

# 基于顺序逻辑的状态事件故障树定性分析模型

范亚琼,陈海燕

(南京航空航天大学 计算机科学与技术学院,江苏 南京 210016)

**摘要:**针对现有的状态事件故障树(SEFT)定性分析方法在反映失效系统中构件状态与事件逻辑顺序关系方面的不足,提出了基于顺序逻辑的状态事件故障树定性分析模型。该模型通过建立构件与逻辑门的端口映射表,定义逻辑门到布尔逻辑的转换规则限定状态和事件的次序关系,根据顺序逻辑转换规则获得导致系统失效的状态事件序列(最小割序集),以解决系统失效应满足的状态与事件的逻辑顺序关系问题。为验证所提出模型的有效性和可行性,以火灾防护系统为研究对象进行了实验验证实验。实验结果表明,所提出的模型有效可行,所获得的最小割序集能够反映各失效事件和状态间的顺序逻辑关系,分析结果符合客观实际,为 SEFT 的定性分析提供了一种新的技术途径和方法借鉴。

**关键词:**顺序逻辑;端口映射表;最小割序集;转换规则;定性分析

中图分类号:TP311

文献标识码:A

文章编号:1673-629X(2017)08-0012-04

doi:10.3969/j.issn.1673-629X.2017.08.003

## Qualitative Analysis Model of State/Event Fault Tree with Sequential Logic

FAN Ya-qiong, CHEN Hai-yan

(Department of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

**Abstract:** In order to overcome the shortcomings of reflecting the relationship between component state and sequential logic in failure system for State/Event Fault Tree (SEFT) qualitative analysis method, a qualitative analysis model of state event fault tree based on sequential logic is proposed. It defines transition rule of logic gate to Boolean logic to define the order of state and event by establishment of the port mapping table of components and logic gates, and gets the sequence of state events (minimum cut sequence set) that leads to system failure according to sequential logic transformation rules to solve the problem of logic order of state and event for system failure. In order to verify the validity and feasibility of the proposed model, a fire protection system has been established as an example of object for verification experiment. It is indicated that the cut order set can reflect the sequential logic relation between the failure event and the state. The feasibility of this model has been verified, which is consistent with the objective reality. Therefore, a new effective approach is provided for the qualitative analysis of SEFT.

**Key words:** sequential logic; port mapping table; minimum cut sequence set; conversion rules; qualitative analysis

## 0 引言

状态事件故障树<sup>[1]</sup>(State/Event Fault Tree, SEFT)是基于软件的构件设计模型建立的一种表达系统失效行为的安全性分析模型。与传统的故障树<sup>[2-4]</sup>不同的是,状态事件故障树具有构件化和基于状态的特性,且严格区分了状态、事件语义,增加了状态依赖的表达。因此,对状态事件故障树进行最小割序集分析不仅要关注构件失效的集合,也要关注构件失效事件和状态

间的顺序关系,即分析出最小割序集。目前国内外安全分析领域的研究人员对 SEFT 的定性分析进行了相关研究。Michael Roth<sup>[5-6]</sup>提出将 SEFT 转换到确定随机 Petri 网(Deterministic and Stochastic Petri Nets, DSPN),在 DSPN 的基础上计算其顺序逻辑并分析导致系统失效的最小割序集,但是该最小割序集只关注失效事件,未涉及失效事件发生时相关构件应满足的状态;徐博士<sup>[7]</sup>提出基于接口自动机的软件失效行为

收稿日期:2016-08-13

修回日期:2016-11-23

网络出版时间:2017-07-05

基金项目:国家“十三五”重点基础科研项目(JCKY2016206B001);江苏省六大人才高峰项目(XXRJ-004);软件新技术与产业化协同创新中心资助项目

作者简介:范亚琼(1990-),女,硕士研究生,研究方向为软件工程、系统建模与仿真;陈海燕,讲师,研究方向为数据挖掘、民航信息化等。

网络出版地址: <http://cnki.net/kcms/detail/61.1450.TP.20170705.1650.032.html>

安全性分析模型。该模型考虑构件状态作为卫式条件对逻辑门输出产生的影响,计算过程繁琐,且最小割序集中未能反映状态和事件间的次序关系。

为此,提出了基于顺序逻辑的状态事件故障树定性分析模型。该模型通过分析构件与外界环境交互的状态或事件,建立逻辑门与构件端口相关联的端口映射表,定义逻辑门的转换规则,通过扩展布尔逻辑提出顺序逻辑转换规则,自顶向下计算分析整体系统的失效序列,通过化简操作获得系统的最小割序集。

1 问题描述

对软件失效行为进行安全性分析的目的是寻找导致软件系统失效的关键事件集合,即最小割序集<sup>[7]</sup>。作为反映构件失效逻辑层次关系的逻辑门,是定性分析的关键所在。

在 SEFT 中,构件的状态虽不能和事件一样触发逻辑门输出的发生,但是它可以允许或禁止逻辑门输出的发生。在进行 SEFT 定性分析的过程中,需要同时考虑构件的基本事件以及部分可作为卫式的构件状态。

对于状态事件混合逻辑门,其状态与事件存在顺序关系,当卫式条件成立时,失效事件的发生导致逻辑门产生输出,当卫士条件不成立时,失效事件的发生不会导致逻辑门产生输出。

2 基于顺序逻辑的 SEFT 定性分析模型

在状态事件故障树中,顶层事件的发生依赖于基本事件的发生顺序,而逻辑门中的卫式条件决定了发生的基本事件是否能对全局系统失效产生影响。提出了基于顺序逻辑<sup>[8]</sup>的 SEFT 定性分析模型,通过记录构件与逻辑门的交互端口,将逻辑门转换成布尔逻辑表达式,根据顺序逻辑转换规则,分析引起系统失效的最小关键状态事件序列。

2.1 端口映射表

构件与外界环境的交互分为三种:构件与构件之间的触发交互、构件与逻辑门的交互、逻辑门与逻辑门的交互。对 SEFT 进行定性分析,只需关注引起系统失效的关键事件或状态,对于构件内部的行为活动,若不产生外部输出,则不影响定性分析的结果。构件与外界环境的交互通过输入输出端口实现,因此,通过端口映射表建立构件和逻辑门与外界交互的关系。端口映射表的建立分三步完成:

(1)当系统规模较大时,为了方便引用,对构件和逻辑门进行编号。

(2)对于每个构件,记录与外界环境交互的关键事件或状态,若每个构件的交互端口较少,只需建立一

个表存放所有的构件交互行为,若构件的交互端口较多,可为每一个构件建立一个组件端口映射表,表中说明构件编号、源端口、目的端口、状态事件类型、输入输出方向、具体行为。

(3)对每个逻辑门,记录与外界交互的端口,若每个逻辑门的交互端口较少,只需建立一张表存放所有的逻辑门交互行为,若逻辑门的交互端口较多,可为每一个逻辑门建立一张逻辑门端口映射表,表中说明逻辑门编号、源端口、目的端口、状态事件类型、数据流向。

2.2 逻辑门转换规则

在 SEFT 的逻辑门<sup>[9]</sup>中,状态不能和事件一样触发逻辑门输出的发生,但是状态可以允许或者禁止逻辑门输出的发生。当所有的卫式条件都为真时,逻辑门才能触发。为了反映状态和事件之间的区别,所有逻辑门的输入和输出都标识了状态和事件端口类型。引入符号<表示先后次序关系,事件类型用  $E$  表示,状态类型用  $S$  表示。对 SEFT 的逻辑门类型及其转换规则描述如下:

(1)状态与门(State-AND Gate):具有一个状态类型的输出和一个或者多个状态类型的输入。它所表示的语义是当所有的输入表达式都满足时,输出状态才会为真。其中输出状态  $S$  唯一,输入状态的个数  $n$  是任意的,且所有输入的顺序是可交互换的,其转换规则为  $S = [S_1 \wedge S_2 \wedge \cdots \wedge S_n]$ 。

(2)事件状态混合与门(Event/state-AND Gate):具有一个事件  $E_1$  输入和  $n$  个状态  $(S_1 \cdots S_n)$  输入,输出为事件类型,通常第一个输入是事件类型。该逻辑门的语义为当输入事件触发,并且输入的状态表达式都满足时,输出事件被触发。其转换规则为  $E = [S_1 \wedge S_2 \wedge \cdots \wedge S_n] < E_1$ 。

(3)状态或门(State-OR Gate):具有一个状态类型的输出和一个或者多个状态类型的输入。它的语义为当一个或者多个输入表达式得到满足时,输出状态为真。其中输入的状态个数  $n$  是任意的,且所有输入状态的顺序是可交换的。其转换规则为  $S = [S_1 \vee S_2 \vee \cdots \vee S_n]$ 。

(4)优先与门(Priority-AND Gate):具有事件类型的输出和一个或者多个事件类型的输入。它的语义为当输入事件以从左到右的顺序依次发生时,输出事件被触发。其转换规则为  $E = E_1 < E_2 < \cdots < E_n$ 。

2.3 顺序逻辑转换规则

通过端口映射表和逻辑门转换规则可以获得由构件状态和事件组成的全局逻辑表达式。对布尔逻辑规则进行扩展,提出顺序逻辑转换规则<sup>[10-13]</sup>,对全局逻辑表达式进行化简,获得引起系统失效的最小割序集。

分配规则：  
 $(S_1 \vee S_2) < E_1 \Leftrightarrow S_1 < E_1 \vee S_2 < E_1$   
 $E_1 < (S_1 \vee S_2) \Leftrightarrow E_1 < S_1 \vee E_1 < S_2$   
交换规则：  
 $E_1 \vee E_2 \Leftrightarrow E_2 \vee E_1$   
最小化规则：  
 $S_1 \vee S_2 \text{ if } S_1 \subseteq S_2 \Leftrightarrow S_1$   
接下来对分配规则  $(S_1 \vee S_2) < E_1 \Leftrightarrow S_1 < E_1 \vee S_2 < E_1$  进行一般性证明：

用  $\alpha, \beta, \gamma$  表示失效状态或事件,所有事件和状态构成的非空集合为  $\Omega$ ,  $\alpha, \beta, \gamma \in \Omega$ ,  $t(\alpha)$  表示事件或状态的发生时间,若  $t(\alpha) < t(\beta)$ ,  $t(\beta) < t(\gamma)$ , 易得  $t(\alpha) < t(\gamma)$ 。考虑在规定的的时间区间  $[0, T]$  内,定义  $\Omega$  的真值指派,用  $\sigma$  表示：

$$\sigma(A) = \begin{cases} 1, & t(A) \in [0, T] \\ 0, & t(A) \in (0, +\infty] \end{cases}$$
  
 $\sigma$  定义了一个从  $\Omega$  到  $\{0, 1\}$  的映射,其含义是:如果事件  $A$  未在规定的的时间区间  $[0, T]$  内失效,不管以后其失效与否,都认为其真值为 0;反之为 1。

$$\sigma(\alpha < \beta) = \begin{cases} 1, & 0 \leq t(\alpha) \leq t(\beta) \leq T \\ 0, & \text{else} \end{cases}$$
  
(1) 当  $t(\alpha) < t(\beta)$ ,  $t(\beta) < t(\gamma)$  且均大于  $T$ , 等号左右两边赋值显然均为 0;  
(2) 任取 2 个大于  $T$ , 也容易得到等号两边赋值均为 0;

(3) 任取一个大于  $T$ , 考虑 6 种情形:  
①  $0 \leq t(\gamma) \leq t(\beta) \leq T < t(\alpha)$   
②  $0 \leq t(\beta) \leq t(\gamma) \leq T < t(\alpha)$   
③  $0 \leq t(\alpha) \leq t(\gamma) \leq T < t(\beta)$   
④  $0 \leq t(\gamma) \leq t(\alpha) \leq T < t(\beta)$   
⑤  $0 \leq t(\alpha) \leq t(\beta) \leq T < t(\gamma)$   
⑥  $0 \leq t(\beta) \leq t(\alpha) \leq T < t(\gamma)$

通过证明可得,左右两边赋值均相同,且无永真式或永假式。满足等价性定理。  
其余顺序逻辑转换规则,同理可证。

3 实验验证

图 1 给出一个火灾防护系统的 SEFT 定性分析实例。顶层事件表示火灾防护系统的失效,PAND 门的输出表示检测到火灾,OR<sub>2</sub> 的输出表示喷淋系统失效。该 SEFT 所描述的危害为:当感烟传感器和感温传感器都相继检测到火灾,但是喷淋系统失效,最终导致火灾发生。

该系统含有 7 个构件,分别是感烟传感器 SD<sub>1</sub> 和 SD<sub>2</sub>,感温传感器 HD,温度传感器 TS,喷嘴 N<sub>1</sub> 和 N<sub>2</sub>,水

泵 P。另外,含有 6 个逻辑门,分别为 AND<sub>3</sub>, PAND, OR<sub>2</sub>, OR<sub>1</sub>, AND<sub>1</sub>, AND<sub>2</sub>。下面将逐步介绍最小割序集的生成过程。

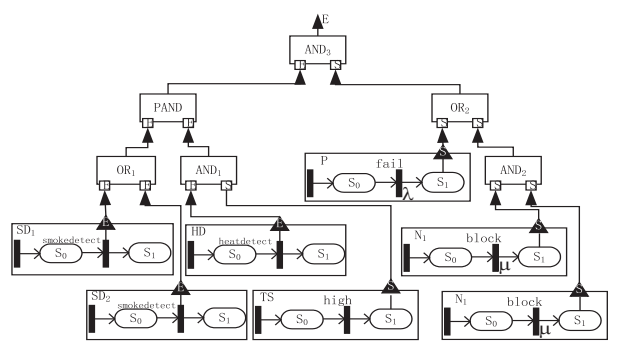


图 1 火灾防护系统

第一步:对 SEFT 的所有逻辑门和组件进行编号,见表 1 和表 2。

表 1 组件编号表

组件名称	编号
P	C <sub>1</sub>
SD <sub>1</sub>	C <sub>2</sub>
HD	C <sub>3</sub>
N <sub>1</sub>	C <sub>4</sub>
SD <sub>2</sub>	C <sub>5</sub>
TS	C <sub>6</sub>
N <sub>2</sub>	C <sub>7</sub>

表 2 逻辑门编号表

逻辑门名称	编号
AND <sub>3</sub>	G <sub>1</sub>
PAND	G <sub>2</sub>
OR <sub>2</sub>	G <sub>3</sub>
OR <sub>1</sub>	G <sub>4</sub>
AND <sub>1</sub>	G <sub>5</sub>
AND <sub>2</sub>	G <sub>6</sub>

第二步:为每个逻辑门和组件建立端口映射表,见表 3 和表 4。

表 3 组件端口映射表

编号	源端口	目标端口	类型	方向	行为
C <sub>1</sub>	C <sub>1out1</sub>	G <sub>3in1</sub>	状态	输出	S
C <sub>2</sub>	C <sub>2out1</sub>	G <sub>4in1</sub>	事件	输出	smokedetect
C <sub>3</sub>	C <sub>3out1</sub>	G <sub>5in1</sub>	事件	输出	heatdetect
C <sub>4</sub>	C <sub>4out1</sub>	G <sub>6in1</sub>	状态	输出	S
C <sub>5</sub>	C <sub>5out1</sub>	G <sub>4in2</sub>	事件	输出	smokedetect
C <sub>6</sub>	C <sub>6out1</sub>	G <sub>5in2</sub>	状态	输出	S
C <sub>7</sub>	C <sub>7out1</sub>	G <sub>6in2</sub>	状态	输出	S

表 4 逻辑门端口映射表

编号	源端口	目标端口	类型	方向
$G_1$	$G_{1out1}$		事件	输出
	$G_{2out1}$	$G_{1in1}$	事件	输入
	$G_{3out1}$	$G_{1in2}$	态	输入
$G_2$	$G_{2out1}$	$G_{1in1}$	事件	输出
	$G_{4out1}$	$G_{2in1}$	事件	输入
	$G_{5out1}$	$G_{2in2}$	事件	输入
$G_3$	$G_{3out1}$	$G_{1in2}$	状态	输出
	$C_{1out1}$	$G_{3in1}$	状态	输入
	$G_{6out1}$	$G_{3in2}$	状态	输入
$G_4$	$G_{4out1}$	$G_{2in1}$	事件	输出
	$C_{2out1}$	$G_{4in1}$	事件	输入
	$C_{5out1}$	$G_{4in2}$	事件	输入
$G_5$	$C_{5out1}$	$G_{2in2}$	事件	输出
	$C_{3out1}$	$C_{5in1}$	事件	输入
	$C_{6out1}$	$C_{5in2}$	状态	输入
$G_6$	$G_{6out1}$	$G_{3in2}$	状态	输出
	$C_{4out1}$	$G_{6in1}$	状态	输入
	$C_{7out1}$	$G_{6in2}$	状态	输入

第三步:将逻辑门转换成逻辑表达式。

$G_1:E=(S_2<E_1)$      $G_2:E=(E_1<E_2)$

$G_3:S=(S_1\vee S_2)$      $G_4:E=(E_1<E_2)$

$G_5:E=(S_2<E_1)$      $G_6:E=(S_1\wedge S_2)$

第四步:结合端口映射表,自上到下转换成系统失效的顺序逻辑表达式,并根据顺序逻辑转换规则进行化简。

$G_{1out1}=(G_{1in2}<G_{1in1})=((C_{1out1}\vee G_{6out1})<(G_{4out1}<G_{5out1}))=((P.S<(SD_1.smokedetect<(TS.S<HD.heatdetect))))\vee(P.S<(SD_2.smokedetect<(TS.S<HD.heatdetect))))\vee((N_1.S\wedge N_2.S)<(SD_1.smokedetect<(TS.S<HD.heatdetect))))\vee(((N_1.S\wedge N_2.S)<(SD_2.smokedetect<(TS.S<HD.heatdetect))))$

第五步:导致系统失效的最小割序集为:

$P.S<(SD_1.smokedetect<(TS.S<HD.heatdetect))$

$P.S<(SD_2.smokedetect<(TS.S<HD.heatdetect))$

$(N_1.S\wedge N_2.S)<(SD_1.smokedetect<(TS.S<HD.heatdetect))$

$(N_1.S\wedge N_2.S)<(SD_2.smokedetect<(TS.S<HD.heatdetect))$

4 评价结果及分析

为了分析基于顺序逻辑 SEFT 定性分析方法的可行性、可靠性及优缺点,采用基于接口自动机的软件失效行为安全性分析方法<sup>[13-14]</sup>对该火灾防护系统进行

定性分析,评价结果如下所示:

$(SD_1.smokedetect<HD.heatdetect)\wedge TS.S\wedge P.S$

$(SD_1.smokedetect<HD.heatdetect)\wedge TS.S\wedge N_1.S\wedge N_2.S$

$(SD_2.smokedetect<HD.heatdetect)\wedge TS.S\wedge N_1.S\wedge N_2.S$

$(SD_2.smokedetect<HD.heatdetect)\wedge TS.S\wedge P.S$

基于接口自动机的定性分析方法仅能获得基本事件的失效序列,如上述割序集  $(SD.smokedetect<HD.heatdetect)\wedge TS.S\wedge P.S$ ,无法体现失效事件 HD.heatdetect 与构件状态 TS.S 的次序关系。若 HD.heatdetect 事件仅发生一次且发生时间先于构件状态 TS.S 成立时间,则不会导致系统失效。应用基于顺序逻辑的定性分析方法获得相应最小割序集  $P.S<(SD.smokedetect<(TS.S<HD.heatdetect))$ ,可以看出 TS.S 优于 HD.heatdetect 发生,且 P.S 应先于 SD.smokedetect<(TS.S<HD.heatdetect) 状态事件序列所产生的输出事件,SD.smokedetect 发生时间先于 TS.S<HD.heatdetect,而 SD.smokedetect 与 TS.S 之间并无逻辑顺序关系,符合客观实际。实验表明,基于顺序逻辑的 SEFT 定性分析模型计算引起系统失效的最小割序集,不仅可以体现基本事件发生的次序关系,而且可以体现状态与相关事件的优先关系,对导致系统失效的割序集序列描述更加准确。

5 结束语

针对现有的 SEFT 定性分析方法中未能体现状态与事件的顺序逻辑关系问题,提出了基于顺序逻辑的状态事件故障树定性分析模型。通过分析构件内部时序活动对系统失效的影响,将构件与外界交互的状态或事件记为可能对系统失效产生影响的候选事件或状态,符合客观实际。同时,定义逻辑门的转换规则,获得由构件状态和事件组成的全局逻辑表达式,根据顺序逻辑规则化简得到最小割序集。

实验结果表明,该方法不仅可以获得导致系统失效的最小割序集,而且可以反映事件和状态之间的优先关系,验证了方法的可行性、可靠性,为 SEFT 的定性分析方法提供了新思路。

参考文献:

[1] Kaiser B. State event trees: a safety and reliability analysis technique for software controlled systems [D]. Kaiserslautern:University of Kaiserslautern,2007.

(下转第 19 页)



形变和透视形变是比较困难的。大多数非刚体三维重构算法通常进行了简化,只考虑正交投影下的非刚体重构。为了解决透视投影下的非刚体重构问题,在基于形状基的非刚体三维运动重构研究以及从动态图像序列中恢复非刚体三维结构和运动信息的基础上,将基于形状基的 EM 算法和仿射迭代方法有机结合,并假设非刚体的三维形状是刚性形状基的加权线性组合,在正交投影下运用 EM 算法,解决追踪点丢失的问题,提高算法的鲁棒性。同时使用仿射迭代的方法,逐步修正投影点到透视投影模型下,逼近全透视投影摄像机模型下的结果,较好地实现了透视投影模型下较好的鲁棒性以及重构效果。

参考文献:

[1] Bregler C, Hertzmann A, Biermann H. Recovering non-rigid 3D shape from image streams[C]//IEEE conference on computer vision and pattern recognition. [s. l. ]:IEEE,2000:690-696.

[2] Torresani L, Hertzmann A, Bregler C. Learning non-rigid 3D shape from 2D motion[ C]//NIPS. [s. l. ]:[s. n. ],2004:1555-1562.

[3] Gonzalez-Mora J,Torre F D L,Guil N,et al. Learning a generic 3D face model from 2D image databases using incremental structure-from-motion[J]. Image and Vision Computing, 2010,28(7):1117-1129.

[4] Lladó X,Bue A D,Oliver A,et al. Reconstruction of non-rigid

3D shapes from stereo-motion[J]. Pattern Recognition Letters,2011,32(7):1020-1028.

[5] Khan I. Non-rigid structure-from-motion with uniqueness constraint and low rank matrix fitting factorization[J]. IEEE Transactions on Multimedia,2014,16(5):1350-1357.

[6] 林义闽,吕乃光,娄小平,等. 用于弱纹理场景三维重建的机器人视觉系统[J]. 光学精密工程,2015,23(2):540-549.

[7] 彭亚丽,刘侍刚,裴国永. 一种线性迭代非刚体射影重建方法[J]. 西安交通大学学报,2015,49(1):102-106.

[8] 吴悦,翁小兰. 非刚体三维运动图像重建优化模型仿真[J]. 微电子学与计算机,2015,32(5):157-160.

[9] 李水平,吴雨. 一种融合纹理的三维图像重建快速实现方法[J]. 计算机技术与发展,2014,24(5):138-141.

[10] 闫晓萌. 基于稀疏逼近的非刚体三维运动重建研究[D]. 杭州:浙江理工大学,2016.

[11] 张珊珊,吕东辉,孙九爱. 近光源光度立体三维重建误差分析[J]. 计算机技术与发展,2015,25(1):168-172.

[12] 彭亚丽,刘侍刚,贲晔烨,等. 基于非刚体的线性迭代相机自标定方法[J]. 电子学报,2016,44(5):1051-1054.

[13] 刘彦宏,王洪斌,杜威,等. 基于图像的树类物体的三维重建[J]. 计算机学报,2002,25(9):930-935.

[14] Christy S,Horaud R. Euclidean shape and motion from multiple perspective views by affine iterations[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,1996,18(11):1098-1104.

(上接第 15 页)

[2] 刘文彬. 基于模块化思想的动态故障树分析方法研究[D]. 南京:南京理工大学,2009.

[3] 郭勇. 基于构件的软件系统的可靠性评估方法研究[D]. 哈尔滨:哈尔滨工业大学,2013.

[4] 刘东. 空间信息处理系统可靠性设计与分析关键技术研究[D]. 长沙:国防科学技术大学,2008.

[5] Roth M R,Liggesmeyer P. Qualitative analysis of state/event fault trees for supporting the certification process of software-intensive systems[C]//IEEE international symposium on software reliability engineering workshops. [s. l. ]:IEEE,2013:353-358.

[6] Roth M,Hartoyo A,Liggesmeyer P. Efficient reachability graph development for qualitative analysis of state/event fault trees[C]//IEEE international symposium on software reliability engineering workshops. [s. l. ]:IEEE,2015:144-151.

[7] 徐丙凤. 构件化嵌入式软件安全性分析方法研究[D]. 南京:南京航空航天大学,2014.

[8] Roth M,Liggesmeyer P. Sequential logic for state/event fault trees:a methodology to support the failure modeling of cyber

physical systems[C]//International conference on computer safety,reliability,and security. [s. l. ]:Springer International Publishing,2015:121-132.

[9] 李彦锋. 复杂系统动态故障树分析的新方法及其应用研究[D]. 成都:电子科技大学,2013.

[10] Guck D,Han T,Katoen J P,et al. Quantitative timed analysis of interactive Markov chains[C]//NASA formal methods symposium. [s. l. ]:[s. n. ],2012:8-23.

[11] 徐丙凤,黄志球,胡军,等. 一种状态事件故障树的定量分析方法[J]. 电子学报,2013,41(8):1480-1486.

[12] Tang Z,Dugan J B. Minimal cut set/sequence generation for dynamic fault trees[C]//Annual symposium on reliability and maintainability. Charlottesville,USA:IEEE,2004:207-213.

[13] 覃庆努. 复杂系统可靠性建模、分析和综合评价方法研究[D]. 北京:北京交通大学,2012.

[14] Boudali H,Crouzen P M. A rigorous,compositional,and extensible framework for dynamic fault tree analysis[J]. IEEE Transactions on Dependable and Secure Computing,2010,7(2):128-143.