软件可靠度的模糊故障树评定方法*

刘博宁[1]，张　鹏[1]，张建业[2]，马秋芝[3]

(1. 空军工程大学 工程学院，西安 710038；2. 空军工程大学 科研部，西安 710051；3. 西安科技大学 通信工程学院，西安 710054)

摘　要：软件可靠性的定量评价是软件可靠性工程的关键问题之一，采用故障树方法对软件进行定性和定量分析，提出了两类情况下对影响软件可靠性的主次因素划分及其模糊权重的计算方法。在此基础上，建立多级模糊评价模型，提出了增广和聚合算法，并给出了软件可靠度算式。选择某型航空装备软件进行了测试实例分析，实验结果表明了该方法评价结构的合理性与评价算法的有效性，适用于软件质量及开发过程控制的工程实践。

关键词：软件；可靠度；故障树；模糊评定

中图分类号：TP311.5　　**文献标志码**：A　　**文章编号**：1001-3695(2012)10-3783-04

doi:10.3969/j.issn.1001-3695.2012.10.046

Method for software reliability assessment based on fuzzy fault tree analysis

LIU Bo-ning[1], ZHANG Peng[1], ZHANG Jian-ye[2], MA Qiu-zhi[3]

(1. *Engineering Institute, Air Force Engineering University, Xi'an* 710038, *China*; 2. *Dept. of Scientific Research, Air Force Engineering University, Xi'an* 710051, *China*; 3. *College of Communication Engineering, Xi'an University of Science & Technology, Xi'an* 710054, *China*)

Abstract: The quantitative assessment of software reliability is one of the most important problem for software reliability project, this paper used firstly the qualitative and quantitative analyzing based on fault tree analysis of software, it proposed the method for fuzzy weight calculating and the primary and secondary ingredients compartmentalizing. It established the multi-grade fuzzy assessment model, discussed the enhanced and converging arithmetic, and proposed the expressions of reliability. This method presented its rationality of assessment configuration and validity of assessment arithmetic through testing and analyzing of an aero equipment's software, it could be used on quality and exploitation control engineering for software.

Key words: software; reliability; fault tree ; fuzzy assessment

0　引言

软件可靠性评估是软件可靠性工程的核心内容，可靠度是衡量软件可靠性的关键参数，表示软件在规定时间内不失效的概率。武器装备软件由于其用途和使用环境的特殊性，基本是使命关键软件(mission-critical software, MCS)或安全关键软件(safety-critical software, SCS)[1]，其可靠性对于满足作战需要、提高作战效能的重要意义不可忽视。国内外学者在软件可靠性评估方面作了许多有益研究，国外经典的可靠性评估模型有 Jelinski-Moranda 模型[2]、LittlewoodVerrall 模型[3]、Goel-Okumoto IDM 和 NHPP 模型[4,5]等；国内学者蔡开元[6]用模糊方法研究了软件可靠性，提出了经典的 Cai 模糊增长和确定模型，楼俊钢等人[7]建立了考虑软件不同失效过程偏差的软件可靠性模型，李海峰等人[8]提出了一种 NHPP 模型与 Logistic 测试覆盖函数相结合的可靠性增长模型，姚金江等人[9]提出了一种改进的 Jelinski-Moranda 模型，张家海等人[10]建立了基于神经网络的组合导航软件可靠性评估模型，石柱等人[11]从不同角度对航天软件开发的整个生命周期的可靠性进行了度量和评价，并有效应用于航天软件的可靠性评估。

本文在此基础上进行了深入研究，通过分析软件控制流程中各子程序模块的故障因果属性，建立了软件故障树，提出以可靠性测试过程中软件各最小模块单元失效数据为依据，从两个方面对软件可靠性进行评估并对可靠度进行评定：一方面，提出将故障因素(底事件)划分为主导、次要因素集，计算主导因素的关键重要度；另一方面，提出增广评价集和聚合向量的概念，在故障树分析的基础上，建立多级模糊评价模型，对软件可靠性进行定量分析，并对软件可靠度进行评定。相对于传统方法而言，该方法与软件测试的实际工作流程结合更加紧密，充分融合了故障树与模糊评价方法在逻辑分析与定量评价方面的优势，评价结构更趋合理，具有更强的工程实用性。

1　软件故障树分析

1.1　软件故障树

故障树分析方法是将系统故障的各种原因(包括硬件、环境、人为因素)由总体至部分按树枝状结构自上而下逐层细化

收稿日期：2012-02-26；**修回日期**：2012-04-13　　**基金项目**：总装备部预研基金资助项目(9140A2XXXX311JB32)；中国博士后科学基金资助项目(201150M1551)

作者简介：刘博宁(1986-)，男，陕西西安人，硕士研究生，主要研究方向为检测技术、软件安全性与风险控制方法(mycelebration@sina.com)；张鹏(1979-)，男，山西太原人，讲师，主要研究方向为软件测试、数据智能处理与挖掘；张建业(1971-)，男，山西忻州人，副教授，硕导，主要研究方向为信息融合、数据挖掘、软件测试；马秋芝(1983-)，女，陕西西安人，硕士研究生，主要研究方向为电力电子与电源 FPGA 设计.

的分析方法[12]。故障树将系统中事件分层为顶事件 T、中间事件 E_i、底事件 $x_i(i=1,2,\cdots,n)$，并用逻辑门将其连接起来，以表达不同事件之间的因果关系。通过对事件间逻辑关系的定性和定量描述，达到对系统分析的目的。图 1 是某 ATM 机的取款系统软件故障树简化示意图[13]。

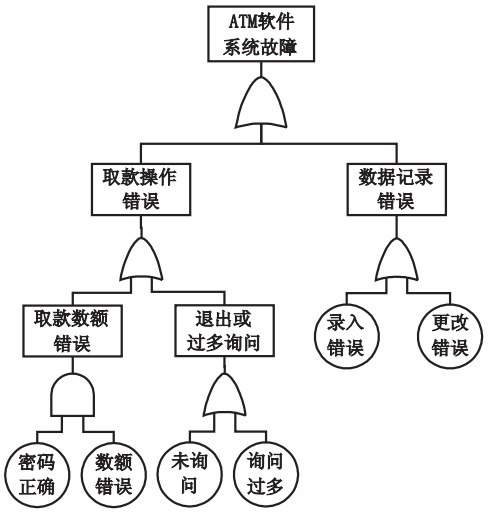

图1 某ATM机的取款系统软件故障树简化示意图

软件工程应用中，常用故障树分析方法对软件故障模式进行测试用例的设计，既可以提高测试的充分性，又能减少测试用例的数量。刘文红等人[14]运用故障树方法获得最小割集并用做软件测试用例设计的依据；徐中伟等人[15]建立了形式化故障树分析模型，并用于软件安全性测试，均取得了良好测试效果。

本文通过分析软件控制流程，以软件故障为顶事件 T，控制流程中间子程序模块故障为中间事件 E_i，最小模块单元失效为底事件 $x_i(i=1,2,\cdots,n)$，将各模块单元的故障因果关联用逻辑门连接，建立软件故障树，对软件可靠度进行评定。

1.2 定量分析

设顶事件 T 发生概率为 $P(T)$，底事件 x_i 发生概率为 $P(x_i)$，软件故障树有 k 个最小割集 $C_i(1\leqslant i\leqslant k)$，则故障树结构函数表示为最小割集的和事件：

$$T=C_1+C_2+\cdots+C_k \tag{1}$$

其中：C_i 表示为其包含的底事件的积事件：

$$C_i=x_1x_2\cdots x_n \tag{2}$$

失效强度是指单位时间内软件发生失效的概率，可用失效强度表示底事件 x_i 发生概率：

$$P(x_i)=\frac{N}{t} \tag{3}$$

其中：N 表示实测软件故障数，t 表示测试时间。

一般情况下，最小割集彼此相交，可知：

$$P(T)=P(C_1+C_2+\cdots+C_k)=\sum_{i=1}^{k}P(C_i)-\sum_{i<j=2}^{k}P(C_iC_j)+\sum_{i<j<l=3}^{k}P(C_iC_jC_l)+\cdots+(-1)^{k-1}P(C_1C_2\cdots C_k) \tag{4}$$

其中：$P(C_i)=\prod_{i=1}^{n}P(x_i)$，$n$ 为 C_i 中包含底事件个数，则最小割集中底事件 x_i 的关键重要度[16]为

$$W_i=\frac{\partial P(T)}{\partial P(x_i)} \tag{5}$$

关键重要度反映了最小割集中底事件 x_i 发生的概率对顶事件 T 发生概率的影响程度。

1.3 因素集划分

在已建立的软件故障树基础上，求出最小割集 C_i，$i=1,2,\cdots,n$。最小割集的意义在于当软件出现某种故障，则表示与其对应的最小割集 C_i 包含的所有底事件 x_i 全部发生。

将底事件称为故障因素（简称为因素），则将因某种共同属性而组成因素的集合称为因素集，对因素集划分提出以下定义。

定义 1 设软件故障树底事件总数为 m，所有最小割集 C_i 包含底事件 x_i 的总数为 n（i 不重复），$m\geqslant n$，$m,n\in N^+$。第一类情况：若 $m>n$，则称这 n 个底事件为软件故障的主导因素，其集合称为主导因素集，其余底事件统称为次要因素，其集合称为次要因素集；第二类情况：若 $m=n$，主导因素集和次要因素集要设定关键重要度阈值 ε 进行划分，当 $W_i\geqslant\varepsilon$ 时，对应 x_i 为主导因素，否则为次要因素。

根据上述定义，将所有因素划分为主导、次要两类因素集。

2 多级模糊评价模型

2.1 模型建立

在软件可靠性评价的工程实际中，由于软件可靠性的影响因素等大多具有模糊特性，很多因子不能用确定的数值来表示，不易进行精确度量。软件可靠性还受多个因素影响，需要由多个因素进行综合评估，如软件可靠性一般可用容错性、成熟性、易恢复性、可靠依从性等因素进行描述[17]，因此可基于模糊集理论和领域专家知识建立多级模糊评价模型。

评价模型有因素集 $U=\{u_1,u_2,\cdots,u_n\}$、评价指标集 $V=\{v_1,v_2,\cdots,v_m\}$ 和评价矩阵 $\boldsymbol{R}=(r_{ij})_{n\times m}\in M_{n\times m}$（$M_{n\times m}$ 是 $n\times m$ 元综合函数）三个基本要素。U 是影响软件可靠性的所有因素集合，即软件故障树所有底事件集合的集合；V 是各因素评价等级的模糊语言变量集，如软件质量一般可分为优秀、良好、合格、不合格四个评价等级[17]；$\boldsymbol{R}$ 中元素表示对各因素对相应评价等级的隶属度，隶属度的范围是[0,1]。

在软件故障树分析基础上，根据定义 1 将所有因素分为主导、次要因素集，自上而下建立多级评价模型，从最低层开始由下而上进行递推评价，最后得到评价Ⅰ层的综合评价向量，这种方法可以统筹兼顾各因素对软件的影响，如图 2 所示。

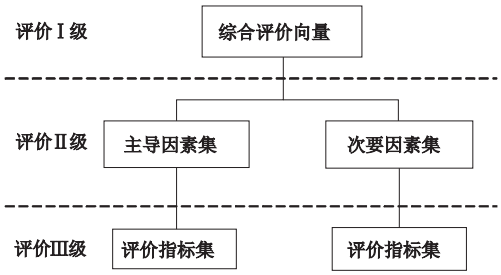

图2 软件可靠性多级评价模型

根据定义 1 中所述两种因素集划分情况，对多级评价模型中单因素和因素集的模糊权重作以下定义。

定义 2 按定义 1，若 $m>n$，则单主导因素模糊权重为

$$\lambda_i=\frac{W_j}{\sum\limits_{1\leqslant j\leqslant m}W_j} \tag{6}$$

单次要因素模糊权重为

$$\lambda_i=\frac{F_j}{\sum\limits_{1\leqslant j\leqslant m}F_j} \tag{7}$$

若 $m=n$,单主导因素和次要因素模糊权重均为

$$\lambda_i=\frac{W_j}{\sum\limits_{1\leqslant j\leqslant m}W_j} \tag{8}$$

其中:j 表示因素集中包含的底事件序数。

定义 3　按定义 1,若 $m>n$,因素集模糊权重为

$$\gamma_i=\frac{\sum\limits_{1\leqslant j\leqslant m}F_j}{\sum\limits_{i=1}^{m}F_i} \tag{9}$$

若 $m=n$,因素集模糊权重为

$$\gamma_i=\frac{\sum\limits_{1\leqslant j\leqslant m}W_j}{\sum\limits_{i=1}^{m}W_i} \tag{10}$$

定义 4　根据软件中因素不同属性,给出不同维度的评价指标集 V,称为增广评价指标集,增广评价集的特点是对主导因素评价指标较多(即评价较为细致)。

定义 5　设因素(底事件)x_i 对应评价指标集 V_i 的维度为 n_i,则 x_i 所在因素集 U_i 对应评价矩阵 $\boldsymbol{R}_i$ 的列数为 $\max\{n_i\}$。若评价指标集 V_j 的维度 $n_j<\max\{n_i\}$,则根据专家经验将不同评价集中同等级评语放于评价矩阵中同一列,矩阵空位用 0 填充,表示不进行此等级评价。由此得出的评价矩阵称为增广评价矩阵,这种方法称为增广算法。

设评价Ⅰ级的因素集为 $U=\{U_1,U_2\}$,对应评价向量为 $\boldsymbol{A}=(\gamma_1,\gamma_2)$,评价矩阵为 $\boldsymbol{R}$,综合评价向量为 $\boldsymbol{B}$;评价Ⅱ级的主导因素集为 $U_1=\{x_i\mid x_i\in C_i\}$,模糊权重为 γ_1,单因素评价向量 $\boldsymbol{A}_1=\{a_i\mid a_i=\lambda_i\}$,增广评价矩阵为 $\boldsymbol{R}_1$;评价Ⅱ级的次要因素集为 $U_2=\{x_i\mid x_i\notin C_i\}$,模糊权重为 γ_2,单因素评价向量 $\boldsymbol{A}_2=\{a_i\mid a_i=\lambda_i\}$,增广评价矩阵为 $\boldsymbol{R}_2$。其中,x_i 对应增广评价集为 v_i。

综上,评价Ⅰ级模型为

$$\begin{cases}\boldsymbol{B}=\boldsymbol{A}\circ\boldsymbol{R}\\ \boldsymbol{R}=(B'_1,B'_2)\end{cases} \tag{11}$$

评价Ⅱ级模型为

$$\begin{cases}\boldsymbol{B}_1=\boldsymbol{A}_1\circ\boldsymbol{R}_1\\ \boldsymbol{B}_2=\boldsymbol{A}_2\circ\boldsymbol{R}_2\end{cases} \tag{12}$$

其中:运算符 $\circ$ 表示模糊算子,为突出主导因素的同时兼顾次要因素对软件的影响,模糊算子选用($\cdot$,$\vee$);为使评价更符合工程实际,$\boldsymbol{R}_i$ 根据专家知识建立;B'_1、B'_2 分别由 B_1、B_2 经过归一化处理和增广运算得到。

2.2　可靠度计算

综合评价向量 $\boldsymbol{B}$ 对可靠度评定的直接效果不明显,尤其是当 $\boldsymbol{B}$ 中各元素之间差异较小时,不易于给出可靠度评定结果,针对此问题提出以下定义。

定义 6　根据定义 5 可求得增广评价矩阵 $\boldsymbol{R}$,令 $\boldsymbol{R}=\begin{bmatrix}r_{11}&\cdots&r_{1n}\\ \vdots&&\vdots\\ r_{m1}&\cdots&r_{mn}\end{bmatrix}$,则有

$$\boldsymbol{S}=(s_1\quad s_2\quad\cdots\quad s_n) \tag{13}$$

其中:$s_j=\dfrac{\sum\limits_{i=1}^{m}r_{ij}}{\sum\limits_{j=1}^{n}\sum\limits_{i=1}^{m}r_{ij}}$,称 $\boldsymbol{S}$ 为聚合向量,这种方法为聚合算法。

聚合向量 $\boldsymbol{S}$ 中元素与综合评价向量 $\boldsymbol{B}$ 中元素一一对应,表示 $\boldsymbol{B}$ 中元素对应的语言变量所占权重,用 $\boldsymbol{S}$ 与 $\boldsymbol{B}$ 相乘即是对评价语言变量隶属度加权求和,得到的是软件不可靠度,由此可定义软件可靠度。

定义 7　由式(11)~(13),软件可靠度为

$$R_s=1-\boldsymbol{B}\boldsymbol{S}^{\mathrm{T}} \tag{14}$$

2.3　可靠度评定方法

根据本文所述方法,对基于模糊故障树的软件可靠度评定方法步骤归纳如下:

a)按软件控制流程,建立软件故障树,对软件进行可靠性测试。

b)求出软件故障树最小割集,根据测试数据求出底事件发生概率及关键重要度。

c)根据定义 1 对故障树底事件进行因素集划分,按专家知识确定增广评价指标集。

d)建立软件可靠性多级模糊评价模型,确定各级评价向量。

e)根据式(11)(12)求解评价Ⅰ级综合评价向量 $\boldsymbol{B}$。

f)按定义 6 求出聚合向量 $\boldsymbol{S}$,根据式(14)计算出软件可靠度 R_s。

3　实例分析

将某航空装备软件用于对某型飞行控制系统的传感器子系统进行实时监测,以确保飞行控制系统正常工作,用本文方法对此装备软件进行可靠度评定。

a)在分析控制流程基础上,建立其软件故障树如图 3 所示,并进行软件可靠性测试。

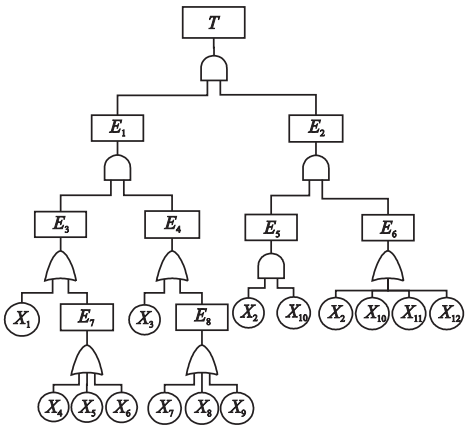

图3　某型航空装备软件故障树示意图

b)经故障树分析,其最小割集涵盖所有底事件,即定义 1 描述的第二类情况,该装备软件可靠性测试数据、底事件发生概率(失效强度)以及关键重要度计算结果如表 1 所示。

表 1　软件可靠性测试数据

最小模块单元(x_i)	代码行数	实测故障数(N)	测试时间 t/h	失效强度	关键重要度
x_1	2 378	13	266	0.045 1	0.6942×10^{-4}
x_2	5 525	43	671	0.034 3	0.1406×10^{-3}
x_3	8 462	55	597	0.026 8	0.6574×10^{-4}
x_4	1 824	23	433	0.034 6	0.9040×10^{-4}
x_5	1 156	14	212	0.047 2	0.6639×10^{-4}
x_6	1 133	17	306	0.042 5	0.7372×10^{-4}
x_7	1 838	22	386	0.023 3	0.7557×10^{-4}
x_8	1 552	13	168	0.041 7	0.4229×10^{-4}
x_9	1 286	9	133	0.067 7	0.2604×10^{-4}
x_{10}	2 329	20	397	0.042 8	0.1126×10^{-3}
x_{11}	1 089	9	189	0.063 5	0.7592×10^{-4}
x_{12}	3 005	17	406	0.051 7	0.9319×10^{-4}

c)根据领域专家知识,设阈值 $\varepsilon = 0.6942 \times 10^{-4}$,则主导因素集为 $U_1 = \{x_2, x_4, x_6, x_7, x_{10}, x_{11}, x_{12}\}$,次要因素集为 $U_2 = \{x_1, x_3, x_5, x_8, x_9\}$,$x_1 \sim x_5$ 对应的增广评价指标集为{严重,一般,不影响}、$x_6 \sim x_9$ 对应的增广评价指标集为{影响,不影响}、$x_{10} \sim x_{12}$对应的增广评价指标集为{严重,一般,影响较小,不影响}。

d)主导因素集增广矩阵为

$$A_1 = [0.2124 \quad 0.1366 \quad 0.1114 \quad 0.1142 \quad 0.1701 \quad 0.1147 \quad 0.1408],$$

$$A_2 = [0.2572 \quad 0.2436 \quad 0.2460 \quad 0.1567 \quad 0.0965]$$

次要因素集的评价向量和增广矩阵分别为

$$R_1 = \begin{bmatrix} 0.3 & 0.2 & 0 & 0.5 \\ 0.1 & 0.7 & 0 & 0.2 \\ 0.7 & 0 & 0 & 0.3 \\ 0.8 & 0 & 0 & 0.2 \\ 0.1 & 0.5 & 0.1 & 0.3 \\ 0.2 & 0.4 & 0.2 & 0.2 \\ 0.3 & 0.5 & 0.1 & 0.1 \end{bmatrix}$$

$$R_2 = \begin{bmatrix} 0.3 & 0.2 & 0.5 \\ 0.1 & 0.7 & 0.2 \\ 0.4 & 0.4 & 0.2 \\ 0.4 & 0 & 0.6 \\ 0.3 & 0 & 0.7 \end{bmatrix}$$

e)由式(11)(12)得评价向量为 $A = (0.6110, 0.3890)$,评价矩阵为 $R = \begin{pmatrix} 0.1748 & 0.1830 & 0.4390 & 0.2032 \\ 0.2711 & 0.4698 & 0 & 0.2591 \end{pmatrix}$,综合评判向量为 $B = (0.1068 \quad 0.1828 \quad 0.2682 \quad 0.1242)$。

f)由定义 6 及式(14)得聚合向量 $S = (0.2230 \quad 0.3264 \quad 0.2195 \quad 0.2311)$,则该软件可靠度 $R_s = 0.8289$。

记录软件可靠性测试过程中各时间阶段对应的累积故障数,并计算其对应的软件可靠度,数据如表 2 所示,于是可以得到测试过程中软件可靠度随累积测试时间及故障数的变化曲线,如图 4 所示。

表 2 测试过程中软件可靠度数据

累积故障数	累积测试时间/h	可靠度	累积故障数	累积测试时间/h	可靠度
21	133	0.976 1	113	494	0.864 3
46	217	0.963 4	129	567	0.861 2
59	298	0.953 8	143	590	0.856 1
71	361	0.899 1	149	620	0.839 0
86	441	0.878 2	165	671	0.828 9

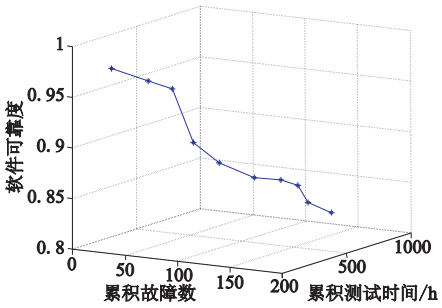

图4 测试过程中软件可靠度变化曲线

实际工程中,如果用可靠度取值区间对可靠性等级进行评定:0.9 ~ 1.0 为强、0.7 ~ 0.9 为弱、0.0 ~ 0.7 为不可靠,可知该软件可靠性弱,需要进一步修改、测试合格后才可列装使用。

4 结束语

本文从分析软件控制流程、软件模块故障逻辑分析、软件可靠性测试等三个环节建立了软件故障树定性和定量分析方法,在此基础上,用多级模糊评价模型对软件可靠性进行了定量描述,提出了基于模糊故障树的软件可靠度评定方法。作为一项软件可靠性评估技术,该方法可用于装备软件质量监督和可靠性管理,对提高软件质量、控制软件开发过程有着重要作用。

参考文献:

[1] SCHNEIDEWIND N F. Reliability modeling for safety-critical software[J]. IEEE Trans on Reliability,1997,46(1):88-98.

[2] JELINSKI Z, MORANDA P B. Software reliability research[C]// Proc of GREIBERGER. Statistical Computer Performance Evaluation. New York: Academic Press, 1972:465-484.

[3] LITTLEWOOD B, VERRALL J. A bayesian reliability growth model for computer software[J]. Applied Statistics, 1973, 22(3): 332-346.

[4] GOEL A L, OKUMOTO K. A markovian model for reliability and other performance measures of software systems[C]//Proc of National Computer Conference. 1979: 769-774.

[5] GOEL A L, KUMOTO O. A time dependent error detection rate for a large scale software system[C]//Proc of the 3 rd USA-Japan Computer Conference. 1978: 35-40.

[6] 蔡开元. 软件可靠性工程基础[M]. 北京:清华大学出版社,1995:56-57.

[7] 楼俊钢,江建慧,靳昂. 考虑软件不同失效过程偏差的软件可靠性模型[J]. 计算机学报,2010,33(7):1263-1271.

[8] 李海峰,李秋英,陆民燕. 基于 Logistic 测试覆盖率函数的软件可靠性建模研究[J]. 计算机研究与发展,2011,48(2):232-240.

[9] 姚金江,鞠瑞年. 一类软件可靠度新模型的统计推断[J]. 计算机工程与应用,2007,43(3):48-53.

[10] 张家海,胡虹,胡恒章. 组合导航软件可靠性的评估[J]. 哈尔滨工业大学学报,2003,35(3):319-323.

[11] 石柱,郑重. 软件可靠性度量实例研究[J]. 系统工程与电子技术,2011,33(1):233-236.

[12] 金星,洪延姬. 系统可靠性评定方法[M]. 北京:国防工业出版社,2005:200-216.

[13] 陆镰,陆民燕,韩峰岩. 装备软件质量和可靠性管理[M]. 北京:国防工业出版社,2006:140-142.

[14] 刘文红,王占武,吴欣. 故障树分析技术在软件测试中的应用[J]. 系统工程与电子技术,2004,26(7):985-988.

[15] 徐中伟,吴美芳. 形式化故障树分析建模和软件安全性测试[J]. 同济大学学报,2001,29(11):1299-1302.

[16] 倪绍徐,张裕芳,易宏,等. 基于故障树的智能故障诊断方法[J]. 上海交通大学学报,2008,42(8):1372-1386.

[17] 陆鑫,廖建明. 基于模糊集理论的软件质量评估研究[J]. 电子科技大学学报,2007,36(3): 652-655.