2017年第3期 总126期
ISSN 1674-9804 CN 31-1614/V

DOI: 10.19416/j.cnki.1674-9804.2017.03.002

航电系统安全性分析工具设计与研究

Design and Research on Safety Analysis Tool for Avionics System

刘 宇 刘永超 / LIU Yu LIU Yongchao
(上海飞机设计研究院,上海 201210)
(Shanghai Aircraft Design and Research Institute, Shanghai 201210, China)

摘 要:
随着航电系统综合化程度的不断提高,传统的安全性分析方法过于依赖工程经验,难以保证失效模式的完备性。同时在系统迭代设计的过程中,由于系统的复杂性,会导致安全性分析工作量过大,增加了时间及经济成本。针对上述问题,设计了一种自动化安全性分析工具,基于 SysML 描述语言建立安全性数据模型,采用路径追溯的方法完成故障树自动建模,并对生成的故障树进行共模分析和区域安全性分析。以某系统为例的实验结果表明,该工具能够实现故障树自动建模与分析,提高了安全性分析的效率和完备性。
关键词:航空电子系统;故障树;系统建模语言;安全性分析
中图分类号:V243 **文献标识码:**A
[**Abstract**] With the continuous improvement of avionics system, the traditional method of safety analysis is difficult to guarantee the completeness of failure mode as it is too dependent on engineering experience. And in the process of system iterative design, due to the complexity of the system, the workload of safety analysis is too large, which increases the time and cost. Aiming at the above problems, an automated safety analysis tool was designed, the safety data model was established based on SysML, automatic fault tree was built by using route tracing method, and the common mode analysis and zone safety analysis were carried out on the generated fault tree. The experimental results of a system show that the tool can realize the automatic modeling and analysis of the fault tree and improve the efficiency and competeness of the safety analysis.
[**Keywords**] avionics system; fault tree; systems modeling language; safety analysis

0 引言

航空电子系统结构经历了从最初的分立式航电系统、联合式航电系统到综合式航电系统的发展历程[1]。目前航空电子的发展以第三代为主,采用更少的、更加集中的处理单元取代了数量众多的独立处理器和外场可更换单元(LRU),在新一代民机上具有减少重量、节约维修费用、减少资源配置、提高资源效能、降低机组人员工作负载等重要的作用[2]。综合化在带来上述收益的同时,也增加了系统复杂性,如功能融合、硬件逐渐由软件替代、系统与飞行员的交互增加,使得系统故障在综合过程中的传播和不确定性对系统安全性产生很大的影响。传统的安全性分析方法(如 FTA,FMEA)[3] 主要依赖于工程经验,并且与系统设计不是同步进行的,随着系统复杂程度的提高,很难列举出系统所有的失效模式和影响,同时由于系统设计的迭代,很难保证安全性需求能及时地反馈到系统设计中。上述问题的根本原因在于系统设计与安全性分析数据不能统一表示,使得系统设计不能展示其安全性属性,安全性分析结果不能直接反馈到系统设计模型中[4]。

针对上述问题,需要对系统设计和安全性分析采用一致的形式化模型,并基于这个模型实现自动化安全性分析。本文基于 SysML 描述语言[5] 的系统架构数据模型 ,研究依附于系统架构数据模型的

安全性数据模型描述方法；接着，利用模型驱动安全性分析的思想，研究自动化故障树分析方法和自动化安全隔离要求检查方法；最后基于自动化分析方法，设计了快速安全性分析工具，并通过实验验证了该工具的有效性。

1 基于 SysML 的安全性数据模型描述

SysML 是 UML 在系统工程应用领域的扩展，能对系统工程的各种问题建模，并有效支持需求说明、系统结构设计、功能行为和分配。

本文通过对基于 SysML 描述语言的系统架构数据模型添加安全性属性来建立安全性数据模型。为了能实现自动化的安全性分析，需要定义出故障树分析和安全隔离分析所需的数据，如表 1 所示。

表 1 安全性分析数据要求

编号	数据	类型	描述
1	Id	String	标识
2	Name	String	名称
3	Premise	EANode	驻留节点
4	Parent	List < String >	源链接
5	Child	String	目标链接
6	Availability	Float	可用性
7	Integrity	Float	完整性
8	Level	String	模块硬件层级
9	Area	RectangleF3D	位置及体积信息

其中架构相关的属性（如表 1 中 1 ~ 5 项）在架构建模中已定义，安全性数据（表 1 中 6 ~ 9 项）在架构建模时未独立建立模型，需对其进行进一步描述。本文采用标签值来描述分析所需的安全性数据模型，并通过断言（Assertion）描述输出异常态与输入流之间的关系，增加的标签值如表 2 所示。

表 2 安全性数据模型描述

编号	数据	类型	描述	SysML 描述方法
6	Availability	Float	可用性	标签值：LossOfFunction
7	Integrity	Float	完整性	标签值：Malfunction
8	Level	String	模块硬件层级	标签值：Level
9	Area	RectangleF3D	位置及体积信息	标签值：StationLine 标签值：WaterLine 标签值：ButtockLine

2 自动化故障树分析方法研究

2.1 自动化故障树生成方法

2.1.1 自动化故障树生成原理

自动化故障树生成原理如图 1 所示。在构建 SysML 系统架构模型的同时建立安全性数据模型，并基于这两种模型导出的 XML 文件，进行解析，自动生成故障树，并根据需要选择分析结果进行显示，以支持故障树分析。

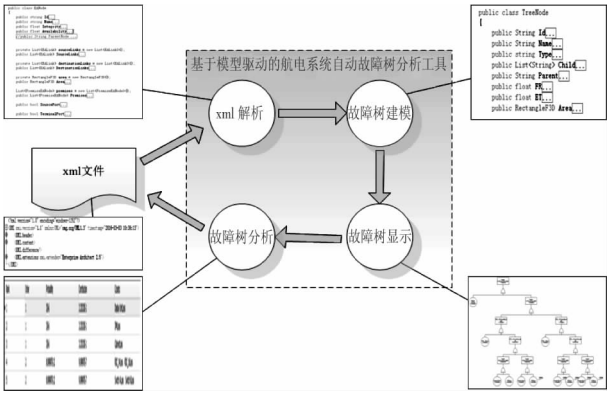


图 1 自动化故障树生成原理

在利用 SysML 模型数据自动生成故障树时，第一步是将层次化结构展开，得到平面化的系统架构模型；其次，查找终端对象的失效模式，确定其为顶事件，并将其与其他对象的失效事件相连；然后利用路径追溯的方法查找故障来源，根据与其他对象的数据流关系得到失效原因的可能组合，依此自动建立顶事件故障树。基于 SysML 的航电系统架构模型数据自动生成故障树的过程，只是按需查找失效原因，只关心与顶事件相关的其他失效事件。

基于上述自动生成故障树基本思想，构建故障树的过程就可以理解为将 SysML 结构转化为故障树结构的过程。在故障树建模时，首先需要将安全性数据模型注入到 SysML 模型中，注入的过程就是基于这两个模型，确定元素的故障模式、故障概率以及故障转移方式。其中，故障模式分为丧失（Loss）和错误（Error）两大类。最后，将注入的结果整理成故障树的形式即可完成故障树建模。

本文所述方法采用的是路径追溯的故障树建模方法，即从终端对象的故障模式开始，根据数据传播的路径逆推可能造成这种故障的原因，由此完成注入的过程。故障树建模的基本单元可以描述为：对于某模块，其功能故障原因必为其本身故障

2017年第3期 总126期
ISSN 1674-9804 CN 31-1614/V

或数据流输入故障,则有对如图 2 所示的 SysML 基本单元,建立如图 3 所示的故障树基本单元。

图 2 SysML 模型中的基本单元

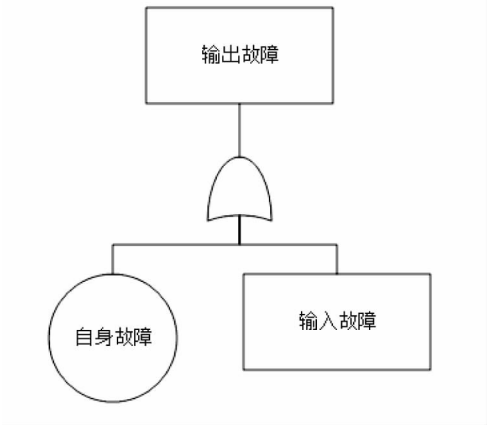


图 3 故障树基本单元

2.1.2 自动化故障树生成算法

基于自动化故障树生成原理,故障树生成算法如下:

步骤 1,如果系统架构模型为层次模型,将其展开,使得模型中不包含可以被继续划分为其他元素的元素。

步骤 2,确定故障树顶事件:查看 Package 元素中的相关描述,根据失效模式标签 LossOfFunction 和 MalFunction 确定顶事件。

步骤 3,针对 LossOfFunction,做如下操作:

1)对于某对象 A,构造如图 4 所示的故障树基本单元。其输出丧失由或门连接,故障原因为 A 本身丧失,即所对应的 A 功能丧失(如果已定义了 LossOfFunction),以及 A 的输入丧失。

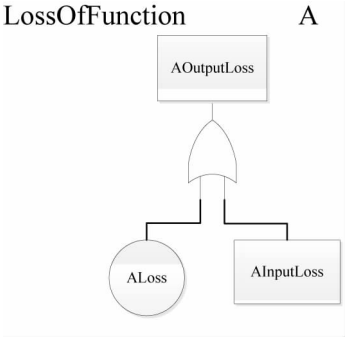


图 4 丧失类故障树基本单元

2)对于某对象 A 的输入丧失,若其数据流只来源于对象 B,则该事件由或门连接,并且其中一个故障原因是 B 丧失,另一个原因是 B 输入丧失。

3)对于某对象 A 的输入丧失,若其数据流来源于多个对象,查看 A 与其他对象数据流关系,查找对个对象对丧失模式的表达式,将其映射为故障树;如果不能找到相应的表达式,则默认这几个对象是冗余备份的关系,用与门连接它们的输出丧失事件。

4)对于没有数据流来源的对象 A,用 A 本身丧失代替 A 的输入丧失和 A 的输出丧失。

步骤 4:针对 MalFuntion 的操作与步骤 3 类似,只是当某元素的数据流来源于多个元素时,如果不能找到相应的 Assertion 表达式,则用或门连接它们的输出丧失事件,因为即使是冗余备份关系,任意一个输入模块错误均会导致该模块发生错误。

2.2 自动化故障树分析方法

本文在故障独立假设前提下完成了故障树建立,为了保证该假设成立,除了对故障树进行基本的定性及定量分析,还需要对其进行共模分析和区域安全性分析。

2.2.1 故障树分析

本文通过开源故障树分析工具 XFTA 进行故障树分析,将生成的故障树文件转化为符合 OPENPSA 标准的 xml 文件,然后将该 xml 文件作为 XFTA 的输入,调用 XFTA 计算算法,计算每个中间事件的最小割集和概率,同时求出结构重要度、概率重要度和关键重要度。

2.2.2 共模分析

自动化共模检查是为了支持共模分析,进一步分成最小割集、共模检查单和共模事件检查三个部分。

1)最小割集

基于故障树割集计算算法,可以计算出某一个中间事件的割集,因此可以通过选择故障树中的某一个中间事件,将最小割集和一阶最小割集通过可视界面显示出来。

2)共模检查单

基于 ARP4761 中对于共模源的分类,将共模分析检查单和记录表通过可视界面显示出来,用于检

查两个硬件类事件的共模源和错误，以支持自动化共模分析。

3）共模事件检查

基于生成的故障树结果，可以给出一个“与”门事件下不同分支中重复出现的相同硬件底事件，并通过可视化界面显示出来，结合最小割集以及共模检查单实现自动化共模分析。

2.2.3 区域安全性分析

自动化安全隔离检查的目的是为了支持区域安全性分析，主要包括交集检查、危险区域检查和隔离距离检查三个部分。

1）交集检查

交集检查是用于判断故障树中两个硬件基本事件是否存在重叠区域。基于故障树生成结果，识别出硬件类型的基本事件，并结合安全性数据模型中的位置和体积信息，生成两个基本事件的立体模型，通过两个立体模型的空间位置，判断这两个硬件类型的基本事件是否存在重叠区域。

2）危险区域检查

危险区域检查是用于判断故障树中同一“与”门下的两个硬件类型的割集事件是否落入同一个危险区域。基于故障树生成结果，识别出同一“与”门下的两个硬件类型的割集事件，并结合安全性数据模型中的位置和体积信息，生成两个基本事件的立体模型，然后基于人工输入危险区域信息（位置和边长），生成危险区域立体模型，通过两个割集事件立体模型的空间位置对于危险区域立体模型的空间位置，判断这两个硬件类型的割集事件是否落入定义的同一危险区域。

3）隔离距离检查

隔离距离检查是用于判断故障树中同一“与”门下的两个硬件类型的割集事件是否满足距离要求。基于故障树生成结果，识别出同一“与”门下的两个硬件类型的割集事件，并结合安全性数据模型中的位置和体积信息，生成两个基本事件的立体模型，计算两个割集事件立体模型空间位置之间的距离，然后基于人工输入距离要求信息，判断这两个硬件类型的割集事件是否满足指定距离要求。

3 自动化安全性分析工具设计

3.1 自动化安全性分析工具软件架构

自动化安全性分析工具架构如图5示，基于EA架构模型XML文件，实现自动化安全性分析。

图5 自动化安全性分析工具架构

3.2 系统架构分析模块

系统架构分析模块的功能是从EA架构模型的XML格式文件中提取信息，转化成“EANode”类，如表3所示，由故障树建模模块直接利用建立故障树。该模块利用Linq到XML的编程接口来分析EA架构模型的XML文件。

表3 EANode类属性描述

EANode属性	类型	描述	XML相关信息
Id	String	唯一的标识符	
Name	String	名称	
Availability	Float	用于“功能丧失”的概率计算	标签值：LossOfFunction
Integrity	Float	用于“功能错误”的概率计算	标签值：Malfunction
Level	String	单元层级	标签值:Level
Premise	EANode	驻留节点	架构图属性
SourceLinks	List<EALink>	关联数据流输入列表	架构图属性
DestinationLinks	List<EALink>	关联数据流输出列表	架构图属性
SourcePort	Bool	源节点	架构图属性
TerminalPort	Bool	目标节点	架构图属性
Area	RectangleF3D	位置和体积信息	标签值:StationLine, WaterLine, ButtockLine

3.3 故障树建模模块

故障树建模模块的功能是将系统架构分析模块提取出来的“EANode”类属性转化为故障树结构的“TreeNode”类属性，“TreeNode”类属性定义如表4所示。在转化过程中，通过分析“EANode”类属性中各个对象之间的相关关系，产生故障树结构数据，同时提取出概率计算需要的信息。

2017年第3期 总126期
ISSN 1674-9804 CN 31-1614/V

表 4 TreeNode 类属性描述

TreeNode 属性	类型	描述
Id	string	唯一的标识符
Name	string	故障树节点(Gate/Event)的名称
Type	string	故障树节点(Gate/Event)的类型,例如:中间事件、基本事件、与门、或门
Child	List < string >	子节点 Id 列表
Parent	string	父节点 Id 列表
FR	float	失效率
ET	float	暴露时间
Area	RectangleF3D	位置与体积信息
Level	string	硬件层级,例如 LRU 或 LRM 等

3.4 故障树图形模型与用户接口模块

故障树图形模型与用户接口模块主要实现以下两个功能:

1) 故障树的图形表达。根据故障树建模模块建立的故障树,采用标准故障树图形符号将故障树表示并显示出来。

2) 用户接口。主要包括三个部分:(1) 用户可以手动对故障树进行修改和编辑;(2) 用户可以通过点击图形和标签选择显示故障树计算的结果;(3) 用户可以通过输入安全性要求,获得安全性要求检查结果。

故障树的图形表达是通过将"TreeNode"类属性与"Northwoods. go, Northwoods. Xml, Northwoods. go. Layout"类库进行关联后实现故障树的显示。

故障树计算结果的输出功能通过结果文件的读取操作实现,并直接显示在故障树图形界面中。

3.5 故障树计算模块

故障树计算模块主要基于 XFTA 开源引擎实现故障树概率计算,通过调用 XFTA 开源故障树分析软件来计算故障树顶事件的概率。XFTA 是故障树评估引擎,包含计算概率、割集等的高效算法。其中,"深度优先调用"功能用于获得故障树的各中间事件概率值和最小割集。

3.6 安全性分析模块

安全性分析模块根据故障树计算模块的结果,进行共模和安全隔离要求的检查,并通过列表形式显示出来。

1) ErrorList:列出了故障树模型结构中的错误,例如或门必须有两个或多于两个子节点;

2) IntersecCheck:列出了故障树中所有基本事件实体之间存在两两交互的情况;

3) MiniCutSet:根据选中的顶事件或中间事件,列出该事件的所有最小割集;

4) 1-MiniCutSet:根据选中的顶事件或中间事件,列出该事件的所有一阶最小割集;

5) CMAChecklist:列出同一与门下的割集事件,通过检查单方式记录两两事件之间的共模检查结果及需求,从方案与设计、制造、安装/综合与试验、操作、维修、测试、校准、环境八个方面进行检查,每个检查结果包括 N/A(不适用)、待分析、派生缓解需求、独立自明性四个方面;

6) CMEventList:列出同一与门下不同分支中重复出现的基本事件;

7) HazardAreaCheck:在属性区"Zones"标签下定义危险区域,包括中心坐标、长、宽、高,再重新计算,则会列出同一与门下割集事件实体是否落入该定义区域中;

8) DistanceHazard:列出同一与门下割集事件的某一层级实体距离定义的危险区域的距离;

9) SeparationCheck:列出同一与门下割集事件的某一层级实体之间距离。

4 案例分析

为了验证所设计的安全性分析工具的有效性,针对如图 6 所示的系统 EA 模型,采用上述工具进行故障树自动建模与分析。

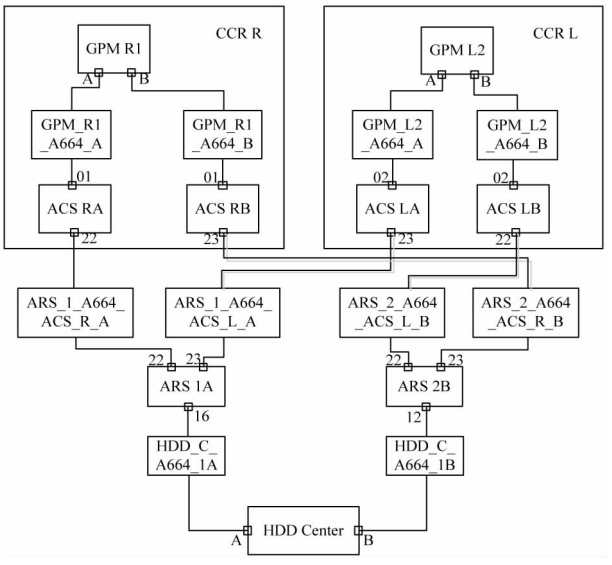


图 6 EA 模型

4.1 故障树建模

在自动化安全性分析工具中生成该 EA 模型的 XML 文件，并导入。安全性分析工具经过后台计算，将 XML 文件直接转化为故障树，如图 7 所示。

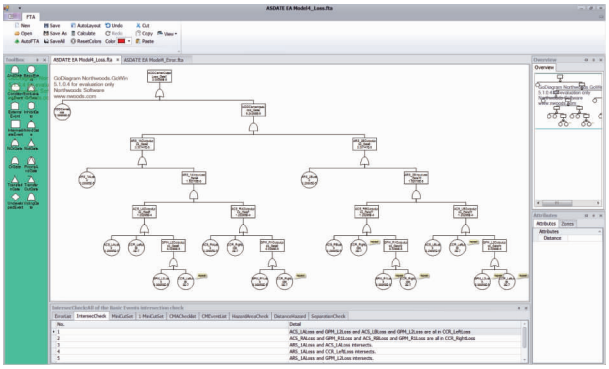


图 7 自动生成故障树结果

4.2 故障树分析结果

针对每一个模型，工具会生成一个“Loss”类型和一个“Error”类型失效状态的故障树，分别显示在两个标签中，同时在右侧的“概要区”中显示整个故障树的情况，如图 7 示。在显示的故障树中，同时也在顶事件、中间事件和基本事件名称下方显示了失效概率。

共模和安全隔离要求检查结果如下：

1）ErrorList 结果

在本例中不存在模型错误，因此 ErrorList 为空，如图 8 所示。

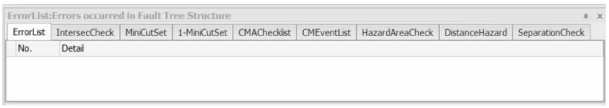


图 8 ErrorList 结果

2）IntersecCheck 结果

在本例中，“Loss”类型的故障树中检查出 6 个存在交集情况的结果，如图 9 所示，包括了包含（前 2 个）和交集（后 4 个）两个类型。

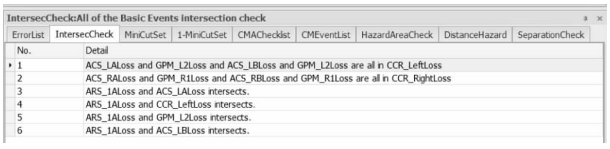


图 9 IntersecCheck 结果

3）MiniCutset 结果

选择“Loss”类型故障树的顶事件，检查出 21 个最小割集结果，如图 10 所示，包括了 1 个一阶割集、5 个二阶割集、14 个三阶割集和 1 个四阶割集，并给出了该割集的失效概率以及对顶事件失效概率的贡献（Contribution）。

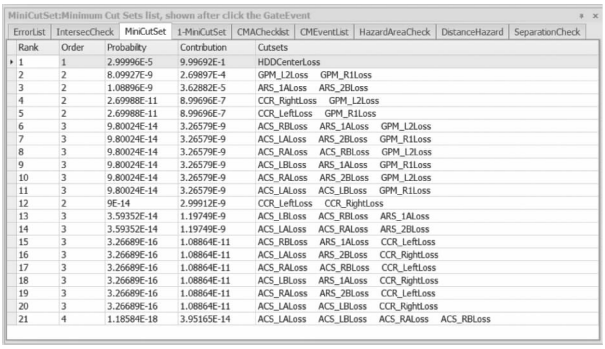


图 10 MiniCutset 结果

4）1-MiniCutset 结果

根据 3）的结果，“Loss”类型故障树的顶事件只有一个割集，如图 11 所示。

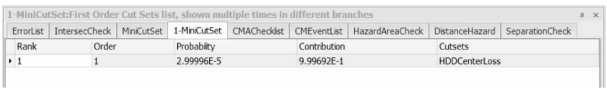


图 11 一阶最小割集结果

5）CMAChecklist 结果

根据 3）的结果，列出同一与门下的割集事件两两事件，如图 12 所示，针对每两个事件进行共模检查和记录，评估是否需要派生需求。

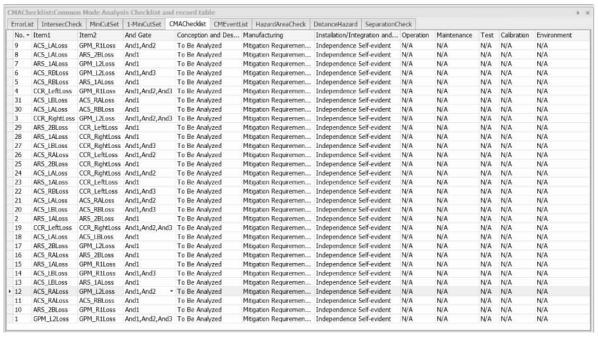


图 12 CMAChecklist 结果

6）CMEventList 结果

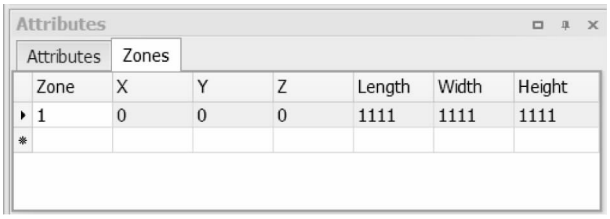
在本例中不存在事件在同一与门下不同分支中重复出现，因此 CMEventList 为空，如图 13 所示。

图 13 CMEventList 结果

2017年第3期 总126期
ISSN 1674-9804 CN 31-1614/V

7) HazardAreaCheck 结果

在进行危险区域检查前,需要先定义一个危险区域,在本例中定义了一个区域1,如图14所示,中心位置为(0,0,0),长宽高分别为1111、1111和1111的立方体区域。定义完危险区域后,进行重新计算,则会在"HazardAreaCheck"标签下显示,如图15所示,共有38条,每条的"Cutsets"一列列出了同一与门下处于同一危险区域的所有割集,"Detail"一列说明了这些割集事件同处于哪一个危险区域中,"LRUs"一列归纳了这些割集同属于哪些LRU。



Zone	X	Y	Z	Length	Width	Height
1	0	0	0	1111	1111	1111

图14 危险区域定义

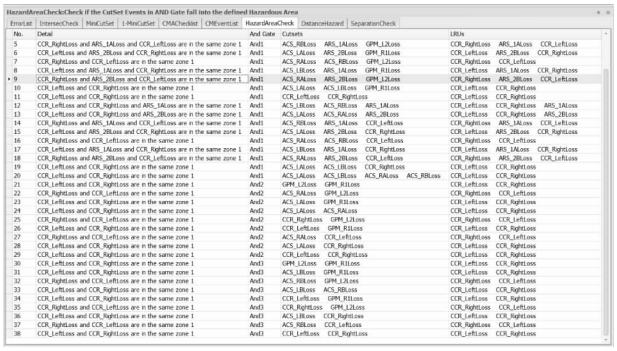
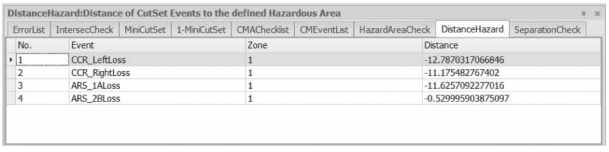


图15 HazardAreaCheck 结果

8) DistanceHazard 结果

在HazardAreaCheck结果基础上,列出了所有处于同一危险区域的LRU事件,并给出了它们距离定义危险区域的距离,如图16所示。

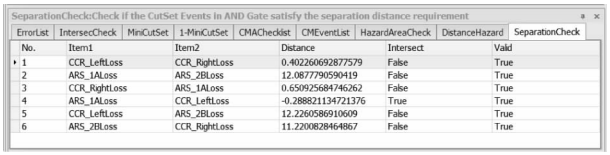


No.	Event	Zone	Distance
1	CCR_LeftLoss	1	-12.787031706684 6
2	CCR_RightLoss	1	-11.175482767402
3	ARS_1ALoss	1	-11.625709227701 6
4	ARS_2BLoss	1	-0.529995903875097

图16 DistanceHazard 结果

9) SeparationCheck 结果

根据3)的结果,列出了同一与门下LRU级割集事件,并计算出两两之间的距离,给出各个LRU所处的矩形区域是否存在交叉的检查结果,如图17所示。



No.	Item1	Item2	Distance	Intersect	Valid
1	CCR_LeftLoss	CCR_RightLoss	0.402260692877579	False	True
2	ARS_1ALoss	ARS_2BLoss	12.0877790590419	False	True
3	CCR_RightLoss	ARS_1ALoss	0.650925684746262	False	True
4	ARS_1ALoss	CCR_LeftLoss	-0.288821134721376	True	True
5	CCR_LeftLoss	ARS_2BLoss	12.2260586910609	False	True
6	ARS_2BLoss	CCR_RightLoss	11.2200828464867	False	True

图17 SeparationCheck 结果

5 结论

1) 本文设计的故障树自动建模方法相比传统手工建树方法效率更高。

2) 本文设计的自动化安全性分析方法与传统方法相比保证了安全性分析与系统设计的同步。

因此无论是从用时长短还是在有效性上,该工具采用的方法都优于传统手工建树方法。

参考文献:

[1] Vesely W E, Goldberg F F, Roberts N H, et al. Fault tree handbook[R]. Washington DC: Nuclear Regulatory Commission, 1981.

[2] Wang G. Integration technology for avionics system[C]// 2012 IEEE/AIAA 31st Digital Avionics Systems Conference. IEEE, 2012:7C6-1-7C6-9.

[3] Li Xiangming, Wu Xuejun. A software design method based on fault-tree analysis[J]. Ordnance Industry Automation, 2011, 30(8): 85-91.

[4] 谷青范, 王国庆,等. 基于模型驱动的航电系统安全性分析技术研究[J]. 计算机科学, 2015, 42(3): 124-127.

[5] Kyle Hampson. Technical Evaluation of the Systems Modeling Language (SysML)[J]. Procedia Computer Science, 2015, 44(1): 403-412.

作者简介:

刘 宇 男,硕士,助理工程师。主要研究方向:航电综合试验。Tel: 021-20864858,E-mail: liuyuchn@sina.com

刘永超 男,研究员。主要研究方向:航电系统综合设计与集成验证。Tel: 021-20864760, E-mail: liuyongchao@comac.cc