无特殊说明,第二节所有证明都基于模型参数的条件最大似然估计)。J-M 模型的无条件与条件最大似然估计不过是形式上略有差异,但没有本质区别[10],因此不影响证明的正确性。J-M 模型关于 N 和ϕ在右删失数据上的条件最大似然估计为(推导过程与第 2 节类似,只是采用了 J-M 模型的条件最大似然函数,过程略)

$$\frac{n}{\phi}-\frac{nt_e}{\exp(\phi t_e)-1}-\sum_{i=1}^{n}t_i=0 \tag{14}$$

$$N=n/[1-\exp(-\phi t_e)] \tag{15}$$

首先证明 N 的评估值关于 t_e 是单调递减的。令 $y=\phi t_e$，则式(14)可以化为

$$nt_e\left(\frac{1}{y}-\frac{1}{e^y-1}\right)-\sum_{i=1}^{n}t_i=0$$

两边关于 t_e 解微分可得

$$\frac{\mathrm{d}y}{\mathrm{d}t_e}=\frac{e^y-1-\dfrac{(e^y-1)^2}{y}}{t_e[e^y y^2-(e^y-1)^2]}$$

式(15)变形为 $N=n/(1-e^{-y})$，则有

$$\frac{\mathrm{d}N}{\mathrm{d}t_e}=\frac{\mathrm{d}N}{\mathrm{d}y}\frac{\mathrm{d}y}{\mathrm{d}t_e}=\frac{-n\left[\dfrac{e^y}{e^y-1}-\dfrac{e^y}{y}\right]}{t_e\left[e^y-\dfrac{(e^y-1)^2}{y^2}\right]} \tag{16}$$

由于 $e^y-1>y$，则式(16)的分子为正。而对于分母，记 $z=e^y y^2-(e^y-1)^2$，可得

$$x(y)=\mathrm{d}z/\mathrm{d}y=e^y(y^2+2y+2-2e^y)$$

又由于 $\mathrm{d}x/\mathrm{d}y=2(y+1)-2e^y<0$ 恒成立，所以 x 关于 y 是单调递减的。又因为 $x(0)=0$，且 y 大于 0，所以 $x(y)<0$ 恒成立。由此可知 z 关于 y 是单调递减的，又因为 $z(0)=0$，且 y 大于 0，则 $z<0$ 恒成立，即式(16)的分母为负。所以，$\mathrm{d}N/\mathrm{d}t_e<0$。即 N 关于 t_e 是单调递减的。由式(15)可以看出，当 $t_e\to\infty$ 时，$N\to n$。

$$\text{又}\ \mathrm{d}\Omega/\mathrm{d}t_e=B_1+B_2=e^{-\phi(t_e-t_n)}\left(\frac{n}{N^2}\frac{\mathrm{d}N}{\mathrm{d}t_e}\right)-\left(1-\frac{n}{N}\right)\left\{\phi(t_e-t_n)e^{-\phi(t_e-t_n)}\frac{\mathrm{d}\phi(t_e-t_n)}{\mathrm{d}t_e}\right\} \tag{17}$$

对于 B_1，由于 $\mathrm{d}N/\mathrm{d}t_e<0$，所以 $B_1<0$。而对于 B_2，记 $y=\phi t_e$，对式(14)两边求导可得

$$\mathrm{d}\phi/\mathrm{d}t_e=(e^y-1-ye^y)\phi^2/(e^y y^2-(e^y-1)^2) \tag{18}$$

式(18)的分母在前面已经被证明恒小于 0($z<0$)。记 $s=e^y-1-ye^y$。显然，$s(0)=0$，又 $\mathrm{d}s/\mathrm{d}y=-ye^y<0$，即式(18)的分子也是恒小于 0 的。所以 $\mathrm{d}\phi/\mathrm{d}t_e>0$。即可得

$$\mathrm{d}\phi(t_e-t_n)/\mathrm{d}t_e=\phi+(t_e-t_n)\mathrm{d}\phi/\mathrm{d}t_e>0$$

则 $B_2<0$。结合 B_1 和 B_2，最终可得 $\mathrm{d}\Omega/\mathrm{d}t_e<0$，结论 1 得证。

基于残留缺陷比例的软件可靠性测试停止准则：由结论 1 可知，随着 t_e 的增长，软件中残留缺陷比率的评估值随之降低。因此，计算 t_e 时刻的 Ω_{t_e}，若其满足事先规定的需求，则可以在 t_e 时刻停止测试。

2.2 基于失效率函数 λ 的测试停止准则

结论 2 λ 的评估值关于 t_e 单调递减。

证明 失效率函数为：$\lambda=\phi(N-n)$，则有

$$\mathrm{d}\lambda/\mathrm{d}t_e=S_1\times S_2=\phi/\{e^y y^2-(e^y-1)^2\}\times\left\{(N-n)\phi(e^y-1-ye^y)-\frac{n}{t_e}\left(\frac{y^2e^y}{e^y-1}-ye^y\right)\right\} \tag{19}$$

由式(15)，$N-n=Ne^{-y}$，代入 S_2 并整理可得

$$S_2=N\phi(e^y-e^{-y}-2y)$$

记 $K(y)=e^y-e^{-y}-2y$，显然 $K(0)=0$，又 $\mathrm{d}K/\mathrm{d}t_e=e^y+e^{-y}-2>0$ 恒成立，则 $K>0$ 恒成立，所以 $S_2>0$。

将 K 和 S_2 代入式(19)，可得

$$\mathrm{d}\lambda/\mathrm{d}t_e=(KN\phi^2)/(e^y y^2-(e^y-1)^2)$$

结论 1 已经证明 $e^y y^2-(e^y-1)^2<0$ 恒成立，则 $\mathrm{d}\lambda/\mathrm{d}t_e<0$，所以结论 2 得证。

基于失效率的软件可靠性测试停止准则：由结论 2 可知，随着 t_e 的增长，软件失效率的评估值随之降低。因此，计算 t_e 时刻的 λ，若其满足事先规定的需求，则可以在 t_e 时刻停止测试。

2.3 基于可靠度函数 R 的测试停止准则

结论 3 R 的评估值关于 t_e 单调递增。

证明 对于变量 t_e，其后固定时间间隔 t_r 的可靠度函数为

$$R(t_r\mid t_e)=\exp\{-(N-n)t_r\phi\} \tag{20}$$

则有 $\mathrm{d}R/\mathrm{d}t_e=R\cdot(-t_r\mathrm{d}\lambda/\mathrm{d}t_e)$，由结论 2 可知 $\mathrm{d}\lambda/\mathrm{d}t_e<0$，则 $\mathrm{d}R/\mathrm{d}t_e>0$，结论 3 得证。

基于可靠度函数的软件可靠性测试停止准则：由结论 3 可知，随着 t_e 的增长，t_e 之后固定时间间隔 t_r 的软件可靠度的评估值随之增加。因此，计算 t_e 时刻后某固定时间间隔的软件可靠度 R，若其满足事先规定的需求，则可以在 t_e 时刻停止测试。

2.4 在无失效软件可靠性评估中的应用

安全关键软件在经过若干次测试和版本升级后，在释放前的软件可靠性测试中不发生失效是很正常的，此时传统的软件可靠性评估技术将无法使用。若将该软件在之前版本的软件可靠性测试中已经收集到历史失效数据集视为 $\{t_1,t_2,\cdots,t_n\}$，将最后阶段的软件可靠性测试中的无失效时间视为 t_e-t_n，则可以利用本文提出改进 J-M 模型对软件中残留缺陷比例、失效率等可靠性参数进行近似估计。当然，这种评估方法的适用性还需进一步的研究与实践。

3 实例计算

以 Musa 提供的失效数据集合 SYS1[10] 作为实例进行分析，SYS1 是一组经典的完全失效数据集合，经常被用于软件可靠性模型的验证工作。该数据集的失效数 n 为 136，t_n 为 88 682 个 CPU 秒。

(1) 缺陷总数 N 的估计。利用传统 J-M 模型对 SYS1 数据进行评估，得出缺陷总数 N 的预计值为 142。当测试停止时间 t_e 为不同值时，利用式(11)对缺陷总数 N 的预计结果如图 1 所示。由图可知，随着 t_e 的增加，N 的估计值呈

下降的趋势, 这也再次证明了结论 1 中关于 N 的估计值随着 t_e 单调递减的论断是正确的。而传统 *J-M* 模型由于无法利用 t_e 的信息量, 则不论测试何时停止, 都只能对 N 得出一个不变的估计值(142), 这显然是不合理的。

(2) 残存缺陷率的估计。当测试停止时间 t_e 分别为不同值时, Ω_{t_e} 的估计结果如表 1 所示。由表 1 可以看出, 残存缺陷率 Ω_{t_e} 随着 t_e 的增加而下降, 这也再次证明了结论 1 的正确。若开发方对软件残存缺陷率的要求为小于0.015, 则可靠性测试在 t_n 之后的无失效时间大于 12 000 个 *CPU* 秒时即可停止。软件可靠度函数以及失效率在不同 t_e 的估计值的变化趋势也与文中的相应结论相吻合, 限于篇幅, 不再展开。

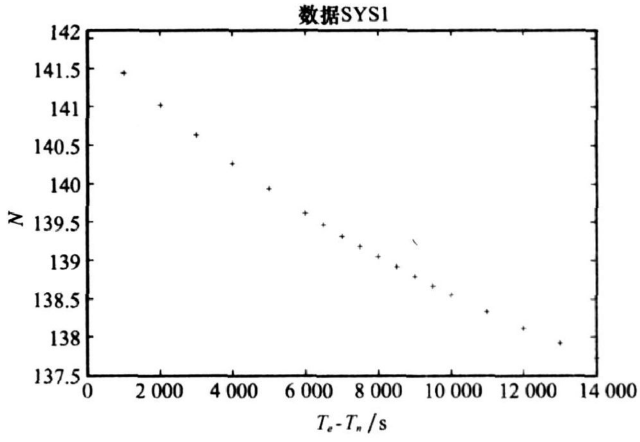

图 1　缺陷总数 *N* 的估计结果

表 1　残存缺陷率 Ω_{t_e} 的估计结果(单位: CPU 秒)

$t_e - t_n$	1 000	2 000	3 000	4 000	5 000	6 000	6 500	7 000
Ω	0.043 9	0.039 8	0.036	0.032 5	0.029 4	0.026 5	0.025 2	0.024
$t_e - t_n$	8 000	8 500	9 000	9 500	10 000	11 000	12 000	13 000
Ω	0.021 7	0.020 6	0.019 5	0.018 6	0.017 6	0.015 9	0.014 3	0.012 9

4　结束语

(1) 考虑右删失数据的改进 J-M 模型可充分挖掘可靠性测试停止时间(右删失数据)所包含的时间信息, 由此得出的评估结果较传统 J-M 模型更准确合理;

(2) 文中第 3 节从理论上严格证明了随着右删失时间数据的增加(即测试停止时间的延迟), 软件的残存缺陷比率和失效率评估值随之降低, 而可靠度评估值则随之增加, 与定性分析相吻合;

(3) 本文提出的基于残存缺陷比率、失效率以及可靠度函数的软件可靠性测试停止准则可有效指导右删失情况下的软件可靠性测试活动, 具有明显的工程意义;

(4) 若软件具有可信的历史失效数据, 则改进的 J-M 模型可以在无失效数据时进行近似的可靠性评估;

(5) 改进的 J-M 模型必须在(t_n, t_e)之间不发生失效时才适用, 若随着测试的进行, 出现第 $n+1$ 个失效, 则必须将$\{t_1, t_2, \cdots, t_n, t_{n+1}\}$作为最新的失效数据集, 进行后续的可靠性评估;

(6) 本文的推导过程及所有结论也适用于其它的二次项式指数类模型。可处理右删失数据的 NHPP 模型将是进一步的研究工作。

参考文献:

[1] 陈家鼎. 生存分析与可靠性[M]. 北京: 北京大学出版社, 2005.

[2] Cai Kaiyuan. Censored software-reliability Models[J]. *IEEE Trans. on Reliability*, 1997, 46(1): 69-75.

[3] 徐仁佐. 软件可靠性工程[M]. 北京: 清华大学出版社, 2007.

[4] 纪元, 白晓颖, 徐睿. 改进的Jelinski-Moranda 软件可靠性增长模型[J]. 清华大学学报(自然科学版), 2006, 46(10): 1759-1762.

[5] 宋晓秋. 适用于成组数据的 Jelinski-Moranda 模型研究[J]. 系统工程与电子技术, 1999, 21(7): 58-60. (Song Xiaoqiu. Analysis on Jelinski-Moranda model for group data [J]. *Systems Engineering and Electronics*, 1999, 21(7): 58-60.)

[6] Seo SunKeun. Estimation methods for the mean of the exponential distribution based on grouped & censored data[J]. *IEEE Trans. on Reliability*, 1993, 42(1): 87-96.

[7] Keiller. Investigating a specific class of software reliability growth models[C] // *Reliability and Maintainability Symposium*, 2002: 242-248.

[8] Han Fengyan. A censored nonparametric software reliability model[J]. *International Journal of Plant Engineering and Management*, 2006, 11(4): 227-233.

[9] Wilson Simon P. Nonparametric analysis of the order-statistic model in software reliability[J]. *IEEE Trans. on Software Engineering*, 2007, 33(3): 198-208.

[10] Musa J D. Software reliability: measurement, prediction, application[M]. *New York: McGraw Hill*, 1987.