基于 S 型测试工作量函数的软件可靠性增长模型

李秋英 李海峰 陆民燕 王学成

(北京航空航天大学 可靠性与系统工程学院, 北京 100191)

摘 要: 软件测试工作量随时间的变化情况对于软件可靠性增长曲线的形状具有显著影响,因此将测试工作量函数(TEF, Testing Effort Function)引入软件可靠性增长模型(SRGM, Software Reliability Growth Model)中以提高软件可靠性评估性能. 分析了软件结构特征及学习因素对实际测试过程的综合影响,研究增长速率为先增后减的 S 型增长趋势的 TEF,提出两种 S 型 TEF,即延迟 S 型 TEF 与变形 S 型 TEF,并提出考虑延迟 S 型 TEF 的 SRGM(DSTEF-SRGM)以及变形 S 型 TEF 的 SRGM(ISTEF-SRGM). 在两组真实失效数据集上,进行了这两种 SRGM 与经典 SRGM 及其它考虑 TEF 的 SRGM 的对比研究. 结果表明,ISTEF-SRGM 的模型拟合效果最优,同时验证了该模型具有优越的软件可靠性评估性能及模型适应性.

关 键 词: 软件可靠性; 测试; 可靠性分析; 增长; 模型

中图分类号: TP 311

文献标识码: A **文 章 编 号**: 1001-5965(2011)02-0149-06

Software reliability growth model with S-shaped testing effort function

Li Qiuying Li Haifeng Lu Minyan Wang Xuecheng

(School of Reliability and Systems Engineering, Beijing University of Aeronautics and Astronautics, Beijing 100191, China)

Abstract: The shape of the observed reliability growth curve depends strongly on the time distribution of the testing effort. Thus, the reliability estimation power of software reliability growth model (SRGM) can be improved by considering the testing effort function (TEF). Due to the integrated effects of software structure and learning factor on testing process, testing effort increasing rate may exhibit an S-shaped varying trend first increasing and then decreasing. To accurately describe this S-shaped varying trend, two S-shaped testing effort functions were proposed first, i.e. delayed S-shaped TEF and inflected S-shaped TEF, which were simple and flexible. Then two new SRGMs were put forward by combining two S-shaped TEFs into SRGM, i.e. DSTEF-SRGM and ISTEF-SRGM. Finally, for two real failure data-sets, the case study was done by comparing these two S-shaped TEF-SRGMs with several classical SRGMs and other representative SRGMs considering TEF in the form of estimation power. The result shows that compared with these comparison models, the proposed ISTEF-SRGM has the best fitting results on each data-set. In other words, the proposed ISTEF-SRGM provides novel applicability and a significantly better power of reliability estimation.

Key words: software reliability; testing; reliability analysis; growth; models

软件可靠性已成为软件发布时用户最为关心的指标之一[1]. SRGM(Software Reliability Growth Model)作为软件可靠性评估与预计的主要手段,从上世纪 70 年代起,就一直是软件可靠性工程领域最为关注的研究方向之一[2-3]. 虽然已有部分经典的 SRGM 被成功地应用于各类安全关键软件中[4-5],但若能在软件可靠性建模中考虑软件测试过程因素的影响,将会进一步提升现有模型的拟合与预计性能[6-8].

测试工作量被认为是一种重要的测试过程因素[6],是指在测试过程中所消耗的资源,可用人力、执行的测试用例数、CPU 时间等信息来度

收稿日期: 2010-07-13

作者简介: 李秋英(1973 -),女,黑龙江大庆人,讲师,li_qiuying@buaa.edu.cn.

量[9]. 与日历时间相比,测试工作量对缺陷探测效率的贡献更为直接[10],其随测试时间的变化情况对软件可靠性增长曲线的形状具有显著影响[11]. 早期 SRGM 或不考虑测试工作量,或假设测试工作量随时间的消耗率是常值[1,3,12]显然是不合理的. 如何建立准确定量描述测试工作量随测试时间变化情况的测试工作量函数(TEF, Testing Effort Function),进而建立考虑 TEF 的 SRGM 已成为软件可靠性建模研究的热点之一[6,9,11,13-22].

目前已有的 TEF 包括:指数、雷利、威布尔[13]、指数威布尔[14]、泛化指数[15]、Burr[16]、Logistic[6,9,11,23]、泛化 Logistic[17-18]、Log-Logistic[19]及 Chatterjee 学习函数模型[20],在此基础上开展的考虑 TEF 的软件可靠性建模研究包括:①结合指数型非齐次泊松过程类(NHPP, Non-Homogeneous Poisson Process)的 SRGM[9,11,13-20];②结合 S 型 NHPP 类的 SRGM[6,21-22];③考虑变点[23-26]、排错情况[21,27-28]及失效类型[29]等因素的扩展模型. 其中,第 1 种类型的模型最为常见且是其它两类模型的研究基础,因此,本文的研究内容与实例验证均围绕这种结合指数型 NHPP 类 SRGM 展开,暂不考虑与后两类模型的对比研究.

实际测试过程经常会出现如下情形:初始时测试工作量快速增长,但进行到某一时间点后,由于软件结构以及学习因素等综合影响,测试工作量的增长速率会逐渐变慢并趋于平稳,因此测试工作量随时间的增长速率会呈现出先增后减的 S 型增长趋势[11]. 此时,指数类、威布尔类等 TEF 显然不再适合. Logistic TEF[11]虽然也是一种 S 型 TEF,但其初始测试工作量不为 0,与直觉不符;且仅有这一种 S 型 TEF 对于 TEF 及其建模研究是远远不够的.

延迟 S 型函数[30]与变形 S 型函数[31]是两种经典的 S 型函数,适于描述 S 型增长趋势,且已被应用于软件可靠性建模研究[30-31]. 本文首先提出两种 S 型 TEF,然后提出两种考虑 S 型 TEF 的指数型 NHPP 类 SRGM(STEF-SRGM),最后将这两种 STEF-SRGMs 与若干传统 NHPP 类 SRGMs 及其它典型的考虑 TEF 的指数型 NHPP 类 SRGMs 进行对比,以验证 STEF-SRGMs 在模型评估性能上的有效性与适用性.

1 TEF 概述

TEF 不但可用于描述测试工作量随时间的变化情况,也是建立考虑 TEF 的 SRGM 的前提与基础. 本节首先介绍现有的 TEF(如表 1 所示). 其中,$W(t)$表示时间间隔$[0,t]$内的累积测试工作量,$W_{\max}$表示最终消耗的测试工作量,即 $W_{\max}=W(\infty)$,$w(t)$是测试工作量在时刻 t 的增长速率函数,$f(t)$是学习因素函数(t 的增函数).

对表 1 中的 TEF,补充说明如下:

①虽然威布尔类 TEF(表 1 的 1~6)具有形式灵活等优点,但其无法很好地描述 S 型增长趋势,且当形状参数 $m>3$ 时,威布尔类型的函数会有较明显的峰值现象[11];②Logistic 类 TEF(表 1 的 7~8)虽然为 S 型 TEF,但当 $t=0$ 时,计算出的初始测试工作量 $W(0)$不为 0,这一结论与直觉不相符;③Chatterjee 提出的考虑学习函数的 TEF 的形式较复杂,且与 Log-logistic TEF 的表现相同,即基于这两种 TEF 的 SRGM 的评估性能一般,因此在相关研究中未得到关注.

表 1 已有的测试工作量函数

序号	TEF 名称	测试工作量函数 $W(t)$
1	指数	$W_{\max}(1-\exp(-bt))$
2	雷利	$W_{\max}(1-\exp(-(b/2)t^2))$
3	威布尔	$W_{\max}(1-\exp(-bt^m))$
4	泛化指数	$W_{\max}(1-\exp(-bt))^\theta$
5	Burr	$W_{\max}(1-\exp(-bt^2))^\theta$
6	指数威布尔	$W_{\max}(1-\exp(-bt^m))^\theta$
7	Logistic	$W_{\max}/(1+A\exp(-\alpha t))$
8	泛化 Logistic	$W_{\max}/\sqrt[k]{1+A\exp(-akt)}$
9	Chatterjee	$w(t)=W'(t)=k/(f(t))^p$
10	Log-Logistic	$W_{\max}((bt)^c/(1+(bt)^c))$

2 考虑 S 型 TEF 的 SRGM

本节首先提出两种 S 型 TEF,在此基础上提出两种考虑 S 型 TEF 的 NHPP 类 SRGM.

2.1 S 型 TEF

2.1.1 延迟 S 型 TEF

利用延迟 S 型函数来描述测试工作量随时间的变化趋势,则$[0,t]$内的累积测试工作量为

$$W(t)=W_{\max}(1-(1+bt)\exp(-bt)) \quad (1)$$

其中 b 为测试工作量消耗率,且 $b>0$.

由式(1)可知:①$W(0)=0$,即初始测试工作量为 0;②$W(t)$为非负值且随时间 t 单调递增,这与测试过程中测试工作量的不断增长相符合;③将$W(t)$对时间 t 求导,可得 TEF 随时间的变化速率函数为

$$w(t)=\frac{\mathrm{d}W(t)}{\mathrm{d}t}=W_{\max}b^2t\exp(-bt) \quad (2)$$

其中,$w(t)$是一个先增后减的函数,在 $t=1/b$ 时,测试工作量的增长速率达到最大值.

2.1.2　变形 S 型 TEF

利用变形 S 型函数描述测试工作量随时间的变化形式,则$[0,t]$内的累积测试工作量为

$$W(t) = W_{\max}\frac{1-\mathrm{e}^{-bt}}{1+\psi\mathrm{e}^{-bt}} \tag{3}$$

其中 ψ 是常量.

由式(3)可知:①$W(0)=0$;②$W(t)$为非负值且随时间 t 单调递增;③增长速率函数为

$$w(t) = \frac{\mathrm{d}W(t)}{\mathrm{d}t} = \frac{Nb\exp(-bt)(1+\psi)}{(1+\psi\exp(-bt))^2} \tag{4}$$

其中,$w(t)$是先增后减的函数,在 $t=\ln\psi/b$ 时,测试工作量的增长速率达到最大值.

2.2　考虑 S 型 TEF 的 SRGM(STEF-SRGM)

STEF-SRGM 的模型假设如下[6,9,11,14]:

①软件缺陷探测过程服从 NHPP 过程,且软件中残存的缺陷总数是一个定值,记为 N;②当前软件失效是由残存缺陷引发的;③在时间间隔$[t, t+\Delta t]$内探测到的缺陷数均值与残存缺陷数成正比,该比值表示测试过程中的缺陷探测率;④利用延迟或变形 S 型 TEF 来描述测试工作量随时间的消耗情况;⑤缺陷一旦被发现,立即被排除且不引入新错.

根据上述假设可得考虑 TEF 的指数型 NHPP 类 SRGM 满足下式[9]:

$$\frac{\mathrm{d}m(t)}{\mathrm{d}t}\frac{1}{w(t)} = r[N-m(t)] \tag{5}$$

其中,$m(t)$是$[0,t]$时间段的探测缺陷数的期望值;r 表示每个缺陷被检测到的概率.

在边界条件 $m(0)=0$ 下,对式(5)求解可得:

$$m(t) = N\{1-\exp[-r(W(t)-W(0))]\} \tag{6}$$

由于 $W(0)=0$,式(6)可以简化为

$$m(t) = N\{1-\exp[-rW(t)]\} \tag{7}$$

由式(7)可得

$$W(t) = \frac{1}{r}\ln\left[\frac{N}{N-m(t)}\right] \tag{8}$$

式(7)、式(8)可以定量地反映缺陷探测数均值与所消耗的测试工作量之间的关系. 根据式(7),结合式(1)、式(3)即可获得如下两种考虑 S 型 TEF 的指数型 NHPP 类 SRGMs.

2.2.1　考虑延迟 S 型 TEF 的 SRGM(DSTEF-SRGM)

将式(1)代入式(7),可以得到考虑延迟 S 型 TEF 的 NHPP 类 SRGM:

$$m(t) = N\{1-\exp[-rW_{\max}(1-(1+bt)\exp(-bt))]\} \tag{9}$$

由式(9)可得失效强度函数和最终软件残存的缺陷数:

$$\lambda(t) = b^2NrW_{\max}\exp\{rw[\mathrm{e}^{-bt}(bt+1)]-bt\} \tag{10}$$

$$m_{\mathrm{rem}} = N-m(\infty) = N\exp(-rW_{\max}) \tag{11}$$

2.2.2　考虑变形 S 型 TEF 的 SRGM(ISTEF-SRGM)

将式(3)代入式(7),得到考虑变形 S 型 TEF 的 NHPP 类 SRGM:

$$m(t) = N\left\{1-\exp\left[-rW_{\max}\frac{1-\exp(-bt)}{1+\psi\exp(-bt)}\right]\right\} \tag{12}$$

由式(12)可得失效强度函数和最终软件残存的缺陷数:

$$\lambda(t) = NbrW_{\max}\frac{k+1}{[k\exp(-bt)+1]^2}\cdot \exp\left\{\frac{rW_{\max}[\exp(-bt)-1]}{\psi\exp(-bt)+1}-bt\right\} \tag{13}$$

$$m_{\mathrm{rem}} = N-m(\infty) = N\exp(-rW_{\max}) \tag{14}$$

3　实例验证

3.1　实例概述

1) 选取 12 个 NHPP 类 SRGMs 作为对比模型,其中前 9 个模型为经常用于软件可靠性建模验证研究的经典 NHPP 类 SRGMs[1,3],后 3 个模型为目前最为典型的 TEF-SRGMs,详见表 2.

2) 选择两组真实的失效数据集 Ohba 与 Wood[6],用于估计模型的参数值并计算模型的拟合结果. 这两组数据集均是用于软件可靠性模型性能比较的经典范例,均包含 3 类数据,即测试时间(t),测试工作量(W)以及探测缺陷数(m). 计算过程中,本文将 Wood 数据集中的测试工作量数据缩小 100 倍,目的是为了提高模型参数 $W_{\max}$ 的估计质量,且这种缩放处理不会对模型的可靠性评估性能对比产生影响.

3) 选用最小二乘估计(LSE, Least Square Estimation)作为模型参数估计方法,LSE 产生的估计结果具有无偏性[32].

4) 选用 3 种经典评价准则作为模型拟合结果的评价准则,分别为均值误差平方和[6](见式(15))、回归曲线方程的相关指数[8](见式(16))、相对误差图[6](RE 图,见式(17)).

$$M_{SE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_i')^2 \tag{15}$$

其中,y_i为真实的观测数据,y_i'为模型计算出的对应 y_i的拟合值. M_{SE}值越小,则模型拟合效果越好,也即表明模型的可靠性评估性能越好.

$$R_{Sq} = 1 - \frac{\sum_{i=1}^{n}(y_i - y_i')^2}{\sum_{i=1}^{n}(y_i - y_{ave})^2} \tag{16}$$

其中,y_{ave}表示观测数据 y_i的均值. R_{Sq}值越接近于1,则模型拟合效果越好.

RE 图由模型在每个观测数据点上计算出的 R_E 值(式(17))绘制而成.

$$R_E(i) = \frac{y_i' - y_i}{y_i} \tag{17}$$

RE 图整体越接近于 x 轴,模型拟合效果越好.

3.2 实例计算结果与分析

表 2 分别列出各模型在两组失效数据集上的参数估计值和模型拟合结果,具体分析如下.

表 2 STEF-SRGMs 与对比 SRGMs 在各组失效数据集上的参数估计与模型拟合结果

序号	SRGM 名称	探测缺陷均值函数 $m(t)$	Ohba(模型参数估计值)		Wood(模型参数估计值)	
			M_{SE}	R_{Sq}	M_{SE}	R_{Sq}
1	GO[3]	$N[1-e^{-rt}]$	$N=760.53, r=0.0323$		$N=130.2, r=0.0832$	
			139.82	0.986	11.62	0.913
2	DS[3]	$N[1-(1+rt)e^{-rt}]$	$N=374.05, r=0.197$		$N=103.98, r=0.265$	
			168.67	0.984	25.26	0.969
3	IS[3]	$N\frac{1-e^{-rt}}{1+\psi e^{-rt}}$	$N=374.39, r=0.179, \psi=3.01$		$N=110.83, r=0.172, \psi=1.2$	
			127.31	0.992	8.98	0.989
4	YW[33]	$N\{1-\exp[-r\alpha(1-\exp(-\beta t^{\gamma}))]\}$	$N=460.81, r=0.0272,$ $\alpha=49.63, \beta=0.00708, \gamma=1.98$		$N=135.74, r=0.00014,$ $\alpha=11711, \beta=0.0235, \gamma=1.46$	
			260.04	– – –	16.59	0.987
5	YR[33]	$N\{1-\exp[-r\alpha(1-\exp(-\beta t^2/2))]\}$	$N=459.09, r=0.0273,$ $\alpha=49.32, \beta=0.0137$		$N=110.61, r=0.000226,$ $\alpha=10534, \beta=0.0149$	
			268.42	0.733	39.7	0.951
6	GGO[6]	$N[1-e^{-rt^c}]$	$N=427.83, r=0.0361, c=1.28$		$N=118.56, r=0.0765, c=1.11$	
			102.12	0.990	10.87	0.987
7	YID1[33]	$N\frac{b}{a+b}(\exp(at)-\exp(-bt))$	$N=406, b=0.0623, a=0.0184$		$N=121, b=0.0912, a=0.0031$	
			154.82	0.986	12.08	0.986
8	YID2[33]	$N(1-\exp(-bt))(1-a/b)+aNt$	$N=374, b=0.0815, a=0.0021$		$N=112, b=0.0909, a=0.0042$	
			565.5	0.986	36.9	0.986
9	OID[6]	$[1-e^{(-r(1-b)t)}]N/(1-b)$	$N=371.65, r=0.066, b=0.511$		$N=129.39, r=0.0837, b=0.00626$	
			139.82	0.986	11.62	0.986
10	威布尔-TEF[13]	$N(1-\exp(-rW_{max}(1-\exp(-bt^m))))$	$N=477.1, r=0.03245$ $W_{max}=792.5, b=0.0023,$ $m=1.115$		$N=135.7, r=0.01421$ $W_{max}=117.1, b=0.00235,$ $m=1.46$	
			106.26	0.990	16.58	0.980
11	Logistic-TEF[11]	$N\left(1-\exp\left(-r\left(\frac{W_{max}}{1+A\exp(-\alpha t)}-\frac{W_{max}}{1+A}\right)\right)\right)$	$N=395.6, r=0.04164$ $W_{max}=54.85, A=13.03,$ $a=0.2263$		$N=112.1, r=0.02409$ $W_{max}=100, A=13.15,$ $a=0.2866$	
			114.05	0.989	21.44	0.974
12	指数威布尔-TEF[14]	$N(1-\exp(-rW_{max}(1-\exp(-bt^m))^{\theta}))$	$N=575.7, r=0.01916, m=0.4619$ $W_{max}=886, b=0.1111, \theta=2.79$		$N=107.1, r=0.0142, m=2.283$ $W_{max}=107.1, b=0.00211, \theta=0.56$	
			113.10	0.990	13.31	0.984
13	延迟 S 型 TEF(DSTEF-SRGM)	$N(1-\exp(-rW_{max}(1-(1+bt)\exp(-bt))))$	$N=513.4, r=0.02274$ $W_{max}=64.62, b=0.132$		$N=132, r=0.01497$ $W_{max}=114.7, b=0.171$	
			178.79	0.983	25.99	0.968
14	变形 S 型 TEF(ISTEF-SRGM)	$N\left(1-\exp\left(-rW_{max}\frac{1-\exp(-bt)}{1+\psi\exp(-bt)}\right)\right)$	$N=357.6, r=0.138, W_{max}=60.08$ $\psi=19.84, b=0.1278$		$N=105.3, r=0.1069,$ $b=0.09223$ $W_{max}=131.6, \psi=14.6$	
			74.05	**0.993**	**7.88**	**0.990**

注:①加黑的数字为该列中的最佳结果;② – – – 为该组数据下的计标结果不合理或远差于同列数据.

1) Ohba 数据集上的拟合结果. ①在 Ohba 数据集上,本文提出的 ISTEF-SRGM 的拟合结果为最优(M_{SE}值最小,为 74.05;R_{Sq}值最接近于 1,为 0.993),且显著好于其它对比模型;但 DSTEF-SRGM 的拟合效果表现一般. ②拟合效果最好的前 5 个模型依次为:ISTEF-SRGM、GGO、威布尔-TEF、指数威布尔-TEF、Logistic-TEF. ③图 1 描述了这 5 个拟合效果最优模型对 Ohba 数据集中每个探测缺陷数均值的拟合值与实际观测值的对比情形,可看出,ISTEF-SRGM 的拟合图形与实际观测值图形的整体增长趋势最为相似,且在许多点上的距离也最为接近. ④图 2 是根据 5 个拟合效果最优模型在 Ohba 数据集中每个数据点上的 R_E 值绘制得到的模型 RE 图. 可看出,ISTEF-SRGM 的 RE 图整体上最接近于 x 轴,且这 5 个模型在多数点上的拟合结果略偏于保守.

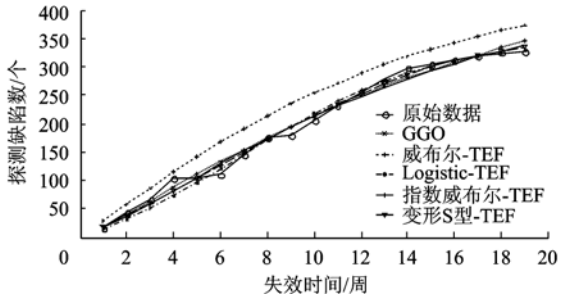


图 1　Ohba 数据集上的拟合结果

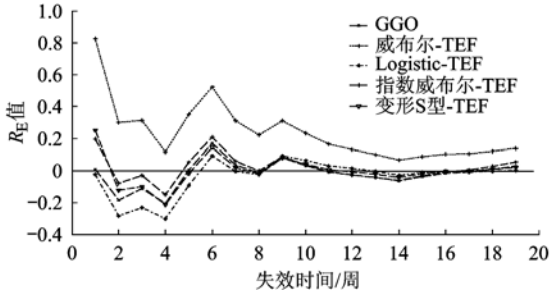


图 2　Ohba 数据集上的 RE 图

2) Wood 数据集上的拟合结果. ①在 Wood 数据集上,本文提出的 ISTEF-SRGM 的拟合效果依然为最优(M_{SE}值最小,为 7.88;R_{Sq}值最接近于 1,为 0.990),且显著好于其它对比模型. 但 DSTEF-SRGM 的拟合效果依然表现一般. ②拟合效果最好的前 5 个模型依次为:ISTEF-SRGM、IS、GGO、GO 和 OID. ③图 3 描述了这 5 个拟合效果最优模型对 Wood 数据集中每个探测缺陷数均值的拟合值与实际观测值的对比情形. 可看出,ISTEF-SRGM 的拟合图形与实际观测值图形的整体增长趋势最为相似且在许多点上的距离也最为接近. ④图 4 是根据 5 个拟合效果最优模型在 Wood 数据集中每个数据点上的 R_E 值绘制得到的模型 RE 图. 可看出,ISTEF-SRGM 在 Wood 数据集上的 RE 图整体上最接近于 x 轴,且这 5 个模型在多数点上的拟合结果略偏于乐观.

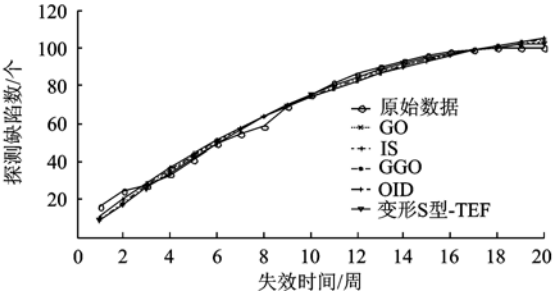


图 3　Wood 数据集上的拟合结果

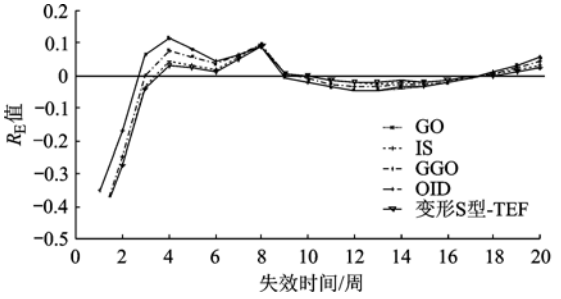


图 4　Wood 数据集上的 RE 图

3) 实例计算结果分析. 综上,可得出如下结论:①本文提出的考虑变形 S 型 TEF 的 SRGM 在两组数据集上的拟合效果均是最优的,显著优于绝大部分对比模型,从而证明 ISTEF-SRGM 在模型评估性能上具有很好的有效性及适用性;②对于 Ohba 数据集,拟合效果最优的 5 个模型中有 4 个是 TEF-SRGM,所以在软件可靠性建模中考虑 TEF,确实可以有效地改进传统模型的评估性能;③ISTEF-SRGM 的评估性能显著优于 2 种最典型的 TEF-SRGM,即 Logistic-TEF 和指数威布尔 TEF 模型,从而表明了本文研究成果具有显著的理论研究意义;④本文提出的考虑延迟 S 型 TEF 的 SRGM(DSTEF-SRGM)在两组数据集上的拟合效果均表现一般. 分析原因,可能是因为延迟 S 型 TEF 并不适合用于描述测试工作量随时间的增长趋势、进而影响了模型的评估性能. 因此,选择合适的 TEF 对于保证 TEF-SRGM 的可靠性评估性能具有重要意义.

4　结 束 语

本文提出两种 S 型 TEF,即延迟 S 型与变形 S 型 TEF,在此基础上提出两种考虑 S 型 TEF 的 SRGMs,并在两组经典失效数据集上,将这 2 种 STEF-SRGMs 与若干经典 SRGMs 及典型的 TEF-

SRGMs进行了对比研究.结果表明,本文提出的考虑变形S型TEF的SRGM的可靠性评估性能及模型适用性显著有效,具有良好的理论研究意义与工程应用前景.

在未来的研究工作中,将关注以下问题:①将变形S型TEF与S型NHPP类软件可靠性模型相结合;②在建模过程中考虑不完美排错的情况;③与基于测试工作量函数的S型NHPP类软件可靠性模型进行对比研究.

参考文献(References)

[1] Lyu M R. Handbook of software reliability engineering [M]. New York: McGraw Hill, 1996

[2] Gokhale S S, Lyu M R. A simulation approach to structure-based software reliability analysis [J]. IEEE Trans on Software Engineering, 2005, 31(8): 643 – 656

[3] Pham H. Software reliability [M]. Singapore: Springer-Verlag, 2000

[4] Gana A, Huang S T. Statistical modeling applied to managing global 5ess-2000 switch software development [J]. Bell Labs Technical Journal, 1997, 2(1): 144 – 153

[5] Kruger G A. Validation and further application of software reliability growth models [J]. Hewlett-Packard Journal, 1989, 40(4): 75 – 79

[6] Huang C Y, Kuo S Y, Michael R L. An assessment of testing-effort dependent software reliability growth models [J]. IEEE Transactions on Reliability, 2007, 56(2): 198 – 211

[7] Shibata K, Rinsaka K, Dohi T. Metrics-based software reliabil-ity models using non-homogeneous Poisson processes [C] // Proc of the 17th IEEE International Symposium on Software Reliability Engineering. Washington DC: IEEE Computer Society, 2006: 52 – 61

[8] 谢景燕,安金霞,朱纪洪.考虑不完美排错情况的NHPP类软件可靠性增长模型[J].软件学报,2010,21(5):942 – 949
Xie J Y, An J X, Zhu J H. NHPP software reliability growth model considering imperfect debugging [J]. Journal of Software, 2010, 21(5): 942 – 949 (in Chinese)

[9] Kuo S Y, Huang C Y, Michael R L. Framework for modeling software reliability, using various testing-efforts and fault-detection rates [J]. IEEE Transactions on Reliability, 2001, 50(3): 310 – 320

[10] Norman E F, Martin N. A critique of software defect prediction models [J]. IEEE Transactions on SoftwareEngineering. 1999, 25(5): 675 – 689

[11] Huang C Y. Analysis of incorporating logistic testing-effort function into software reliability modeling [J]. IEEE Transactions on Reliability, 2002, 51(3): 261 – 270

[12] Musa J D, Iannino A. Software reliability, measurement, prediction and application [M]. New York: McGraw Hill, 1987

[13] Yamada S, Hishitani J, Osaki S. Software reliability growth model with Weibull testing effort: a model and application [J]. IEEE Transactions on Reliability, 1993, 42(1): 100 – 106

[14] Ahmad N, Bokhari M U, Quadri S M K, et al. The exponentiated Weibull software reliability growth model with various testing-efforts and optimal release policy [J]. International Journal of Quality & Reliability Management, 2008, 25(2): 211 – 235

[15] Quadri S M K. Non homogeneous Poisson process software reliability growth model with generalized exponential testing effort function [J]. RAU Journal of Research, 2006, 16(2): 159 – 163

[16] Ahmad N, Khan M G M, Quadri S M K, et al. Modeling and analysis of software reliability with Burr type X testing-effort and release-time determination [J]. Journal of Modeling in Management, 2009, 4(1): 28 – 54

[17] Huang C Y. Cost-reliability-optimal release policy for software reliability models incorporating improvements in testing efficiency [J]. The Journal of Systems and Software, 2005, 77: 139 – 155

[18] Huang C Y. Optimal release time for software systems considering cost, testing-effort, and test efficiency [J]. IEEE Transactions on Reliability, 2005, 54(4): 583 – 591

[19] Bokhari M U, Ahmad N. Analysis of a software reliability growth models: the case of log-logistic test-effort function [C]//Proceedings of the 17th International Conference on Modeling and Simulation. Montreal, Canada: Acta Press, 2006: 540 – 545

[20] Chatterjee S, Misra R B, Alam S S. Joint effect of test effort and learning factor on software reliability and optimal release policy [J]. International Journal of System Science, 1997, 28(4): 391 – 396

[21] Ahmad N, Khan M G M. A study of testing-effort dependent inflection S-shaped software reliability growth models with imperfect debugging [J]. International Journal of Quality & Reliability Management, 2010, 27(1): 89 – 110

[22] Jha P C, Gupta D, Yang B, et al. Optimal testing resource allocation during module testing considering cost, testing effort and reliability [J]. Computers & Industrial Engineering, 2009, 57: 1122 – 1130

[23] Lin C T, Huang C Y. Enhancing and measuring the predictive capabilities of testing-effort dependent software reliability models [J]. The Journal of Systems and Software, 2008, 81: 1025 – 1038

[24] Huang C Y. Performance analysis of software reliability growth models with testing-effort and change-point [J]. The Journal of Systems and Software, 2005, 76: 181 – 194

[25] Kapur P K, Gupta A. Testing effort control using flexible software reliability growth model with change point [J]. International Journal of Performability Engineering, 2006, 2(3): 245 – 262

[26] Lin C T, Huang C Y, Chang J R. Integrating generalized Weibull-type testing-effort function and multiple change-points into software reliability growth models [C]//12th Asia-Pacific Software Engineering Conference. Los Alamitos: IEEE Computer Society, 2005: 431 – 438

(下转第160页)

其中 $\boldsymbol{W}_0$ 为待判样本数据 $\boldsymbol{U}_0$ 在多维状态空间中的表示.

假定 $\boldsymbol{U}_0 = [17.0, -0.20, -0.70, 55.0, 2.80, 2.30, 8.50, -0.30, 6.80, 0.90, -2.80, -4.0, 48.0, 0]^{\mathrm{T}}$,代入式(2)求得 $\boldsymbol{W}_0$,再将 $\boldsymbol{W}_0$ 代入式(3),计算得到 $\begin{bmatrix} f \\ s \end{bmatrix} = \begin{bmatrix} 94.3355 \\ 91.3168 \end{bmatrix}$. 根据判别分析的原理[10],比较 s 和 f 两类计算结果,将待判样本划分到结果值较大的一类即“挂索失败”类别中去,由前述模型验证结果可知,该分类结果的可信度在 97.3% 以上.

3 结 束 语

本文将多元统计方法和舰载机着舰过程的工程机理分析相结合,提出了从非线性相关的多变量统计数据中提炼舰载机着舰过程特征并以此为基础进行机-舰系统安全状态监控和判断的方法. 该方法具有一定的通用性,可推广用于其他产品研制和使用过程的监控及分析.

本文以美军舰载机着舰数据为例,确定了着舰过程多变量对应的安全状态空间,包括空间的维数和空间的工程含义. 从案例中着舰安全多维状态空间的分析计算结果与系统真实状态的对比来看,基于主元分析的安全特征提取及判别算法对于舰载机着舰这样的非线性过程系统的描述准确性较高,兼具较强的工程实用性.

参考文献 (References)

[1] 史青海. 舰载机着舰控制技术研究[D]. 哈尔滨:哈尔滨工程大学自动化学院,2006
Shi Qinghai. Study on control technology for carrier aircraft landing research on carrier landing technology[D]. Harbin: College of Automation, Harbin Engineering University, 2006 (in Chinese)

[2] Chief of NAVAL operations CV NATOPS manual[R]. NAVAIR 00-80T-105, 0800LP0291740, 1999

[3] Thomas Rudowsy, Stephen Cook, Melvin Luter, et al. Review of the carrier approach criteria for carrier-based aircraft phase I: final report[R]. NAWCADPAX/TR-2002/71, 2002

[4] Aircraft Carriers-CV/CVN [EB/OL]. [2009-12-23]. http://navysite.de/carriers.htm

[5] Tripathi S M. Modern control systems: an introduction[M]. Hingham, MA: Infinity Science Press, 2008: 2 - 57

[6] Zdzislaw Bubnicki. Modern control theory[M]. Berlin: Springer, 2005: 17 - 37

[7] 戎梅. 系统事故推演建模与分析方法研究[D]. 北京:北京航空航天大学工程系统工程系,2009
Rong Mei. Research on system accident rehearsal modeling and analysis method[D]. Beijing: Deportment of System Engineering of Engineering Technology, Beijing University of Aeronautics and Astron autics, 2009 (in Chinese)

[8] Richard P Micklos. Carrier landing parameters from survey 45, fleet and training command aircraft landing aboard USS enterprise CVN-65 (main text and appendix A) [R]. AD-A258873, 1991

[9] 张立军,任英华. 多元统计分析实验[M]. 北京:中国统计出版社,2009: 72 - 90
Zhang Lijun, Ren Yinghua. Multivariate statistical analysis experiment[M]. Beijing: China Statistics Press, 2009: 72 - 90 (in Chinese)

[10] 何晓群. 多元统计分析[M]. 北京:中国人民大学出版社,2004: 98 - 113
He Xiaoqun. Multivariate statistical analysis[M]. Beijing: China Renmin University, 2004: 98 - 113 (in Chinese)

(编 辑:董立君)

(上接第 154 页)

[27] Kapur P K, Goswami D N, Amit B, et al. Flexible software reliability growth model with testing effort dependent learning process[J]. Applied Mathematical Modeling, 2008, 32: 1298 - 1307

[28] Kapur P K, Grover P S. Modelling an imperfect debugging phenomenon with testing effort [C]//Proceedings of the 4th Software Reliability Engineering. Los Alamitos: IEEE, 1994: 178 - 183

[29] Kapur P K, Omar S, Anu G A, et al. Unified framework for developing testing effort dependent software reliability growth models [J]. WSEAS Transactions on Systems, 2009, 4 (8): 521 - 531

[30] Yamada S, Ohba M, Osaki S. S-shaped software reliability growth models and their applications [J]. IEEE Trans on Reliability, 1984, R-33 (4): 289 - 292

[31] Ohba M. Inflection S-shaped software reliability growth models [M]. Stochastic Models in Reliability Theory. Berlin: Springer, 1984: 144 - 162

[32] Su Y S, Huang C Y. Neural-network-based approaches for software reliability estimation using dynamic weighted combinational models[J]. The Journal of Systems and Software, 2007, 80: 606 - 615

[33] Yamada S, Tokuno K, Osaki S. Imperfect debugging models with fault introduction rate for software reliability assessment[J]. International Journal of Systems Science, 1992, 23 (12): 2241 - 2252

(编 辑:娄 嘉)