Availability Assessment of Avionics Display System based on MBSA using Fault Dependent Matrix

Haiyong Dong\(^1,\)*, Zhiyuan Cao\(^2\), Zhengjun Zhai\(^1\), Qingfan Gu\(^3\), Yanhong Lu\(^1\), Guoqing Wang\(^1,4\) and Miao Wang\(^4\)

\(^1\)School of Computer Science and Engineering, Northwestern Polytechnical University, Xi’an, 710072, China
\(^2\)University of Michigan – Shanghai Jiao Tong University Joint Institute, Shanghai, 200240, China
\(^3\)China National Aeronautical Radio Electronics Research Institute, Shanghai, 200241, China
\(^4\)School of Aeronautics and Astronautics, Shanghai Jiao Tong University, Shanghai, 200240, China

*Corresponding e-mail: donghaiyong@mail.nwpu.edu.cn

Abstract. The availability of avionics display systems plays an important role in flight safety and requires a safety analysis during the design phase. The current safety analysis process is not carried out based on the system design model, so there may be inconsistencies. This paper proposes a model-based safety analysis method using a fault dependent matrix, which can be applied to commonly used system modelling tools such as Enterprise Architect. This article illustrates the utilization of the method by providing the elapsed flight time function, using MATLAB to generate the fault dependent matrix, and then obtaining the availability of the function. This method can be used to aid system safety design and improve system design efficiency.

1. Introduction
The avionics display system is an important system on the aircraft, providing information for pilots. To meet the airworthiness needs, the designation process and the development process of the display system need to follow the 4754 process [1]. During the system development process, safety analysis such as Preliminary System Safety Assessment (PSSA) and System Safety Assessment (SSA) is required, and there are multiple iterations in these processes. In the industry practice, tools such as EA are commonly used for system modelling while Relex is used for safety analysis [2-3]. This process is inefficient and may result in inconsistent system designation models and system safety models.

To solve the above problem, many researchers tried to change the system model directly to make it a system design model with safety properties, known as Model-based Safety Analysis (MBSA) [4]. Dong used Simfia to model the IMA network, and then wrote a program to analyse the availability of the exported model in the XML format [5]. Zhang once studied the safety of Head Up Display (HUD), and construct the safety model based on the system designation model to XML [6]. However, the safety analysis method of the above research is mainly the method of 4761, including Fault Tree Analysis (FTA), Dependence Diagram (DD), Markov Analysis (MA), Petri Net (PN), etc. [7]. Traditional analytical methods can hardly meet the practice needs of system safety analysis. Many
researchers have expanded these methods. Due to October 2016, more than 847 safety analysis methods have been proposed and widely used in nuclear power, aviation, aerospace, etc. [8]. We proposed a safety analysis method based on a fault dependent matrix [9]. This method combines the advantages of FTA and PN methods. It can obtain the cut set using the top-down method and analyze the fault propagation process using the bottom-up method. This method has been expanded, but the model modeling process is always separated from the system design process and easily leads to inconsistencies [10-11].

This paper proposes an availability analysis method that satisfies MBSA based on a fault dependent matrix. The following structure of this paper is as follows. Section II introduces the availability analysis method based on MBSA using fault dependent matrix. Section III takes the availability of the display system as an example to explain the usage phase of the method. The last Section summarizes this paper and proposes the future study.

2. Availability Assessment Method Based on MBSA using Fault Dependent Matrix

Safety analysis mainly includes availability and integrity analysis. The availability analysis is to analyze whether the system or function can provide the service, while the integrity refers to whether the service provided by the system is correct. The quantitative representation of the two is usually expressed by the indicator which represents the service is not available [7]. The safety analysis method based on the fault dependent matrix can be used to construct the safety model and evaluate the quantity probability of the availability of the specified function.

EA is a widely recognized graphical system modelling tool in the industry, its elements (Block, Port, etc.) support adding custom Tagged Value, and the system model can be exported to XML and other formats. Traditional safety analysis method does not perform the safety analysis process based on the system design model. This paper proposes adding safety related attributes directly to the system design model in EA and then perform the availability analysis. The method flow is shown in the figure below.

| Enterprise Architect |
|----------------------|
| (1) Understand the background |
| (2) Add Tagged Value for Block and Port |
| (3) Export model (.xml) in XMI2.0 |

| Matlab |
|--------|
| (4) Generate Fault Dependent Matrix |
| (5) Availability Assessment |

| Fault Dependent Assessment |
|-----------------------------|
| (1) Determine the Safety Requirement |
| (2.1) Determine the MTBF of Block |
| (2.2) Determine the Failure Mode of Block and Port |
| (2.3) Assign the Failure Mode State of Block and Port |
| (2.4) Determine the Fault Causal Relationship |

![Figure 1. Availability assessment process based on MBSA using fault dependent matrix.](image)

As shown in figure 1, the safety requirement is the goal of safety analysis, that is, to determine whether the system designation meets the safety requirement of the system. If so, then continue; if not, the system designation needs to be revised. Adding the safety information to the system design model in EA directly can ensure the consistency between the safety model and the system designation. The Tagged Value added to the Block and Port in the model is slightly different, as shown in the table 1.

| Tagged Value | Description | Detail | BlockPort |
|--------------|-------------|--------|-----------|
| FailureMode  | The Failure Mode of the Type=String; component | Default=Loss of the ability to do sth. | ✓ b ✓ |
| FailureModeState | The working status of the Type=Enum; | | ✓ |
component Values=Normal, Loss; Default=Normal; Type=Custom; Mask=ddddDD; Template=___,___FH;

The Mean Time Between Failure of the component MTBF

The Fault Causal Relationship among the Failure Mode FaultCausalRelationship

Failure Mode

| Default=FM(device) AND/ORFM() LEADFM(self); |
|---------------------------------------------|

Each Tagged Value could be predefined its structure type before utilization [12].

☑ means the Tagged Value should be defined in the element.

☒ means the Tagged Value should not be defined in the element.

The fault causal relationship describes the causal relationship between failure modes. In this paper, all cause relationships are static fault causal relationships, which can be expressed as causality. The failure mode on the left side of LEAD can be device and port, and the self on the right side indicates the failure mode status of the port.

Suppose a port output is normal, if and only if the device and port inputs are normal, then the fault causal relationship will be expressed as FM(device) OR FM(inPort) LEAD FM(self), which can be expressed as fault dependent matrix as follows.

\[
M_{FPM} = \begin{bmatrix}
V_{FPM,1} \\
V_{FPM,2}
\end{bmatrix} = \begin{bmatrix}
-1 & 0 & 1 \\
0 & -1 & 1
\end{bmatrix}
\]

(1)

Since the fault causal relationship is an OR relationship, it is split into multiple fault causal relationships. Where -1 indicates the source of the fault, 1 indicates the affected state of the fault, and 0 does not have meaning.

After obtaining the fault dependent matrix, the fault dependent matrix algorithm can be used to obtain the cut set of the failure condition. The cut set is composed of several device failure modes, and then the quantitative evaluation of the failure condition is obtained.

3. Case Study

3.1. Avionics Display Model

The current avionics display system of civil aircraft is based on the IMA architecture. The RDIU is responsible for collecting information such as sensors. The GPM with multiple hosted functions is responsible for processing information. The Switch is responsible for transferring data. The IDU is responsible for displaying the data in the form of graphic images for pilots. The examples in this paper have been simplified to illustrate the utilization of the method. The simple display system design architecture for displaying “Elapsed Flight Time” is shown in figure 2.

The Elapsed Flight Time displaying function collects the UTC Time provided by the flight management system and the wheel signal of the landing gear. After processing by GPM, the Elapsed Flight Time has been obtained and displayed to the pilot.

3.2. Assumptions

To simplify the system model, this paper takes several common assumptions.

- Failure will only occur on the devices and will not occur on external systems, cables, and ports;
- The failure mode state of the device can only be classified into two categories, one is Normal and the other is Loss;
- The devices are independent of each other, and the failure rate remains unchanged during use.

3.3. System Model with Safety Tagged Value
Add safety information to the model in figure 2, that is, assign the value to the Tagged Value of Blocks and Ports, as shown in table 2 and table 3.

**Figure 2.** Physical architecture of display function “Elapsed Flight Time”.

**Table 2.** Function and safety information of all Blocks in the model.

| Component | Function       | Failure Mode (Device Type)          | FM No. | MTBF       |
|-----------|----------------|-------------------------------------|--------|------------|
| GPM       | Data Process   | Loss of the ability to process data | 1      | 50,000FH   |
| IDU       | Data Display   | Loss of the ability to display info | 2      | 16,000FH   |
| RDIU1     | Data Source collection | Loss of the ability to provide data | 3      | 14,000FH   |
| RDIU2     | Data Source collection | Loss of the ability to provide data | 4      | 14,000FH   |
| Switch    | Data Transfer  | Loss of the ability to transfer data| 5      | 100,000FH  |

All failure modes (FM) are numbered in table 2. The failure modes and fault causal relationship (FCR) are also numbered in the following table. These numbers are not stored in Tagged Value. All FMs and FCRs are numbered by MATLAB programs.

**Table 3.** Function and safety information of all Ports in the model.

| Port      | Failure Mode (Function Type)                                      | FM No. | Fault Causal Relationship                  | FCR No. |
|-----------|------------------------------------------------------------------|--------|--------------------------------------------|--------|
| GPM: inPort1 | Loss of the ability to obtain source data from Switch            | 6      | FM(Switch: outPort1) LEAD FM(self)          | 1      |
| GPM: outPort1 | Loss of the ability to provide source data to GPM               | 7      | FM(device) OR FM(inPort1) LEAD FM(self)     | 2      |
| IDU: inPort1 | Loss of the ability to obtain processed data from Switch        | 8      | FM(Switch: outPort2) LEAD FM(self)          | 3      |
| IDU: outPort1 | Loss of the ability to provide information (Elapsed Flight Time) for the crew | 9      | FM(device) OR FM(inPort1) LEAD FM(self)     | 4      |
| RDIU1: inPort1 | Loss of the ability to obtain source data (UTC Time)            | 10     | / a                                        | /      |
| RDIU1: outPort1 | Loss of the ability to provide source data to Switch         | 11     | FM(device) LEAD FM(self) b                  | 5      |
| RDIU2: inPort1 | Loss of the ability to obtain source data (WeightOnWheel)      | 12     | /                                          | /      |
Since external systems are assumed to keep working, the INPUT of both RDIUs could keep normal.

\(^a\) Normally, the state of the output depends on the input, but since the input is always normal, it is simplified.

### 3.4. Fault Dependent Matrix Generation

MATLAB parses the XML file describing the safety model, obtains all the failure modes and fault causal relationships of the system, and then expresses the fault causal relationship one by one in the matrix. Take FCR#1 as an example. The fault causal relationship of GPM: inPort1 is FM(Switch: outPort1) LEAD FM(self), the FM number of Switch: outPort1 is 17, and the FM number of GPM: inPort1 is 6. Then the fault dependent matrix is shown as follows.

\[
V_{FPM,1} = \begin{bmatrix} \ldots & 1 & \ldots & -1 & \ldots \end{bmatrix} \tag{2}
\]

Similarly, taking FCR#2 as an example, the fault causal relationship of GPM: outPort1 is FM(device) OR FM(inPort1) LEAD FM(self), the FM number of GPM is 1, and the FM number of GPM: inPort1 is 6. GPM: The FM number of outPort1 is 7, and the fault dependent matrix is shown as follows.

\[
\begin{bmatrix} V_{FPM,2} \\ V_{FPM,3} \end{bmatrix} = \begin{bmatrix} -1 & \ldots & 0 & 1 & \ldots \\ 0 & \ldots & -1 & 1 & \ldots \end{bmatrix} \tag{3}
\]

Since the rows of the same fault cause can be merged, the fault dependent matrix of the system is obtained as follows.

\[
M_{FPM} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{bmatrix} \tag{4}
\]
There is a -1 and two 1 in the 11th line of equation (4), indicating that the loss of the switch can cause loss of both outport1 and outport2.

3.5. Analysis Results
As shown in table 2, the MTBF of the GPM is 50,000 FH, and the failure rate can be assumed to be constant 2E-5 / FH.

The function of this model is providing the “Elapsed Flight Time” for pilots. The failure mode FM#9 is the failure condition that needs to be analysed. Using the top-down analysis method, the cut set of failure modes that lead to the failure condition can be obtained, and the probability of failure condition is 2.354E-4 / HF. Based on the result, it is determined whether the safety requirement is met, and if it is not satisfied, the system designation needs to be improved.

4. Conclusions
This paper introduces an availability assessment method based on MBSA using fault dependent matrix, which ensures the safety model constructed could be consistent with the system designation model. Taking the availability analysis of providing Elapsed Flight Time for the pilot as an example, the EA model is built, the fault dependent matrix is generated by MATLAB program, and the availability is calculated, which can be to verify whether the safety requirements are met by the designed system.

However, this method is unable to deal with the availability analysis of the system design model with fuzzy fault causal relationship and time-related dynamic fault causal relationship, which will be further studied in the future.

Acknowledgments
The authors would like to thank Hao Rong for his technical assistance in the construction of the case model, Zilong He for his advice and assistance in proofreading the article.

References
[1] ARP4754A 2010 Guidelines for Development of Civil Aircraft and Systems (Pennsylvania: SAE International)
[2] Enterprise Architect. Available online: https://sparxsystems.cn/ (accessed on 29 August 2019)
[3] PTC Windchill Quality Solutions Software (Formerly Relex Software). Available online: http://www.relexsoftware.it/ (accessed on 29 August 2019)
[4] Gu Q, Wang G and Zhai, M 2014 Proc. of the 14th AIAA Aviation Technology, Integration, and Operations Conference (Atlanta: AIAA) p 2226
[5] Dong H, Gu Q, Wang G, Zhai Z, Lu Y and Wang M 2019 Processes 7 p 117
[6] Zhang F and Dong H 2019 3rd Int. Conf. on Circuits, System and Simulation (Nanjing: IEEE) pp 116-120
[7] ARP4761 1996 Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment (Pennsylvania: SAE International)
[8] Everdij M and Blom H 2016 Safety Methods Database. Version 1.1 (Netherlands: Netherlands Aerospace Centre NLR)
[9] Dong H, Gu Q, Wang G, Zhai Z, Lu Y and Wang M 2019 Int. J. Performability Eng. 15 p 2392-2399
[10] Dong H, Gu Q, Wang G, Zhai Z, Lu Y and Wang M 2019 9th Int. Conf. on Electronics Information and Emergency Communication (Beijing: IEEE) pp 1-5
[11] Dong H, Gu Q, Wang G, Zhai Z, Lu Y and Wang M 2019 2nd Int. Conf. on Artificial Intelligence and Pattern Recognition (Beijing: ACM Press) pp 196-200
[12] Predefined Structured Types. Available online: http://tool.uml.com.cn/ToolsEA/UserGuide/modeling_tools/predefinedtaggedvaluetypes.html (accessed on 29 August 2019)