

文章编号 1004-924X(2008)11-2180-07

# 定量化故障树分析技术在空间相机软件开发中的应用

李 杨<sup>1,2</sup>, 徐抒岩<sup>1</sup>, 韩诚山<sup>1</sup>, 于 涛<sup>1</sup>, 邢忠宝<sup>1</sup>

(1. 中国科学院 长春光学精密机械与物理研究所, 吉林 长春 130033;

2. 中国科学院 研究生院, 北京 100039)

**摘要:** 为保证空间相机软件系统可靠安全地运行, 多采用定量化故障树方法对其进行可靠性分析。由于获取故障知识及确定故障树结构有一定难度, 提出了一种基于粗糙集的软件故障树建立方法。针对软件故障原因错综复杂, 无法获取大量有效数据的情况, 给出一种采用 Vague 集方法对底事件进行建模, 从而计算底事件相对重要度和顶事件发生概率的定量分析方法, 根据分析结果指导软件的可靠性设计并确定测试重点。实验结果表明: 使用提出的故障树分析方法, 有效地指导了设计人员通过容错差错和避错等设计措施提高系统的可靠性, 解决了由于软件部件可靠性指标很难精确量化, 无法进行定量分析的问题。

**关键词:** 空间相机; 软件可靠性; 故障树分析(FTA); 粗糙集; Vague 集

**中图分类号:** V475.3; TP311.1 **文献标识码:** A

## Application of quantitative fault tree analysis to software development for space camera

LI Yang<sup>1,2</sup>, XU Shu-yan<sup>1</sup>, HAN Cheng-shan<sup>1</sup>, YU Tao<sup>1</sup>, XING Zhong-bao<sup>1</sup>

(1. *Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China;*

2. *Graduate University of Chinese Academy of Sciences, Beijing 10039, China*)

**Abstract:** Quantitative Fault Tree Analysis (FTA) is very effective in analyzing the reliability and safety of space camera software system. Aiming at the difficulty to get information of system faults and construct of fault tree, a method establishing software fault tree based on Rough set is put forward. Because software failure cause is anfractuouse and available data are hard to get, a modelling method for bottom event is proposed based on Vague set to calculate the relative significance of bottom event and the occurrence probability of top event. Then, analyzed result from the model is used to guide design of software reliability. Experimental results show that the new method of software FTA can help designer improve system reliability by fault tolerance and error design. The results also show that the new method makes software quantitative analysis possible and optimizes the precision of software measurement.

**Key words:** space camera; software reliability; Fault Tree Analysis(FTA); rough set; Vague set

收稿日期: 2008-08-06; 修订日期: 2008-09-03.

基金项目: 国家自然科学基金资助项目(No. 60507003)

# 1 引 言

故障树是一种分析复杂系统可靠性、诊断系统故障的有效方法,特别适合航空、航天等高可靠性领域的可靠性评定。目前,故障树分析方法已被应用到软件开发中,有效地降低了软件测试的盲目性,提高了软件的可靠性。

空间相机是航天器的重要载荷,其软件系统功能多、接口复杂、信息处理量大、实时性要求高。由于系统内部大量的难以捉摸的相互作用,很难确定引起某故障的所有原因,也难以确定某故障带来的所有影响,因此如何有效获取故障知识和它们之间的逻辑关系,并利用已有知识成功地建立故障树是故障树分析的一个难点。

现有关于故障树的很多建树方法都没有很好地解决故障树生成过程中知识获取问题<sup>[3]</sup>。本文针对空间相机多为单点失效的特点提出了一种基于粗糙集的故障树建立方法。

故障树技术的关键是对建立的故障树进行分析,从上至下找出导致软件系统故障的所有可能因素,并尽可能地进行定量分析。但对于软件来说,其故障原因错综复杂,并且有时无法获取大量的有效数据,许多部件的可靠性指标很难精确量化,因此传统的硬件故障树定量分析方法在这里难以适用<sup>[1-2]</sup>。针对上述问题,本文提出一种用定义在 $[0,1]$ 上的三角形 Vague 集对底事件进行建模,从而计算底事件相对重要度和顶事件发生概率的方法<sup>[4]</sup>。

## 2 理论简介

### 2.1 粗糙集理论

粗糙集理论是一种知识自动发现理论,它是

刻画不完整性和不确定性的数学工具。该方法从不精确、不一致、不完备的信息系统中发现知识,生成并简化决策规则。粗糙集中的知识用一个决策表来表示,决策表是一种特殊的信息表,根据知识表达系统决策表可以定义如下:

定义 1: 设  $S = (U, R, V, f)$  是一个知识表达系统,  $U$  为样本集合,  $R = C \cup D$  为属性集合,  $C \cap D = \emptyset$ ,  $C$  称为条件属性集,  $D$  为决策属性集,  $V$  是属性值的集合,  $f: U \times R \rightarrow V$  为信息函数, 为  $U$  中每个样本的属性值, 称  $S$  为一决策表。

### 2.2 Vague 集理论

Vague 集理论是一种处理不确定信息的数学工具, 由 W. L. Gau 和 D. J. Bueher 于 1993 年提出。它本质上是 Fuzzy 集的扩展, 但与 Fuzzy 集相比, 它能更好和更准确地表达不确定信息, 在处理不确定信息方面更具有优势。

定义 2: 令  $X$  是一个对象空间, 其中的任意一个元素  $x$  中的任意一个 Vague 集  $A$  用一个真隶属函数  $t_A(x)$  和一个假隶属函数  $f_A(x)$  表示,  $t_A(x)$  是从支持  $x \in A$  的证据所导出的隶属度下界,  $f_A(x)$  是从否定  $x \in A$  的证据所导出的隶属度的下界,  $t_A(x)$  和  $f_A(x)$  将区间  $[0, 1]$  的一个实数与其中的每一个点联系起来。即:  $t_A: U \rightarrow [0, 1]$ ,  $f_A: U \rightarrow [0, 1]$ ; 其中,  $t_A(x) + f_A(x) \leq 1$ 。称  $A = \{[t_A(x), 1 - f_A(x)] | x \in X\}$  是一个 Vague 集。

对于图 1 所示的两个凸三角形 Vague 集  $A$  和  $B$ , 用元组  $A = \langle (a_1, b_1, c_1); [t_1, 1 - f_1] \rangle$ ,  $B = \langle (a_2, b_2, c_2); [t_2, 1 - f_2] \rangle$  表示, 其中  $0 \leq t_2 \leq t_1 \leq 1 - f_2 \leq 1 - f_1 \leq 1$ ,  $a_1, b_1, c_1, a_2, b_2, c_2$  分别为三角形模糊数的起点、中点和终点, 两元组分别表示模糊值大约是中点的隶属度, 用 Vague 集  $[0, 1]$  上的子区间来刻画。设  $k$  为一实数常量, 可得如下运算规则<sup>[5]</sup>:

$$kA = k \langle (a_1, b_1, c_1); [t_1, 1 - f_1] \rangle = \langle (ka_1, kb_1, kc_1); [t_1, 1 - f_1] \rangle, \quad (1)$$

$$A + B = \langle (a_1, b_1, c_1); [t_1, 1 - f_1] \rangle + \langle (a_2, b_2, c_2); [t_2, 1 - f_2] \rangle = \langle (a_1 + a_2, b_1 + b_2, c_1 + c_2); [\max(t_1, t_2), \max(1 - f_1, 1 - f_2)] \rangle, \quad (2)$$

$$A - B = \langle (a_1, b_1, c_1); [t_1, 1 - f_1] \rangle - \langle (a_2, b_2, c_2); [t_2, 1 - f_2] \rangle = \langle (a_1 - a_2, b_1 - b_2, c_1 - c_2); [\min(t_1, t_2), \min(1 - f_1, 1 - f_2)] \rangle, \quad (3)$$

$$A \times B = \langle (a_1, b_1, c_1); [t_1, 1 - f_1] \rangle \times \langle (a_2, b_2, c_2); [t_2, 1 - f_2] \rangle = \langle (a_1 \times a_2, b_1 \times b_2, c_1 \times c_2); [\min(t_1, t_2), \min(1 - f_1, 1 - f_2)] \rangle. \quad (4)$$

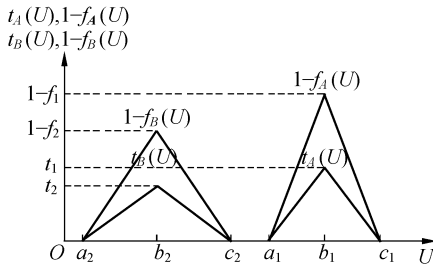


图 1 三角形 Vague 集 A 和 B

Fig. 1 Triangle Vague set A and B

### 3 基于粗糙集故障树的 Vague 分析方法

#### 3.1 基于粗糙集的故障树建立

空间相机系统的结构决定空间相机软件系统多为单点失效,因此在故障树中,其逻辑关系也多为或关系,基于以上特点本文提出一种基于粗糙集的故障树建立方法。

决策表  $S=(U, R, V, f)$ ,  $U$  为需要分析的部件集合,记为  $U=\{X_1, X_2, X_3, \dots, X_n\}$ ;  $R$  为软件系统所有可能出现的故障集合,记为  $R=\{R_1, R_2, R_3, \dots, R_m\}$ ;  $V$  为各部件发生各种故障的可能性,采用离散化的形式表示,对于某个部件  $X_i$ ,其发生故障  $R_j$  的可能性为  $V_{ij}$ ,则  $V=\{V_{11}, V_{12}, \dots, V_{1m}, V_{21}, V_{22}, \dots, V_{2m}, \dots, V_{n1}, V_{n2}, \dots, V_{nm}\}$ ;  $f: U \times R \rightarrow V$  为  $U$  中每个部件的属性值。

表 1 故障树决策表

Tab. 1 Fault tree decision table

U \ R	条件属性		...	决策属性		条件属性
	R <sub>1</sub>	R <sub>2</sub>		R <sub>j</sub>	R <sub>m</sub>	
X <sub>1</sub>	V <sub>11</sub>	V <sub>12</sub>	...	V <sub>1j</sub>	...	V <sub>1m</sub>
X <sub>2</sub>	V <sub>21</sub>	V <sub>22</sub>	...	V <sub>2j</sub>	...	V <sub>2m</sub>
...	...	...	...	...	...	...
X <sub>i</sub>	V <sub>i1</sub>	V <sub>i2</sub>	...	V <sub>ij</sub>	...	V <sub>im</sub>
...	...	...	...	...	...	...
X <sub>n</sub>	V <sub>n1</sub>	V <sub>n2</sub>	...	V <sub>nj</sub>	...	V <sub>nm</sub>

将属性集合  $R$  分为条件属性  $C$  和决策属性

$D$ , 因为空间相机软件多为单点失效,为获得部件间的联系,取故障集中的  $R_j$  为决策属性,其他属性  $R=\{R_1, R_2, \dots, R_{j-1}, R_{j+1}, \dots, R_m\}$  为条件属性。根据  $V$  取值得到  $M$  个分类的集合,据此可知部件间的关系,即某一故障出现,可能是由哪些模块引起的。这样的决策表有  $m$  个,以  $R_j$  为决策属性的决策表如表 1 所示。

上述利用决策表进行分析所建立的故障树,其顶事件仍为系统最不希望发生的故障事件,中间事件为故障事件,底事件为可能引起这些故障的部件。各底事件之间的逻辑关系为“或”关系。

#### 3.2 基于 Vague 集的故障树定量分析

故障树的定量分析,主要是计算顶事件发生的概率和底事件的相对重要度。

顶事件发生概率与底事件相对重要度都与底事件发生的概率有直接关系,但对于软件系统来说,其底事件发生的概率不像硬件数据那样容易得到,因此如何获得相关数据计算底事件发生的概率是难点。

本文先对底事件发生概率进行建模,进而利用底事件与顶事件关系来计算顶事件发生概率和底事件相对重要度。

原有用于底事件发生概率的计算公式  $P_s = 1.2 \times 10^{-3} \times e^{0.05 \times D_0} \times I_s \times W_1 \times W_2$  只考虑代码规模、软件错误密度、预防底事件所采取手段有效性加权因子、软件成熟度加权因子,根据工程经验,底事件发生的可能性还与开发本部件人员的数量、内部接口和外部接口数量有关系。对底事件发生可能性进行重新建模:

$$P_s' = 1.2 \times 10^{-3} \times 0.459 \times e^{0.05 \times D_0} \times I_s \times W_1 \times W_2 \times N \times I_1 \times I_2, \quad (5)$$

式中:

$I_s$ : 有效代码行,以千行计算;

$D_0$ : 与软件开发成熟度有关,按软件开发等级进行确定,等级越高,软件错误密度越小;

$W_1$ : 根据所采取手段的有效性进行取值,其值介于 0 与 1 之间,有效性越高,值越小;

$W_2$ : 根据软件本身的成熟度进行取值,其值介于 0 与 1 之间,软件成熟度越高值越小,一般新研软

件的成熟度为 1;

$N$ :开发本部件的人员数量,人员越多,底事件发生的可能性就越大;

$I_1$ :与本部件有关的内部接口数,接口越多,底事件发生的可能性越大;

$I_2$ :与本部件有关的外部接口数,接口越多,底事件发生的可能性越大。

由于底事件发生概率是通过建模计算得到的,而各参数取值也并非确定值,因此底事件发生的概率不应是一个精确值,应该在一定的区间范围内。为此本文提出采用定义在 $[0,1]$ 上的三角形 Vague 集来刻画底事件发生概率的方法。

具体方法如下:

(1) 根据公式(5)分别取各项参数最小值和最大值进行计算,所得结果分别记为  $a_{\min}, c_{\max}$ ;

$$(2) \text{ 计算 } b_{\text{ave}} = \frac{a_{\min} + c_{\max}}{2};$$

(3) 采用“头脑风暴”法,确定真隶属函数和假隶属函数。即由软件设计者、可靠性设计师、经验丰富者共 3 人根据历史数据和经验分别给出真隶属函数和假隶属函数。为保证所确定的值比较准确,采用加权平均的方法,即将 3 者给出的隶属度值按由大到小顺序分别乘以权 1、4 和 1 然后除以 6,所得结果作为最终的结果,因此第  $i$  个底事件的隶属函数计算公式为:

$$t_i = \frac{1 \times t_{\max} + 4 \times t_{\text{mid}} + 1 \times t_{\text{mix}}}{6}, \quad (6)$$

$$f_i = \frac{1 \times f_{\max} + 4 \times f_{\text{mid}} + 1 \times f_{\text{mix}}}{6}. \quad (7)$$

### 3.2.1 顶事件发生概率的计算

依据所建立软件故障树的与或逻辑关系,自下而上确定故障树的结构函数,并计算故障概率函数,记为

$$P(p_1, p_2, \dots, p_n) \quad (i=1, 2, \dots, n),$$

其中与/或门算子分别为:

$$P_{\text{AND}} = \prod_j^n P_j, \quad (8)$$

$$P_{\text{OR}} = 1 - \prod_j^m (1 - P_j), \quad (9)$$

根据故障概率函数和 Vague 集计算公式(1)

~(4),计算顶事件发生的概率。

### 3.2.2 底事件相对重要度的计算

由于系统构成不同,系统中各部件的重要性也并非完全相同,有些部件一有故障就会引起系统失效,有些则不然。一个部件发生故障对顶事件发生的贡献称为重要度。在进行系统设计时,可根据系统的结构、底事件的可靠性等来计算各部件的相对重要度,并把各部件相对重要度进行排队。若要提高系统的可靠性,则首先致力于提高相对重要度大的部件的可靠性。这样,有了各部件定量的相对重要度数据,就可找到系统设计的薄弱环节加以改进。第  $i$  个底事件相对重要度的计算方法为:

$$I_{(i)} = 1 - \frac{\left( \frac{P_{\text{Sc}}'(X_i)}{P_{\text{Sc}}'(\text{top})} + \frac{P_{\text{Sb}}'(X_i)}{P_{\text{Sb}}'(\text{top})} + \frac{P_{\text{Sa}}'(X_i)}{P_{\text{Sa}}'(\text{top})} \right)}{3}, \quad (10)$$

$P_{\text{Sa}}'(X_i), P_{\text{Sb}}'(X_i), P_{\text{Sc}}'(X_i)$  分别为去掉第  $i$  个底事件后的顶事件发生概率的起点、中点和终点; $P_{\text{Sa}}'(\text{top}), P_{\text{Sb}}'(\text{top}), P_{\text{Sc}}'(\text{top})$  分别为顶事件发生概率的起点、中点和终点。

将计算所得的底事件相对重要度按从大到小顺序排列。要提高系统可靠性,首先应考虑底事件相对重要度大的底事件。同时在测试时也可以根据相对重要度大小进行测试时间和重点的合理分配。

## 4 用基于粗糙集故障树的 Vague 分析法分析空间相机软件可靠性

为验证所提出的方法,本文以空间相机主控软件为例,说明如何建立故障树以及如何如何进行定量分析。空间相机主控软件有通讯、A/D 采集与处理、成像控制、调焦控制、调偏流控制和加热控制等功能。为防止故障树分析漫无边际,首先需确定分析的广度和深度。将整个空间相机主控软件系统作为一个整体元件进行处理,分析到软件部件并且不考虑硬件故障和人为因素。

### 4.1 故障树建立

决策表  $S = \{U, R, V, f\}$ ,  $U = \{X_1, X_2, X_3, X_4, X_5, X_6, X_7\}$ , 其中  $X_1$ : 1553B 通讯,  $X_2$ : 指令处理,  $X_3$ : RS-422 通讯,  $X_4$ : 调焦控制,  $X_5$ : 调偏流控制,  $X_6$ : A/D 采集与处理,  $X_7$ : 热控制;  $R = \{R_1, R_2, R_3, R_4, R_5\}$ , 其中  $R_1$ : 通讯故障,  $R_2$ : 数据获取错误,  $R_3$ : 数据处理故障,  $R_4$ : 电机控制故障,  $R_5$ : 热控制故障;  $V = \{V_1, V_2\}$ ,  $V_1$ : 可能,  $V_2$ : 不可能。以  $R_5$  为决策条件决策表见表 2。

以主控软件失效为顶事件, 经分析建立的故障树如图 2 所示。

表 2 空间相机主控软件决策表

Tab. 2 Control software of space camera decision table

部件	故障模式				
	$R_1$	$R_2$	$R_3$	$R_4$	$R_5$
$X_1$	$V_1$	$V_1$	$V_2$	$V_2$	$V_2$
$X_2$	$V_2$	$V_1$	$V_1$	$V_2$	$V_2$
$X_3$	$V_1$	$V_2$	$V_1$	$V_2$	$V_2$
$X_4$	$V_2$	$V_1$	$V_1$	$V_1$	$V_2$
$X_5$	$V_2$	$V_1$	$V_1$	$V_1$	$V_2$
$X_6$	$V_2$	$V_1$	$V_1$	$V_2$	$V_1$
$X_7$	$V_2$	$V_2$	$V_2$	$V_2$	$V_1$

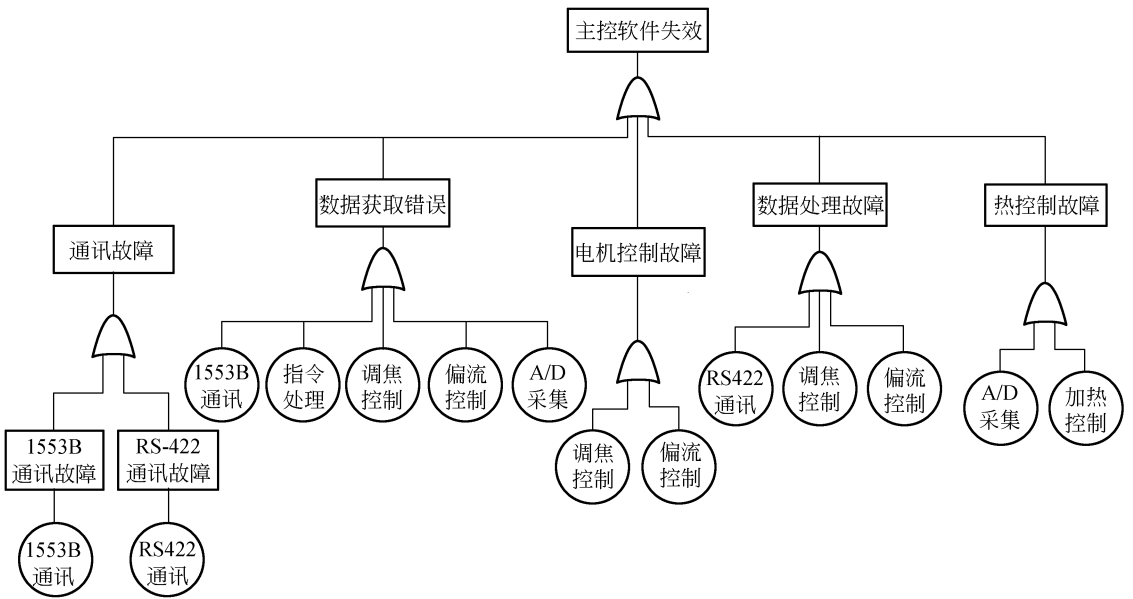


图 2 空间相机主控软件故障树

Fig. 2 Fault tree of control software for space camera

### 4.2 定量分析

根据本软件开发情况, 软件成熟度为 GJB5000 二级, 软件错误密度数  $D_0$  取 8, 关于底事件的描述和其他参数见表 3 及表 4。

应用本文提出的方法求得各底事件发生的最大可能性、最小可能性及故障概率函数分别为:

$$P_{Sa}'(X_1) = 0.000\ 986, P_{Sc}'(X_1) = 0.002\ 52,$$

$$P_{Sa}'(X_2) = 0.002\ 59, P_{Sc}'(X_2) = 0.006\ 63,$$

$$P_{Sa}'(X_3) = 0.001\ 23, P_{Sc}'(X_3) = 0.006\ 16,$$

$$P_{Sa}'(X_4) = 0.000\ 719, P_{Sc}'(X_4) = 0.001\ 53,$$

$$P_{Sa}'(X_5) = 0.000\ 986, P_{Sc}'(X_5) = 0.002\ 52,$$

$$P_{Sa}'(X_6) = 0.000\ 205, P_{Sc}'(X_6) = 0.000\ 526,$$

$$P_{Sa}'(X_7) = 0.000\ 308, P_{Sc}'(X_7) = 0.000\ 789.$$

$$P(p_{X_1}, p_{X_2}, \dots, p_{X_7}) = 1 - (1 - p_{X_1})^2 (1 - p_{X_2}) (1 - p_{X_3})^2 (1 - p_{X_4})^3 (1 - p_{X_5})^3 (1 - p_{X_6})^2 (1 - p_{X_7})$$

表 3 底事件的描述及相关数据

Tab.3 Description of bottom events and correlative data

部件标号	部件名称	$I_i$ (千行)	$N$ (个)	$I_1$ (个)	$I_2$ (个)	$W_1$ (min/max)	$W_2$ (min/max)
$X_1$	1553B 通讯	1.6	1	1	3	0.5/0.8	0.5/0.8
$X_2$	指令处理	2.1	1	1	6	0.5/0.8	0.5/0.8
$X_3$	RS-422 通讯	1.0	1	2	3	0.5/0.8	0.5/0.8
$X_4$	调焦控制	0.7	1	1	5	0.5/0.8	0.5/0.8
$X_5$	调偏流控制	0.8	1	2	3	0.5/0.8	0.5/0.8
$X_6$	A/D 采集与处理	0.5	1	1	2	0.5/0.8	0.5/0.8
$X_7$	热控制	0.5	1	1	3	0.5/0.8	0.4/0.9

表 4 底事件发生可能性及隶属函数

Tab.4 Possibility of bottom events and subjective functions

部件标号	部件名称	发生可能性(起点、中点、终点)	支持隶属函数	反对隶属函数
$X_1$	1553B 通讯	(0.000 986,0.001 75,0.002 52)	0.89	0.1
$X_2$	指令处理	(0.002 59,0.004 61,0.006 63)	0.9	0.05
$X_3$	RS-422 通讯	(0.001 23,0.003 70,0.006 16)	0.9	0.05
$X_4$	调焦控制	(0.000 719,0.001 12,0.001 53)	0.9	0.05
$X_5$	调偏流控制	(0.000 986,0.001 75,0.002 52)	0.9	0.05
$X_6$	A/D 采集与处理	(0.000 205,0.000 366,0.000 526)	0.9	0.05
$X_7$	热控制	(0.000 308,0.000 549,0.000 789)	0.8	0.1

根据上述所得数据,利用所给公式计算顶事

### 5 结 论

件发生的概率和底事件的相对重要度分别为:

$$P_{top}' \langle (0.016, 0.027, 0.039); [0.8, 0.9] \rangle;$$

$$I_{(X_1)} = 0.134; I_{(X_2)} = 0.171; I_{(X_3)} = 0.285; I_{(X_4)} = 0.171; I_{(X_5)} = 0.213; I_{(X_6)} = 0.021; I_{(X_7)} = 0.009$$

其中  $P_{top}'$  为顶事件发生概率的三角形 Vague 集表示,  $I_{(X_1)}, I_{(X_2)} \dots I_{(X_7)}$  分别为底事件  $X_1, X_2 \dots X_7$  的相对重要度。

$$I_{(X_3)} > I_{(X_5)} > I_{(X_2)} \geq I_{(X_4)} > I_{(X_1)} > I_{(X_6)} > I_{(X_7)}$$

由上述计算可知,底事件  $X_3$  对顶事件的影响最大,底事件  $X_7$  对顶事件的影响最小。

本文详细讨论了如何利用粗糙集知识进行故障树的建立以及怎样采用 Vague 集方法计算底事件相对重要度和顶事件发生概率。结果表明:利用粗糙集理论可以进行知识获取,可以快速有效地建立故障树。同时采用本文所提出的故障树分析方法使得软件故障树不仅可以进行定性分析,还可以进行定量分析。而通过对故障树顶事件发生概率和底事件相对重要度的定量分析,在软件测试时可以对测试时间和测试重点进行合理分配。

### 参考文献:

[1] 王永传, 郝文贤, 庄钊文. 一种故障树模糊重要度分析的新方法[J]. 国防科技大学学报, 1999, 21(3): 63-66.  
WANG Y CH, YU W X, ZHUANG ZH W. A new method of the importance analysis of FTA [J]. *Journal of National University of Defense Technology*, 1999, 21(3):63-66. (in Chinese)

[2] CHENG C H, MON D L. Fuzzy system reliability analysis by confidence interval [J]. *Fuzzy Sets and Systems*,

1993, 56(5): 29-35.

- [3] 谭天乐,李平,宋执环. 基于粗糙集的逻辑故障树方法及其应用[J]. 仪器仪表学报,2004, 25(1): 18-22.  
TAN T L, LI P, SONG ZH H. Rough set-based logical fault tree and its application[J]. *Chinese Journal of Scientific Instrument*, 2004, 25(1):18-22. (in Chinese)
- [4] 周海京,遇今. 故障模式、影响及危害性分析与故障树分析[M]. 北京:航空工业出版社,2003.  
ZHOU H J, YU J. *Failure Effect, Effect and Criticality Analysis* [M]. Beijing: Aviation Industry Press, 2003. (in Chinese)
- [5] 石柱,何新贵,武庄. 软件可靠性及其评估[J]. 计算机应用,2000,20(11):1-5.  
SHI ZH, HE X G, WU ZH. Software reliability and its estimation [J]. *Journal of Computer Applications*, 2000, 20(11): 1-5. (in Chinese)
- [6] 陈伟,钟建. 基于精度成本控制的精密传输机器人系统可靠度优化配置方法[J]. 光学精密工程,2007,15(8): 1253-1256.  
CHEN W, ZHONG J. Optimum reliability distribution of precise conveying robot system based on precision and cost controls[J]. *Opt. Precision Eng.*, 2007,15(8):1253-1256. (in Chinese)
- [7] 陈伟,钟建. 面向工业机器人系统的三种可靠度配置策略的研究[J]. 光学精密工程,2002,10(2): 165-170.  
CHEN W, ZHONG J. Research on three reliability distribution methods of an industrial robot system [J]. *Opt. Precision Eng.*, 2002,10(2):165-170. (in Chinese)

作者简介:李 杨(1981—),女,黑龙江安达人,在读博士研究生,主要研究方向为软件工程、可靠性工程。E-mail: victoria\_liy@163.com。

邢忠宝(1943—),男,研究员,副总工程师,博士生导师,主要从事软件工程、计算机控制系统、专用控制系统等方面的研究工作。

## ● 下期预告

# 新型惯性式直线超声电机的运动机理及实验

时运来,李玉宝,赵淳生

(南京航空航天大学 精密驱动研究所,江苏 南京 210016)

提出了一种新型惯性式直线型压电电机。它利用压电晶体的逆压电效应和惯性位移原理,由定子的轴向变形,通过摩擦力的作用,以惯性位移的形式把运动传递出去。该电机由两部分组成,定子为具有位移放大功能的压电复合换能器和轴组成,动子为一带有缺口和环形凹槽的环。该文详细分析了该电机的工作原理,并通过仿真和实验相结合的方法分析了两种激励方式对电机性能的影响。实验结果表明:样机采用方波驱动比采用锯齿波驱动的效果好;在采用方波驱动信号驱动时,样机的最大空载速度可达到 11 mm/s,最大输出力达到 0.5 N。