IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 12, 2021, accepted May 5, 2021, date of publication May 10, 2021, date of current version May 19, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3078813

Equipment Mission Safety Evaluation Method Based on Function-Structure

YUJIN CHEN, JIHUI XU, LIJUAN KAN, JIAHUI SHI, AND WENJIE TIAN
Equipment Management and UAV Engineering College, Air Force Engineering University, Xi'an 710051, China

Corresponding author: Yujin Chen (ivan_safety@foxmail.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 52074309.

ABSTRACT Safety is the eternal theme of aviation activities. With the rapid development of military technology and equipment construction, the new equipment requires safer and more reliable technology and management guarantees. The aim of this paper is to evaluate the safety level of equipment in a single-stage mission. The mission safety is defined as the capacity of equipment to avoid unacceptable accidents during mission execution. A safety analysis framework is constructed according to the logic of mission-function-safety structure. The mission profile ensures the success of the mission by specifying the functional requirements and the safety structure realizes the functional requirements by adjusting the structure and scheduling resources. On this basis, the functional dependency network analysis (FDNA) method is improved to study the interaction failures between the components in the specific state. A numerical simulation method for mission safety is proposed to analyze the evolution relationship between components and behaviors, which could obtain the mission safety level under different failure modes. A case verifies the application process. The results revealed that the mission safety assessment needs to be evaluated with the consideration of safety structure and mission profile. The scale of system failures conforms to Weibull distribution in different attack modes. It is shown that the escalating risks will cause serious consequences.

INDEX TERMS Mission safety, mission profile, functional dependency network analysis, function-structure.

I. INTRODUCTION

The safety of equipment is a complex systemic problem [1]. Its internal structure, interface interaction and function combination are intricate. It is influenced by multiple factors such as liveware, environment, software and hardware. It is also influenced by the interaction between these factors [2]. On one hand, due to the inability to estimate all possible mission requirements at the design stage, the mission complexity and risk uncertainty [3] are underestimated, resulting in unacceptable safety accidents in specific mission environment. On the other hand, even if they meet safety the design standards, there will be ''acclimatized'' to expose the potential risks unexpected due to the external environment, personnel level and other objective constraints. Therefore, to ensure the bottom line of safety is to ensure the safety [4], [5] of the ''liveware-environment-software-hardware'' [6], [7] system during mission execution.

The associate editor coordinating the review of this manuscript and approving it for publication was Rosario Pecora.

Traditional models attribute the causes of ''unsafe'' problems to accident chains, and express the process of ''unsafe'' problems as a series of discrete events that occur in a specific time sequence. Failure Mode and Effects Analysis(FMEA) [8]–[10], Fault Tree Analysis (FTA) [8] and other models belong to this category. This type of model is convenient for people to understand the entire accident chain from the top [11], and is more suitable for analyzing accidents caused by the failure of physical components or human errors in simple systems [9]. However, due to linearity and over-mechanism [12], such models cannot demonstrate the specific details of the ''unsafe'' problem, especially systemic and interactive information. They also cannot systematically explain how the accident occurred. The models are not suitable for large and complex systems [12]. In addition, this type of model lacks the criteria for judging the initial event and is highly arbitrary. People are prone to misjudge initial events due to cognitive limitations.

In recent years, modern accident models have emerged. Modern accident models are also called accident models based on system theory [12]–[14]. Such models consider

multiple factors lead to the occurrence of ''unsafe'' problem in the joint action at a specific time. Rasmussen's method [14] based on a hierarchical social technology framework and Leveson's [12], [13], [15] System Theoretic Accident Modeling and Process (STAMP) are typical representatives of such models. The fundamental difference between the modern model and the traditional model is that the accident is regarded as a kind of safety emergence failure. Accidents are caused by inappropriate interactions between system components. The occurrence of ''unsafe'' problems in complex systems is not only caused by component failure at the micro level, but also caused by inappropriate interaction between components. The ''unsafe'' problems caused by system interactions often have the characteristics of high concealment and destructiveness, which seriously restricts the level of equipment intact rate and the improvement of use support efficiency [1], [2], [16]. In particular, as mission types and equipment application scenarios become more diverse, new technologies have been added to the equipment, and new roles such as ''liveware'' and ''environment'' have been introduced. This type of ''unsafe problem'' is becoming more common.

Therefore, the analysis and evaluation of the mission safety cannot be limited to the equipment itself, it is necessary to comprehensively analyze the interaction between liveware [17], environment [7], [18], software, and hardware [6], [7]. These interactions have clear controlling party and controlled party, with significant topology nature [15]. So that the model can be built by the network. Sohag *et al.* [19] combined Hip-hop with Petri net and Bayes net to determine the sequence of different failure event combinations, and extended the results to dynamic reliability analysis. In order to characterize human factors more vividly, Zhou *et al.* [20] used fuzzy logic and Bayes network to improve the cognitive reliability and error analysis method (CREAM) and proposed a method for quantitative analysis of human reliability. Wang *et al.* [21] combined GERT network and opportunity theory to study the risk transmission mechanism of complex equipment system and applied it to the risk analysis of fighter mission execution. Shuang *et al.* [22] used the complex network to simulate the water pipe network, studied its reliability analysis, and analyzed the fragile components of the water supply system through evolutionary relationships. The above methods make full use of the topological nature to describe the ''unsafe'' problem, but they only focus on one or several factors that lead to the occurrence of the ''unsafe'' problem, and lack the ability to analysis the ''unsafe'' problem from a higher level. On the contrary, various extended models developed from the STAMP model have obvious topological properties. They can fully consider the role of various factors in the emergence failure, but they lack a complete mathematical basis. For example, Dakwat [15] used model checking to perfect the System Theoretic Process Analysis (STPA) method and constructed the control structure of the flight simulator. These frameworks have specific topological properties as other STPA models [13], [18], [23], but its analysis is limited to qualitative analysis and cannot be interpreted in more detail.

Based on the above analysis, the aim of this paper is to provide more insight into the mission safety of equipment by studying the various factors and their inherent topology affecting the safety emergence failure. Main work as follows:

1) Combining the partial ideas of STAMP, a mission safety analysis framework is constructed according to the logic of mission-function-safety structure. The mission profile ensures the success of the mission by specifying the requirements of the function; the safety structure realizes the requirements of the function by adjusting the safety structure and scheduling resources. The safety structure fully considers the various factors and their inherent topology affecting the safety emergence failure.

2) The mission safety analysis framework is formalized by heterogeneous network. To describe the emergence of safety failure, we introduced the Functional Dependency Network Analysis (FDNA) [24]–[29] and made corresponding improvements.

3) A set of numerical simulation method for mission safety is proposed to analyze the evolution relationship between components and behaviors, which could obtain the mission safety level under different failure modes [30].

The results revealed that the mission safety assessment needs to be evaluated with the consideration of safety structure and mission profile. The scale of system failures conforms to Weibull distribution in different attack modes. It is shown that the escalating risks will cause serious consequences.

The remainder of this paper is structured as follows. Section II describes the equipment mission safety analysis framework based on Function-Structure. Section III describes the mathematical description of equipment mission safety analysis framework. Section IV proposes the evaluation method of equipment mission safety. In Section V, an example verifies the application process, and Section VI contains a brief summary and conclusion.

II. EQUIPMENT MISSION SAFETY ANALYSIS FRAMEWORK BASED ON FUNCTION-STRUCTURE

A. MISSION SAFETY

Mission safety refers to the ability in which risks associated with equipment activities, or in direct support of the operation of equipment, are reduced and controlled to ensure the success of the mission at any random moment in the specified mission profile. In short, it refers to the ability of the equipment to avoid unacceptable accidents during mission execution.

Mission safety is for a specified mission profile, not a full-service cycle, which does not include safety problems at the stages of parking and daily maintenance. However, the service conditions at the above stage will affect the safety during mission execution. Through the improvement

of technology and personnel level, the level of mission safety can be improved.

B. FRAMEWORK DESCRIPTION

The research object is single-equipment single-stage mission. It does not consider the logic relationship between missions and the coordination relationship between equipment platforms in cluster operations.

The military activity in this system that is restricted by a complete purpose is called as the mission, such as the unmanned aerial vehicle (UAV) carries out a given reconnaissance or strike mission in a given area. Here we take a typical mission profile of military aircraft as an example, as shown in Figure 1. The mission includes 5 stages: a: takeoff and climb, b: cruise to mission area, c: perform mission, d: exit mission area and return, e: descent and landing. The single-stage mission, the research object of this manuscript, refers to the complete process from take-off to landing, and the execution phase c includes only one mission.

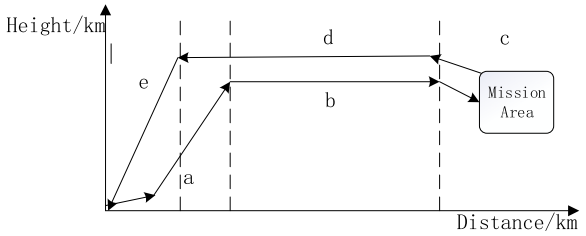


FIGURE 1. Typical mission profile of military aircraft.

The equipment mission safety analysis framework is a complex whole, as shown in Figure 2. The mission profile ensures the success of the mission by specifying the requirements of the function; the safety structure layer realizes the requirements of the function by adjusting the safety structure and scheduling resources. The internal structure of the safety structure layer is intricate and complex, and is affected by the combination of liveware, environment, software, hardware and other components.

In addition to mission profiles and system components, the analysis framework also includes constraints between layers and dependencies within layers. These constraints/dependencies have a clear controlling party and a controlled party, which have obvious topological structure properties. At the same time, the complexity in the safety structure layer determines the heterogeneity of its network modeling. The function layer and the safety structure layer are also heterogeneous. Therefore, the heterogeneous network is introduced to build the equipment mission safety analysis framework.

C. DECOMPOSITION LEVEL OF SAFETY STRUCTURE LAYER

The internal components (software, hardware, environment, and liveware) of the safety structure layer have problems that cannot be directly measured due to fuzzy concepts and boundaries. Therefore, the internal components are taken as the root node and then it should be decomposed layer by layer. It is considered that the decomposition is complete until all the leaf nodes meet the following two requirements:

1) The safety state of the factor can be obtained through direct observation of related indicators or clear calculation methods;

2) The factor can independently complete a single basic function or state with an atomic level.

The following takes the UAV SHEL system as an example to illustrate the decomposition process and granularity. According to the maintenance manual, environment factors can be decomposed into conventional indicators that affect the performance of software and hardware, internal and external environment indicators that affect liveware's decision-making states, and other unexpected risk that can be defined as environment factors. The conventional indicators can be decomposed into temperature, humidity, flight altitude, atmospheric pressure, wind speed, etc. These indicators are directly observable and can independently denote an environment state, so they are the leaf nodes. In the same way, factors such as load operator, EO/IR camera, vertical tail, and attitude sensor all have the ability to independently complete a single function, and can also obtain the safety state through a clear calculation method, so they are also the leaf nodes. If the interaction is lower than the leaf nodes, the internal interface interaction and function combination should be ignored.

Moreover, the granularity of decomposition is up to the mission level which the decision-maker wants to accomplish.

D. DECOMPOSITION LEVEL OF FUNCTION LAYER

Generally speaking, the hierarchy dimensions of the function level and safety structure level should match each other. Meanwhile, the appropriate modeling level about the function level is up to the decomposition process from mission profile to function layer.

Meta-mission [31] is the basic unit of mission profile decomposition. The meta-mission is determined according to the internal logic and operational primitives of each mission stage. Meta-activity is the smallest activity unit during the mission profile execution. It is an intermediate that connects the meta-mission and the meta-combat functions. The decomposition granularity of meta-activities is finer than that of meta-missions. Meta-combat function is the ability of equipment to meet mission requirements under the guidance of combat concepts and missions.

According to the time sequence and logical relationship of each meta-mission, the target mission profile can be decomposed layer by layer until the state of the meta-combat function corresponding to the decomposed meta-activities can be directly observed.

The following takes the mission profile of UAV as an example to illustrate the decomposition process. The mission profile can be decomposed into several meta-missions such as reconnaissance. The execution of reconnaissance can be decomposed into early warning, reconnaissance, positioning, communication, identification and other meta-activities.

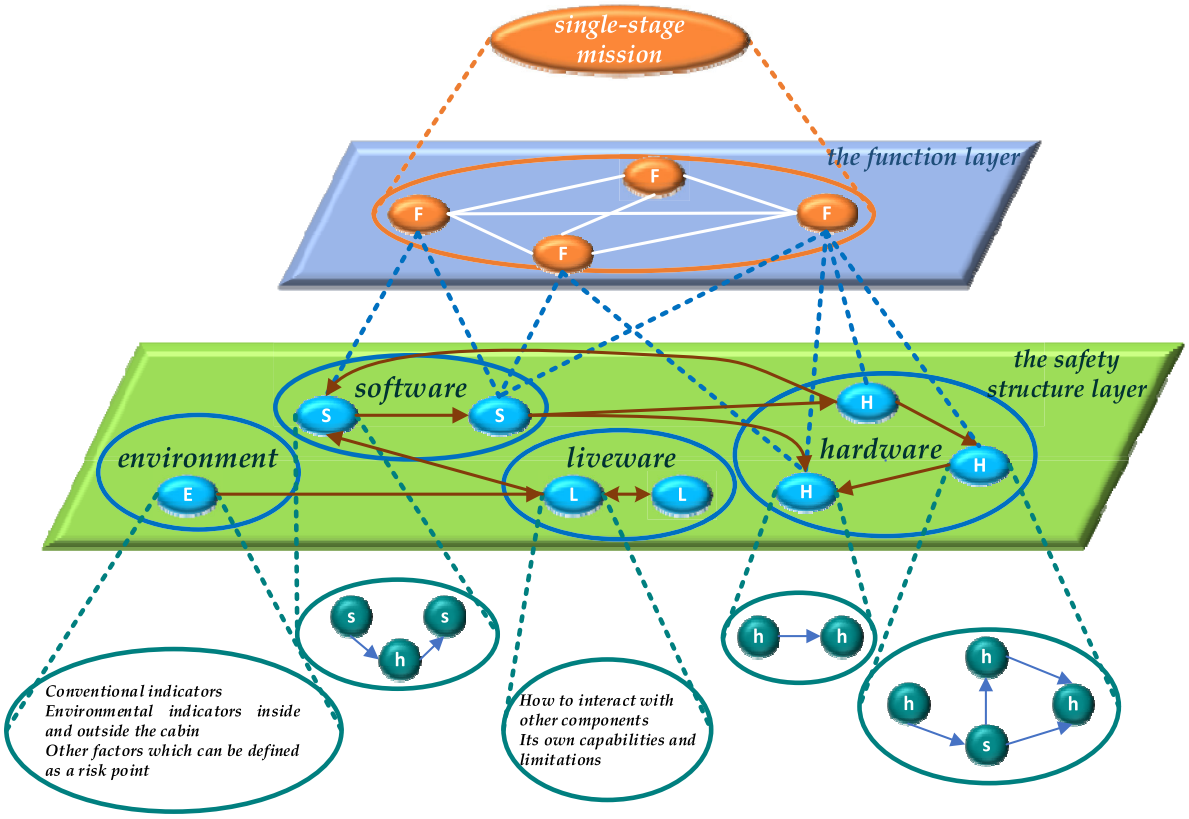


FIGURE 2. Equipment mission safety analysis framework based on function-structure.

These meta-activities require the support of early warning abilities, reconnaissance abilities, communication abilities, navigation abilities, power supply, flight control abilities and other meta-combat abilities. The flight control ability is a meta-combat ability, and its safety state can be directly obtained by docking with the safety structure level. Therefore, the flight control ability is selected as a function in the function layer [2].

III. MATHEMATICAL DESCRIPTION OF EQUIPMENT MISSION SAFETY ANALYSIS FRAMEWORK

A. TOPOLOGICAL REPRESENTATION

The mission safety analysis framework is defined as a quadruple [32] $G = (M, L, V, E)$. Among them, M indicates the complete mission profile, which will be split into a collection of several meta-missions, $M = \{m_1, m_2, \cdots, m_n\}$. Meta-missions are mutually exclusive, arranged in chronological order, and can be regarded as time marks. L indicates the layers of components, $L = \{l_{function}, l_{structure}\}$. $l_{function}, l_{structure}$ represent the function layer and the safety structure layer. V indicates a collection of nodes. The node type mapping function is defined as $\varphi : V \rightarrow V_{type}$. V_{BF}, V_{MF} represent basic function nodes and mission function nodes, V_L, V_E, V_S, V_H represent liveware, environment, software and hardware. E indicates a collection of edges and the edge type mapping function defined as $\varphi : E \rightarrow E_{type}$. $|E_{type}| > 1$.

B. MODELING OF NODES

According to the different roles and functions of components in safety analysis, the nodes are divided as follows,

1) Basic function node V_{BF}: the function that equipment must provide to ensure mission safety, such as providing thrust, fuel and electricity.

2) Mission function node V_{MF}: the specific functions that equipment provides based on mission type and payload, such as perceptual detection, fire control.

3) Liveware node V_L: the personnel who directly controls the equipment during the mission.

4) Environment node V_E: general environmental indicators that affect the performance of software and hardware, environmental indicators inside and outside the operating cabin that affect human physiological state, and other unexpected risk points that can be defined as environmental factors (such as bird strike).

5) Software node V_S: the components that regard logical relationships or passing signals as apparent functions, such as various operating manuals and systems.

6) Hardware node V_H: all kinds of perceptible entities and their physical parameters, such as manipulation devices.

To better illustrate the attributes of the node, two concepts are introduced: Measure of Safety Performance (MoSP) and Measure of Safety Effectiveness (MoSE) [15]. MoSP indicates the physical or functional attribute value output by the node. For example, the temperature that the rudder of an

aircraft can adapt is minus 10 degrees to 45 degrees. MoSE indicates the output by the node can be expressed as value or utility. It also can be called safety operation level (SOL).

In the framework, safety is regarded as an efficiency, and the efficiency of the component system is represented by the SOL of the node. The SOL is the state of node at a certain performance level, and the performance level can be measured by the size of MoSP. Based on the VNM utility theory, it is easy to realize the mapping from MoSP to MoSE. MoSE is generally a dimensionless value from 0 to 100.

The following takes an alarm device as an example to illustrate the mapping process. As shown in the figure 3, if the device issues an alarm within 1s after a threat occurs, MoSE is 100; when the alarm time MoSP = 2s, MoSE = 40. And with the extension of the alarm response time, the safety efficiency is getting lower and lower [16].

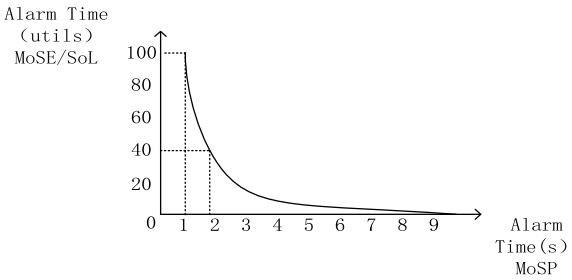


FIGURE 3. The mapping process from MoSP to MoSE.

Based on MoSE and MoSP, All nodes contain two attributes: baseline safety operating level (BSOL) and current safety operating level (CSOL).

BSOL indicates the SOL when all dependencies are unavailable and when the node is working alone. The BSOL of the node in the function layer is determined by the mission type and payload. The BSOL of the node in the safety structure layer is determined by the main technical and tactical indexes of the node.

CSOL indicates the SOL of a node at the current moment, and is determined by the dependency relationship. It is supposed that all nodes can run independently and exert their capabilities relying on BSOL [33], [34], but may not be able to complete the roles and functions that they should play in the system. Most nodes need to exert their normal safety operating capability, mainly relying on the SOL of the nodes that control them.

In addition, the completion of function needs the support of the nodes (software, hardware) in the safety structure layer. Therefore, the SOL of the nodes in the safety structure layer needs to be higher than the SOL of the nodes in the function layer it supports. The CSOL of the function layer node is taken as the minimum safety operating level (Min_SOL) of the safety structure layer node. If the same safety structure layer node supports multiple function nodes at the same time, the largest CSOL is used as the Min_SOL of the safety structure layer node.

C. MODELING OF EDGES

As shown in Figure 4, the equipment mission safety analysis framework contains a total of 15 edge types (two-way edges are regarded as one type). The default edge type is because that it does not exist or has a very low probability of occurrence in practical applications.

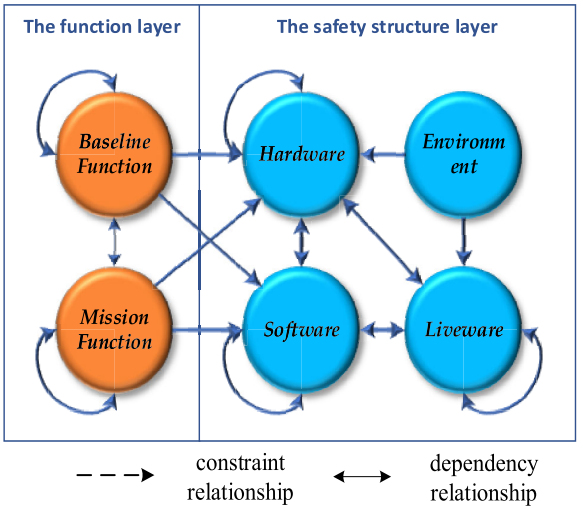


FIGURE 4. The schematic diagram of dependencies.

The edges between layers is the constraint relationship between V_{BF}, V_{MF} and V_S, V_H, which indicates the specific requirements of equipment safety for mission completion. The edges within the layer have clear dependencies for safe operation. The dependency relationship expresses a state between the controlling node and the controlled node. In addition to BSOL, the CSOL of controlled node also includes the SOL passed by the controlling node through the directed edge. At this time, the two have formed a dependency relationship. This dependency relationship is ubiquitous in the mission safety analysis framework, as shown in Table 1. It can be seen from the STAMP that inappropriate dependencies are an important source of ''unsafe'' problems.

Besides its timeliness and health, the edges attribute also include the Strength of Dependency [15], [33], [34] (SOD) and the Criticality of Dependency [15], [33], [34] (COD). Among them, SOD refers to the degree of contribution to the node's SOL, and COD refers to the degree of restriction to the node's SOL.

The following still takes the alarm device above as an example to illustrate the physical/actual meaning of BSOL, CSOL, COD and SOD. If the alarm device A is used alone, MoSP = 2s, and BSOL = 40. Suppose that A appears as a controlled node, and an attitude sensor device B has a dependency on it. B directly affects the response time of A, that is to say, the MoSE of B affects the MoSE that A actually displays in the system. B is the controlling node of A. Also due to B's control, A's MoSE during mission execution is not necessarily equal to BSOL. B's MoSE may restrict the effectiveness of A's MoSE (This is the COD) due to its low

TABLE 1. Description of dependencies.

Relationship	Type	Instruction
V_{BF} - V_{BF}	Two-way	The support/constraint relationship between functions mainly considers the combination of functions in the process of mission completion.
V_{MF} - V_{MF}	Two-way	
V_{BF} - V_{MF}	Two-way	
V_L - V_H	Two-way	Operation relationship between personnel and hardware.
V_L - V_S	Two-way	Information/operation relationship between personnel and various support systems in the cabin.
V_L - V_L	Two-way	Information relationship between personnel in the cabin. It mainly considers the correctness of communication and understanding between personnel.
V_E - V_L	One-way	Environment constraints/support relationships. It mainly considers the influence of environmental factors on human physiology, psychology and decision-making.
V_E - V_H	One-way	The environment supports/constrains the physical parameters of hardware devices.
V_H - V_S	Two-way	It mainly considers the parameter interaction logic, interaction mechanism, control authority at the component interface.
V_H - V_H	Two-way	It mainly involves the operation/physical/logical relationship between devices.
V_S - V_S	Two-way	It mainly involves the logical/physical/operation relationship between software.

sensitivity or other performance limitations; However, B's MoSE may also increase the performance of A's MoSE (This is the SOD) due to its high sensitivity. Here the MoSE during the mission execution is A's CSOL. A's CSOL and BSOL are essentially its MoSEs, but one is possessed by itself, and the other is manifested in the specific safety structure system of the specific mission profile.

D. FDNA-BASED FRAMEWORK MODELING

FDNA [24]–[29] is a system performance evaluation method that can better demonstrate the ''interactive, loosely coupled'' characteristics of the system. It can be used to analyze the chain reaction caused by the failure of a certain system performance to other dependent performance, as well as the possible consequences of the common cause failure of multiple components [26], [27]. It is improved according to the needs of the mission safety analysis framework, which temporal relationships, dependency state, and dependency effects, and modify the physical meaning of relevant parameters are introduced. And it is applied to mission safety analysis for the first time.

A quadruple $D_{ij} = \{DS_{ij}, DE_{ij}, SOD_{ij}, COD_{ij}\}$ is defined to describe the dependency between the controlling node v_i and the controlled node v_j, which is shown in the Figure 5.

1) DEPENDENCY STATE DS_{ij}

DS_{ij} indicates whether the dependency is occurring at the moment. $DS_{ij} = 1$ indicates that the dependency relationship is happening; $DS_{ij} = 0$ indicates that the dependency relationship does not exist at the moment. At this time,

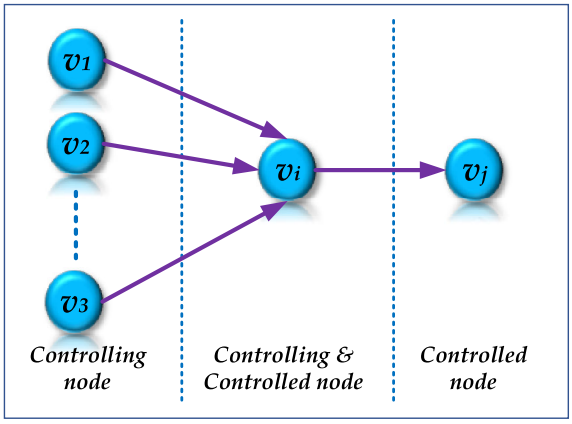


FIGURE 5. The schematic diagram of dependencies.

$SOD_{ij} = COD_{ij} = null.$

$$DS_{ij} = \begin{cases} 1 & m_i \in [m_s, m_e] \\ 0 & otherwise \end{cases} \quad (1)$$

Among them, m_i indicates the executing meta-mission. m_s indicates the meta-mission when the dependency begins. m_e indicates the meta-mission when the dependency ends.

2) DEPENDENCY EFFECT DE_{ij}

DE_{ij} indicates whether the dependency is a positive impact. The positive impact here is ''safe'' or not, depending on the mission.

$$DE_{ij} = \begin{cases} 1 & \text{it is a positive impact} \\ -1 & \text{it is a negative impact} \end{cases} \quad (2)$$

3) THE STRENGTH OF DEPENDENCY SOD_{ij}

SOD_{ij} indicates the degree of contribution to safety operation between v_i and v_j. SOD_{ij} is the indicator that v_j relies on the contribution of v_i to continuously increase its CSOL, and to ensure that v_j can also be safe when v_i is completely operating in a safe state. SOD_{ij} is determined by integrating the current state of the edge, the SOL of the controlled node, and the interface interaction.

For the edge of the function layer, it can be expressed as

$$SOD_{ij} = \alpha_{ij} CSOL_i DE_{ij} + (1 - \alpha_{ij}) BSOL_j \quad (3)$$

Among them, $CSOL_i$ is the current SOL of v_i and $BSOL_j$ is the baseline SOL of v_j. α_{ij} indicates the dependency strength parameter between v_i and v_j. The larger α_{ij}, the stronger the dependence of v_j on v_i, and the greater the contribution of v_i to v_j. It can be expressed as

$$100 (1 - \alpha_{ij}) = BSOL_j \quad (4)$$

For the edge of the safety structure layer, it can be expressed as

$$SOD_{ij} = \alpha_{ij} CSOL_i DE_{ij} MI_{ij} + (1 - \alpha_{ij}) BSOL_j HI_j \quad (5)$$

Among them, the health index [35], [36] HI_j indicates the health of the node v_j, that is, the exerting extent of BSOL when the node works alone. $HI_j \in (0, 1]$. HI_j considers the node's own non-dependency performance change, that is, not caused by the v_i provided. When $HI_j = 1$, v_j fully exert its BSOL; when $HI_j \in (0, 1)$, v_j partially exert its BSOL; when $HI_j = 0$, v_j completely fail. At this time, all v_j's dependency states DS are set to 0. The matching index [15] MI_{ij} indicates the effectiveness of the dependency relationship, that is, the match level of the interactive interface between the nodes. In engineering applications, it can be expressed as: the understanding level of person-to-person communication, permission allocation for manual operation and software independent decision-making. $MI_{ij} \in (0, 1]$. When $MI_{ij} = 1$, the interface is perfectly matched. When $MI_{ij} \in (0, 1)$, the data interaction is not smooth. When $MI_{ij} = 0$, DS_{ij} is set to 0.

4) THE CRITICALITY OF DEPENDENCY COD_{ij}

COD_{ij} indicates the degree of restriction to safety operation between v_i and v_j. It is to express the importance of the v_i's contribution to the v_j's realization of its SOL goals, and to express the restriction degree of v_i's SOL to the v_j's SOL. In other words, while the controlling node supports the controlled node, it also limits the SOL of the controlled node because of the SOL of the controlling node itself.

$$COD_{ij} = MI_{ij} CSOL_i + \beta_{ij} \tag{6}$$

Among them, β_{ij} indicates the criticality parameter between v_i and v_j. The smaller the β_{ij}, the stronger the restriction. It can be expressed as

$$\beta_{ij} = \begin{cases} \frac{1}{h}\sum_{i=1}^{h}\left(SOD_{ij}\right) - \frac{1}{h-1}\sum_{i=1,i\neq j}^{h}\left(SOD_{ij}\right) & h \geq 2 \\ 0 & h = 1 \end{cases} \tag{7}$$

Among them, h indicates the number of nodes that directly depend on the controlled node at the current moment.

Based on the above analysis, v_j's CSOL can be expressed as

$$CSOL_j = \begin{cases} \min\left(\frac{1}{h}\sum_{i=1}^{h} SOD_{ij}, \min_{i=1}^{h} COD_{ij}\right) & \exists v_k, DS_{kj} \neq 0 \\ BSOL_j HI_j & \forall v_k, DS_{kj} = 0 \end{cases} \tag{8}$$

It should be noted that COD_{ij} takes the minimum value because the control effect depends on the node with the strongest control effect under the premise of multiple control nodes. SOD_{ij} takes the average value to express the relationship of multiple control nodes and multiple dependency relationships.

IV. EVALUATION METHOD OF EQUIPMENT MISSION SAFETY

A simplified model of equipment safety structure is selected in the modeling process of mission safety. The simplified model comes from the abstraction of the safety structure with many redundant and multiplexed, and its modeling degree is between the macro model and the micro model.

A. THE SAFETY OPERATING LEVEL OF THE NODE

CSOL is used as an indicator to observe. The CSOL of the nodes in the safety structure layer is transmitted between each pair of nodes along the direction of the dependency relationship, and changes dynamically as meta-mission and topology change. Since the mission safety analysis framework has its own physical meaning, the CSOL is studied with $BSOL_j HI_j$ as the initial value.

B. THE SAFETY OPERATING CAPABILITY OF THE NODE

The CSOL in the network changes dynamically. The following six changes will cause the safety structure layer to update the CSOL of all nodes.

1) A New node is added;
2) Existing nodes failure/function failure;
3) HI_i changes due to the joint effect of its own performance and external factors;
4) A new dependency is generated;
5) the dependency relationship is generated/end due to the change of DS_{ij};
6) MI_{ij} changes due to changes in interface interaction.

The safety operating capability of a node characterizes the SOL that the node can carry when participating in the operation of the network. If the updated CSOL exceeds the safety operating capability of the node, it may be damaged and lose its function, which may cause potential cascading failure of the system and even affect the mission safety.

When $HI_i = 0$, the node v_i fails and it is irreversible for a single-stage unrepairable mission. This state is limited to the nodes of the safety structure layer. It indicates that the node has reliability or safety problems and cannot continue to perform its function. At this time, the node loses the ability to operate safely.

When $CSOL_i < Min\_SOL_i$, the node v_i function failure. Although the performance of the node is not damaged, it cannot support the corresponding function node to complete the mission safely. When $CSOL_i$ is raised above Min_SOL_i, its function is restored again.

C. DETERMINATION OF NODE PARAMETERS

For the nodes of the safety structure layer, operating level indicates the value or effectiveness converted from the physical or functional attribute value output by the node, which is usually a dimensionless value between 0-100.

Among them, the environment node determines its BSOL according to the severity of the environment, which takes the requirements in the design standard and completely unsuitable mission execution as the standards of the efficiency value of 100 and 0. The liveware node determines its BSOL according to the liveware's physiological state, which takes the ability to operate correctly according to the manual and loses the ability to operate as the standard of the efficiency

value of 100 and 0. The BSOL of the software and hardware nodes is calculated based on the indicators specified in the design stage. For example, if a software node completes the calculation and alarms in 0.5s, the BSOL is 100. If the response time is 2s, the BSOL is 25. As the time increases, the lower BSOL. The BSOL of the basic function node and the mission function node is determined according to the type, load, time and environment of mission.

For the determination of HI, the software and hardware nodes are determined according to the ratio of their remaining life to the expected life. The environment and liveware nodes have been considered for their health level in the BSOL, and will not be repeated here, so they are set to 1 as the initial value.

Of course, how to determine the parameters requires the analysis based on specific mission. Sometimes expert determination methods can also be used.

D. EXPRESSION OF UNSAFE DEPENDENCIES

Combining with the STPA method proposed by Levenson, unsafe dependency can be divided into four categories.1) The dependency is not provided; 2) An inappropriate dependency is provided; 3) The dependency is provided too early or too late; 4) The dependency lasts too short or too long.

The performance of the above unsafe dependency in the framework can be divided into the following three categories:

1) An error occurs in DS_{ij}. It means that the dependency should be provided but not provided/ the dependency should not be provided but provided.

2) An error occurs in DE_{ij}. It means that the dependency that should provide the negative influence provides the positive influence/the dependency that should provide the positive influence provides the negative influence.

3) $MI_{ij} \neq 1$. It means that the dependency between nodes cannot be completely matched (unsafe or inappropriate).

E. BASIC ASSUMPTIONS

Based on the above analysis, the following basic assumptions are given:

1) There are three states of the analysis framework: mission completion (the mission profile can be fully executed), equipment is safe but the mission cannot be completed (the basic function node runs normally, the mission function node function fails), and the mission is unsafe (the basic function node function fails).

2) There are two states of the function layer nodes: operating normally (the corresponding safety structure layer nodes are all running normally), and function failure (there are failure/function failure nodes in the corresponding safety structure layer nodes).

3) There are three states of software, hardware, and liveware node: operating normally ($Min - SOL_j \leq CSOL_j$, $HI_j \neq 0$), failure ($HI_j = 0$), and function failure ($CSOL_j < Min\_SOL_j, HI_j \neq 0$).

4) The environment node is not affected by other nodes, and it is stipulated that $HI = 1, CSOL = BSOL$.

5) If and only if the node function is operating normally, the node can participate in the system operation.

6) All dependencies can be implemented in full.

7) In the event of node failure, the system will not return to its original operating state without external interference.

Assumption 3) needs to be explained. The Min_SOL of the liveware node integrates its own capabilities and limitations, and is given by the commander or expert according to risk preferences. The normal operation of the liveware node can be understood that the person is in good working condition and able to make decisions and operate correctly.

F. THE MEASUREMENT OF MISSION SAFETY

For the safety measurement of single-stage mission, two main aspects are considered: mission completion and mission safety. Mission completion degree MC indicates the possibility that the entire mission profile executed safely. It uses the ratio of the time that all basic function nodes are running normally to the duration of the entire mission profile as an indicator.

$$MC = \frac{|M_B|}{|M|} \tag{9}$$

Among them, $|\bullet|$ indicates the number of elements contained in the set $|\bullet|$. M indicates the collection of all meta-missions and M_B indicates the collection of meta-missions with basic function nodes running normally.

Mission safety degree MS indicates the possibility of the successful execution of the mission. It uses the ratio of the time that all mission function nodes are running normally to the duration of mission stage c as an indicator.

$$MS = \frac{|M_{MC}|}{|M_C|} \tag{10}$$

Among them, M_C indicates the collection of meta-missions in mission execution stage c. M_{MC} indicates the collection of meta-missions with mission function nodes running normally in mission execution stage c.

In addition, the structure integrity is defined to determine whether the equipment itself is intact. Structure integrity indicates the integrity of the safety structure layer at the current meta-mission. From the perspective of engineering, if the nodes of the safety structure layer are lower than the requirements of the minimum equipment list, the system has no structure integrity.

$$SI = \begin{cases} \dfrac{|V_{tc}|}{|V|} & |(V - V_{tc}) \cap V_{\min}| = 0 \\ 0 & |(V - V_{tc}) \cap V_{\min}| \neq 0 \end{cases} \tag{11}$$

Among them, V indicates the collection of initial nodes, V_{tc} indicates the collection of nodes that are operating normally at the current meta-mission, and $V_{\min}$ indicates the collection of nodes belongs to the minimum equipment list.

V. CASE STUDY

A typical mission profile of UAV is taken as a case. Its mission profile mainly includes 5 stages: a) take-off and climb,

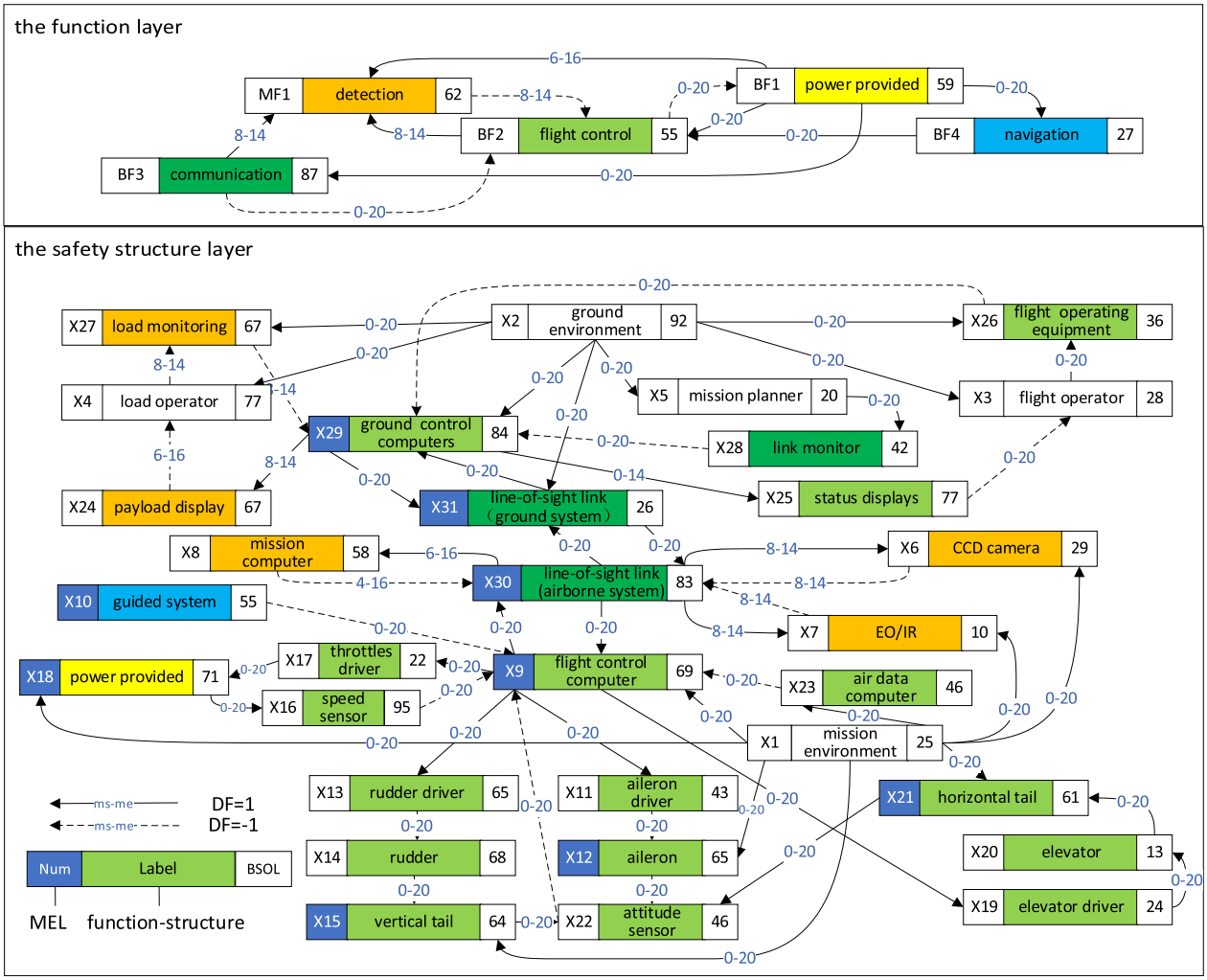


FIGURE 6. The case.

b) cruise to the mission area, c) perform the mission, d) exit the mission area and return, e) descent and landing. The types of missions that may be include daily training, reconnaissance and surveillance, coordinated air combat, ground strikes, decoy deception, electronic jamming, communication relay, etc. The payload that may be involved include communication loads, electronic countermeasures load, weapons and ammunition, etc. The main basic functions include thrust supply, fuel supply, power supply, flight control, take-off and landing, communication and navigation, etc. The mission functions include command and control, perception and detection, weapon fire control, etc.

The case includes 1 mission function node, 4 basic function nodes, 31 safety structure layer nodes, 10 function layer dependencies, and 55 safety structure layer dependencies. The mission is divided into 20 meta-mission in chronological order, among which the m_8 to m_{14} belong to stage c. The detail of case is shown in the figure 6. The process of abstraction can be found in Section II.B, II.C and VI.C. In the process of abstraction, some nodes that express the same function and are similar in space are merged into one node. In addition, since the mission involved in this case are defined as single-stage mission and the logic and timing relations are easily decomposed, the specific dismantling process is not decomposed in detail.

It should be noted that Matlab 2019b is used for simulation, and Origin 2018 is used for drawing, data analysis and fitting in this case.

A. RESULTS AND ANALYSIS OF SINGLE NODE FAILURE

In this section, the value of HI is adjusted to simulate the impact that a single node failure may have on mission safety during mission execution. Under the premise of keeping other parameters unchanged, the test changed the HI of 29 nodes except the environment nodes one by one, and performed the mission profile completely according to the method in section II. HI was set at 0.05 intervals to observe the changes of the indicators: MC, MS, and SI.

As shown in the figure 7, the following conclusions can be drawn: 1) The change of HI will not have a significant

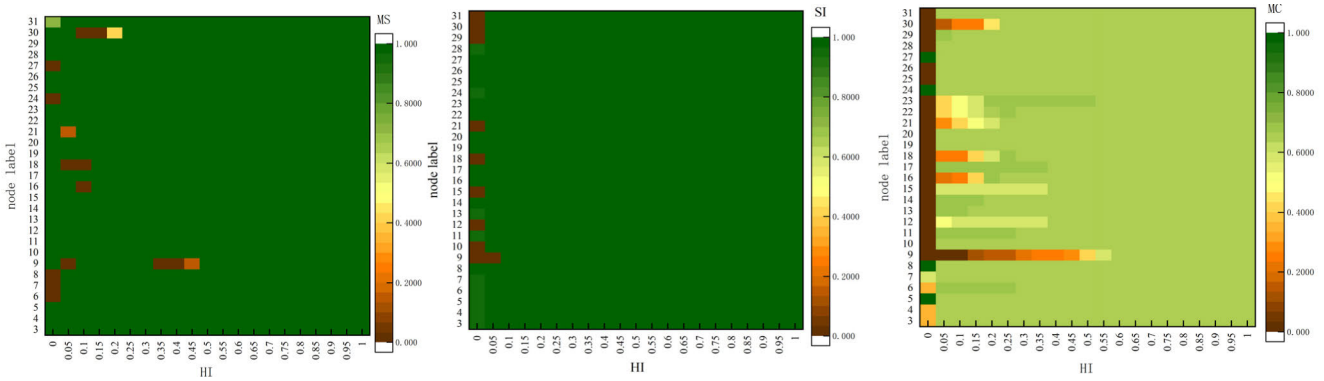


FIGURE 7. The test results of single node failure.

TABLE 2. The label of dependencies.

label	1	**2**	**3**	4	5	**6**	**7**	**8**	9	10	11	12
i	1	**1**	**1**	1	1	**1**	**1**	**1**	2	2	2	2
j	6	**7**	**9**	12	15	**18**	**21**	**23**	3	4	5	26
label	13	14	15	16	17	18	19	20	**21**	22	23	**24**
i	2	2	2	3	4	5	6	7	**8**	9	9	**9**
j	27	29	31	26	27	28	30	30	**30**	11	13	**17**
label	24	25	**26**	**27**	**28**	**29**	30	**31**	**32**	**33**	**34**	**35**
i	9	9	**9**	**10**	**11**	**12**	13	**14**	**15**	**16**	**17**	**18**
j	17	19	**30**	**9**	**12**	**22**	14	**15**	**22**	**9**	**18**	**16**
label	36	**37**	**38**	**39**	**40**	41	42	43	44	45	46	47
i	19	**20**	**21**	**22**	**23**	24	25	26	27	28	29	29
j	20	**21**	**22**	**9**	**9**	4	3	29	29	29	24	25
label	48	49	50	**51**	**52**	53	54	55				
i	29	30	30	**30**	**30**	30	31	31				
j	31	6	7	**8**	**9**	31	29	30				

impact on the MS and SI, but the complete removal of a single node (HI = 0) will produce obvious failure process. 2) The HI change has a more obvious impact on the MC. 3) The safety structure layer nodes corresponding to the basic function nodes fluctuate greatly in the test. Their sensitivity to HI is higher, such as nodes 9, 12, 15, 16, 18, 21, 22, 23, 30, etc. These points are also the key nodes of the system. Among them, nodes 9, 16, 18, 21, and 30 are more important and require special attention.

B. RESULTS AND ANALYSIS OF SINGLE DEPENDENCY FAILURE

In this section, the incomplete expression of the dependency relationship in the safety structure layer by adjusting the MI may have an impact on the mission safety during the mission execution. Under the premise of keeping other parameters unchanged, the test changed the MI of 55 dependencies one by one (the label as shown in the table 2), and executed the mission profile completely according to the method in section II. MI was set at 0.05 intervals to observe the changes of the indicators: MC, MS, and SI.

As shown in the figure 8, the following conclusions can be drawn: 1) Compared with HI, the system is more sensitive to MI changes, especially MC and MS. The change of MI has little effect on SI. 2) The parts in bold in the table 2 are dependencies that are more sensitive to MI changes, and the parts marked in red are the key nodes of the system. It can be seen that the dependencies that can cause significant changes in the system are usually connected to the key nodes, but not all dependencies of key node are sensitive to MI.

C. CASCADE FAILURE TEST RESULTS AND ANALYSIS UNDER DIFFERENT ATTACK STRATEGIES

The components of the analysis framework can be divided into two forms: node and dependency. The failure of dependencies will trigger the update of CSOL. Only when all the dependencies received by the node are destroyed, it will fail/function failure. In other words, there is a certain possibility that the node can operate normally when the dependency failure. However, the failure of a node will cause all dependencies associated with the node to be updated, and attacks on the node will cause more serious cascading failures. Therefore, the test chooses a node-based attack mode.

The cascading failure trigger conditions can be divided into random failures and intentional attack. Random failure refers to randomly reducing the nodes' HI. In engineering applications, random failures mainly come from the reliability of nodes themselves, unintentional human errors and other node failures. The frequency of random failure is high. Its damage degree is random and it is inevitable. However, random failures require a certain amount of accumulation before they may cause great damage to the system. Intentional attack has a clear purpose, and are often a strategic attack on the key components of the system when mastering part or all of the network information. The intentional attack includes deliberate sabotage, large overload maneuver, and mechanism failures. Although intentional attack has a low probability of occurrence, they are likely to cause serious consequences once they occur. And they are often accompanied by random failures. In this case, CSOL, in-SOL, and out-SOL are selected as intentional attack indicators for failure tests. among them,

$$in - SOL_j = \sum_{DS_{ij}=1 HI_i \neq 0 MI_{ij} \neq 0} SOD_{ij}$$
$$out - SOL_i = \sum_{DS_{ij}=1 HI_j \neq 0 MI_{ij} \neq 0} SOD_{ij} \quad (12)$$

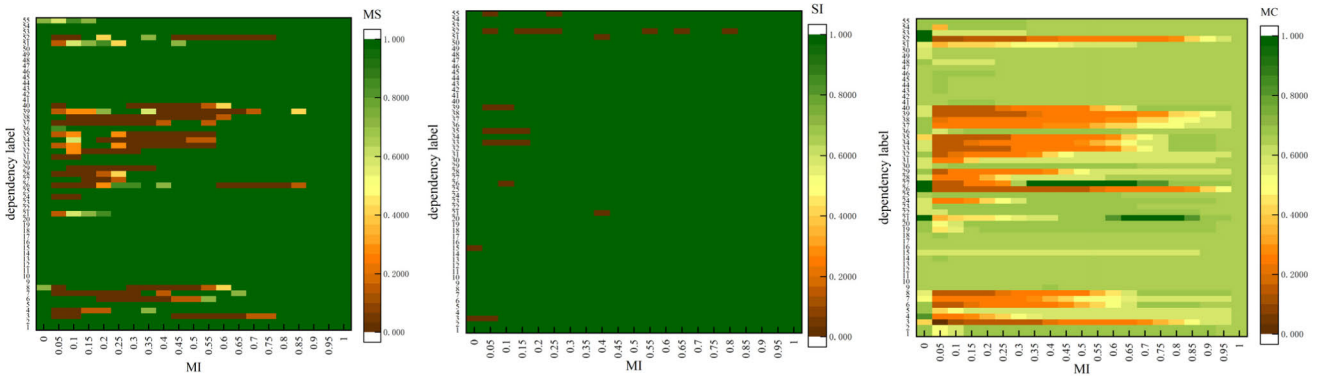


FIGURE 8. The test results of single dependency failure.

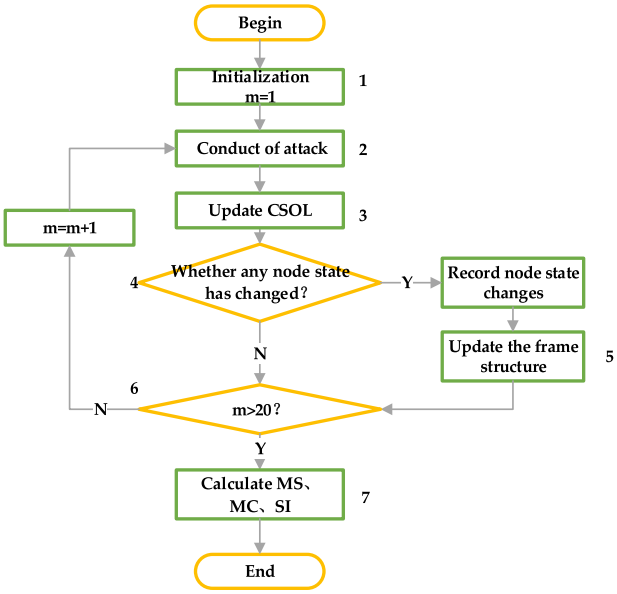


FIGURE 9. The calculation flowcharts.

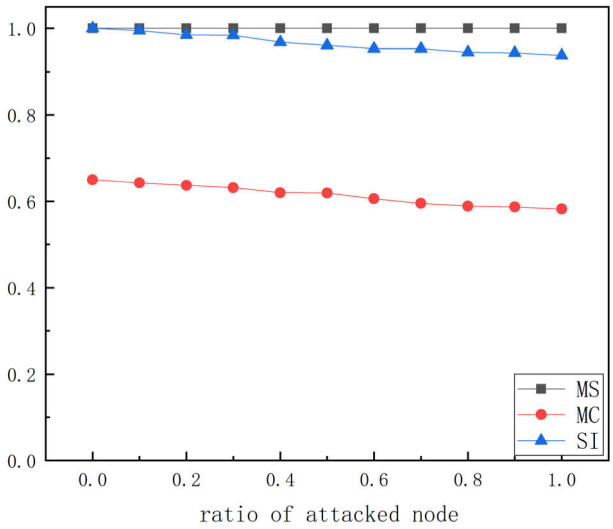


FIGURE 10. The test result of random failure.

The calculation flowcharts are shown in figure 9. It should be noted that (1) Model initialization. This step is mainly to input the basic information of framework and calculate the initial CSOL. It determines the attack strategy used in this round of simulation and the number of nodes that need to be attacked, and randomly select the number of nodes that need to be attacked during the execution of each meta-mission. (2) Attack according to the number of nodes and attack strategy of the current meta-mission. When the intentional attack is adopted, the node with the maximum value of the corresponding indicator at the current meta-mission is selected and fail. When random failure is adopted, the method of randomly generating values is adopted to adjust HI in order to better simulate the failures in reality. (3) Update CSOL according to FDNA. (4) Estimate and record the node state according to the method in section II. (5) Update the frame structure according to the state of nodes. (6) The termination condition of the loop is the complete execution of the mission. (7) Record the relevant indicators of the round.

To avoid contingency, the algorithm runs 1000 times to get the average value, and the node failure rate is set at intervals of 0.1. The result of random failure is shown in the figure 10. The results of three intentional attack strategies (CSOL, in-SOL, and out-SOL) are shown in the figure 11.

From the above figures, the following conclusions can be drawn: 1) The three indicators (MS, MC, SI) all decrease with the increase in the proportion of node failures. Among the three indicators, the MS is generally higher than the other two, MC is the second, and the SI is the lowest. 2) Except for random failures, when the ratio of attacked node is greater than 0.2, the UAV has completely lost the possibility of safe flight. The random failures have no obvious impact on mission safety as a whole. 3) Under intentional attack strategies, MS are quite different, and the other two are basically the same. 4) Under intentional attack strategies, the damage effect according to CSOL is more obvious. 5) It is necessary to comprehensively analyze MS, MC and SI in order to better predict and evaluate the SOL of the UAV in the current mission.

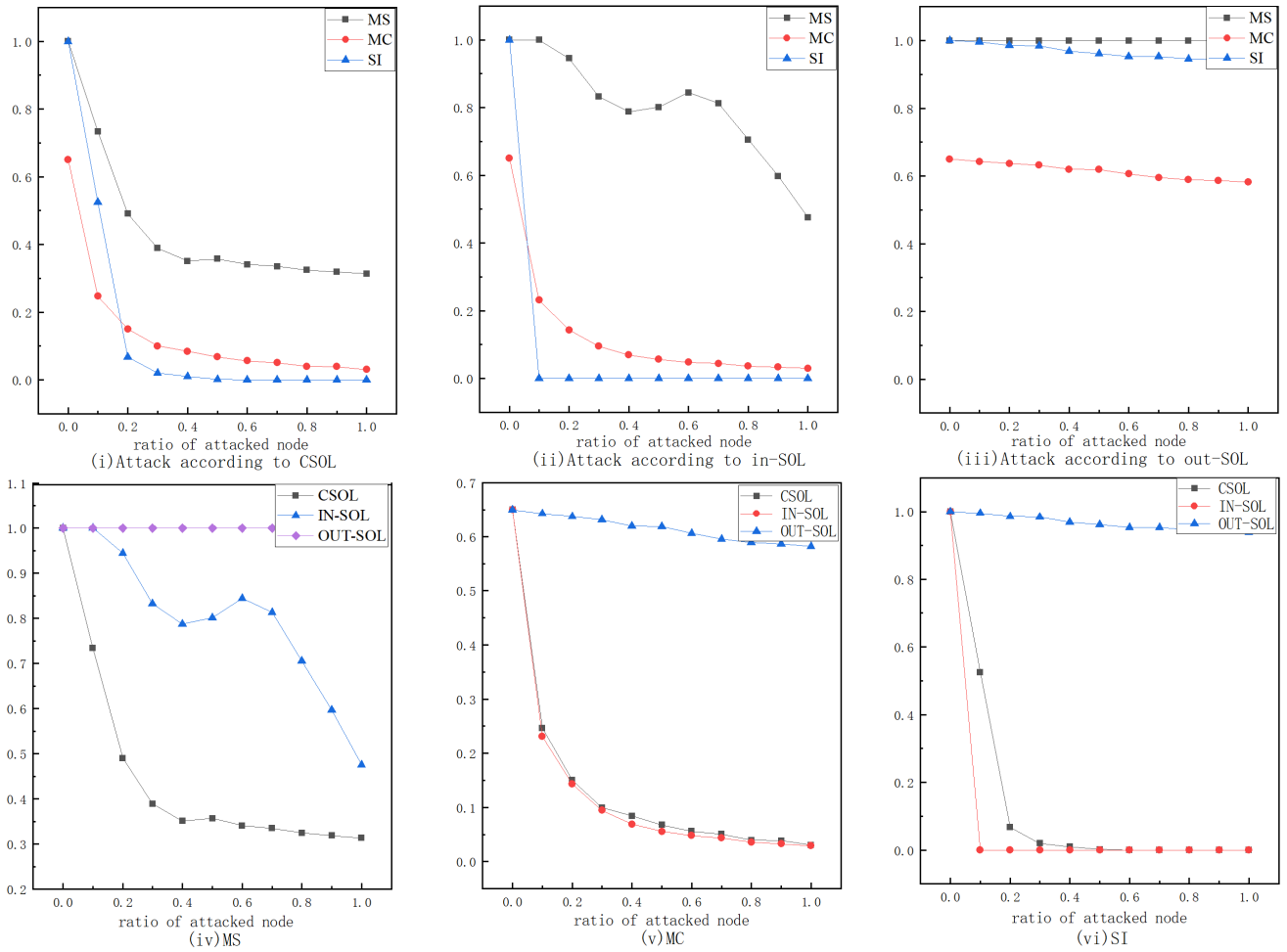


FIGURE 11. The test result of three intentional attack strategies.

TABLE 3. The fitting results of the Weibull distribution.

		y_0		a		r		u		statistics	
		value	std	value	std	value	std	value	std	Reduced Chi-Sqr	Adjusted R squared
CSOL	MS	0.29789	0.0179	0.00126	0.00236	0.19457	0.13469	9.74E-	0.03375	0.04349	0.2571
	M	-	0.0272	93.2577	211.916	0.3663	0.26715	2.83E-	0.04376	0.00861	0.31676
	SI	-	0.0151	373057.	6.88E+0	0.20544	0.37301	0.08976	0.01028	0.02687	0.47301
IN-SOL	MS	0.3336	0.2038	1.41097	1.61456	2.17032	2.51543	-1.089	2.13603	0.04837	0.29166
	M	-	0.0251	111.388	261.752	0.36575	0.25903	4.12E-	0.04224	0.00698	0.34349
	SI	0	0	0	0	1.37E-	0	0	0	0	-3.00E-04
OUT-SOL	MS	0.25582	0.0115	36.4880	0.71527	76.8278	2.85237	-	0.69598	0.02451	0.76362
	M	-	0.0326	68.9398	161.639	0.40482	0.29649	1.07E-	0.05072	0.01014	0.26717
	SI	5.49E-	0.0024	3.23E-	168768.	1.37E-	0.00403	0	2.57E+1	0.00455	-0.00715
Random failure	MS	1	0	0	0	0	0	0	0	0	-3.00E-04
	M	0.35632	0.0154	3.27052	0.78723	1.00838	0.04682	-	0.47644	0.0179	0.02812
	SI	0.91914	0.7407	9.73348	2807.20	6.10788	1612.75	-11.594	2792.56	0.03243	0.01267

In addition, it is found that the cascading failures under the four attack strategies all conform to the Weibull distribution through numerical fitting, as shown in the figure 12. The relevant parameters of the Weibull distribution obtained by the fitting are shown in the table 3 (using the Levenberg-Marquardt optimization algorithm for iteration). The further research is to verify whether the four attack strategies are all conformable to the Weibull distribution through numerical fitting through other fitting methods or mathematical proof.

$$y = y_0 + \frac{r}{a}\left(\frac{x-u}{a}\right)^{r-1} e^{-\left(\frac{x-u}{a}\right)^r} \tag{13}$$

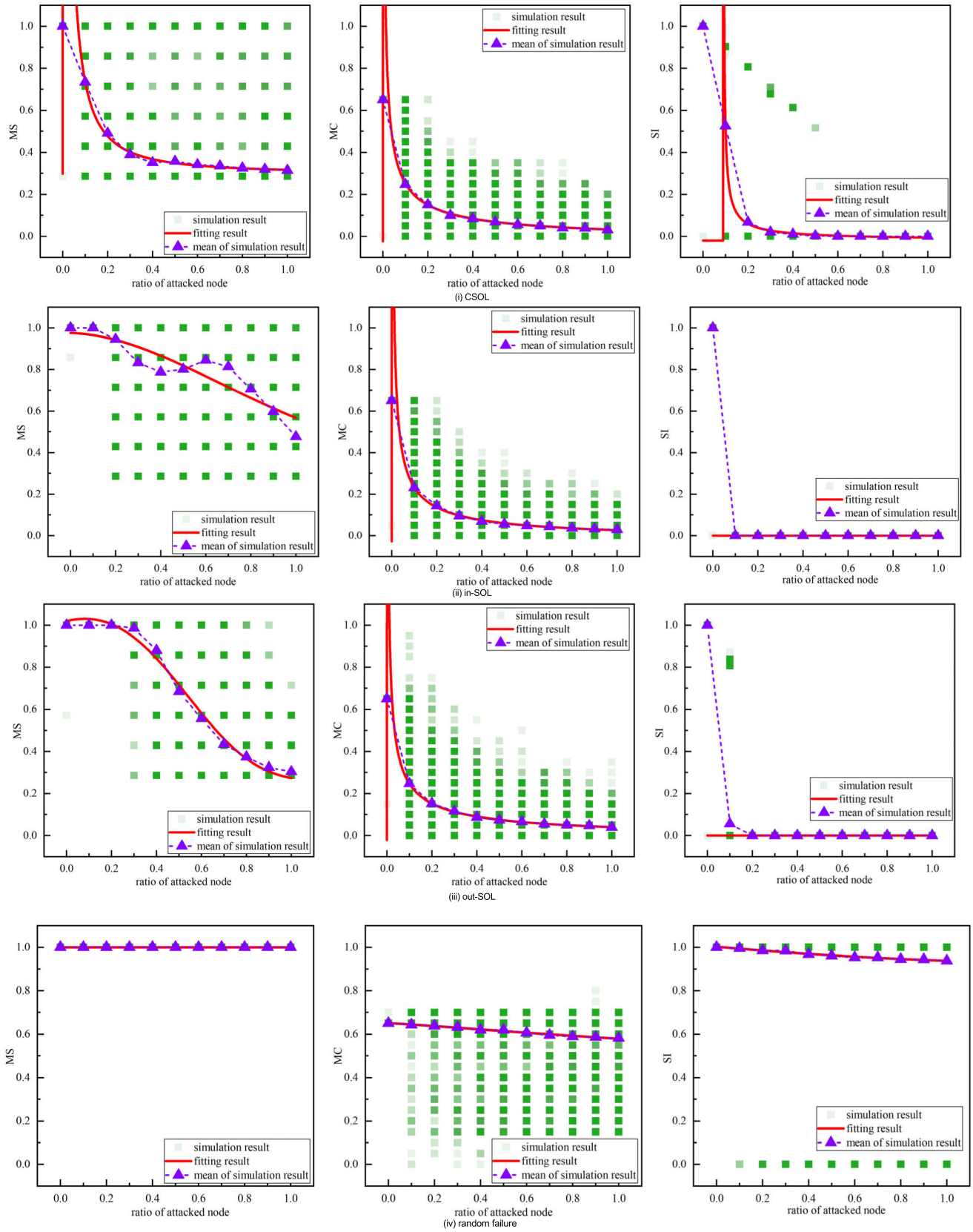


FIGURE 12. The fitting results of the Weibull distribution.

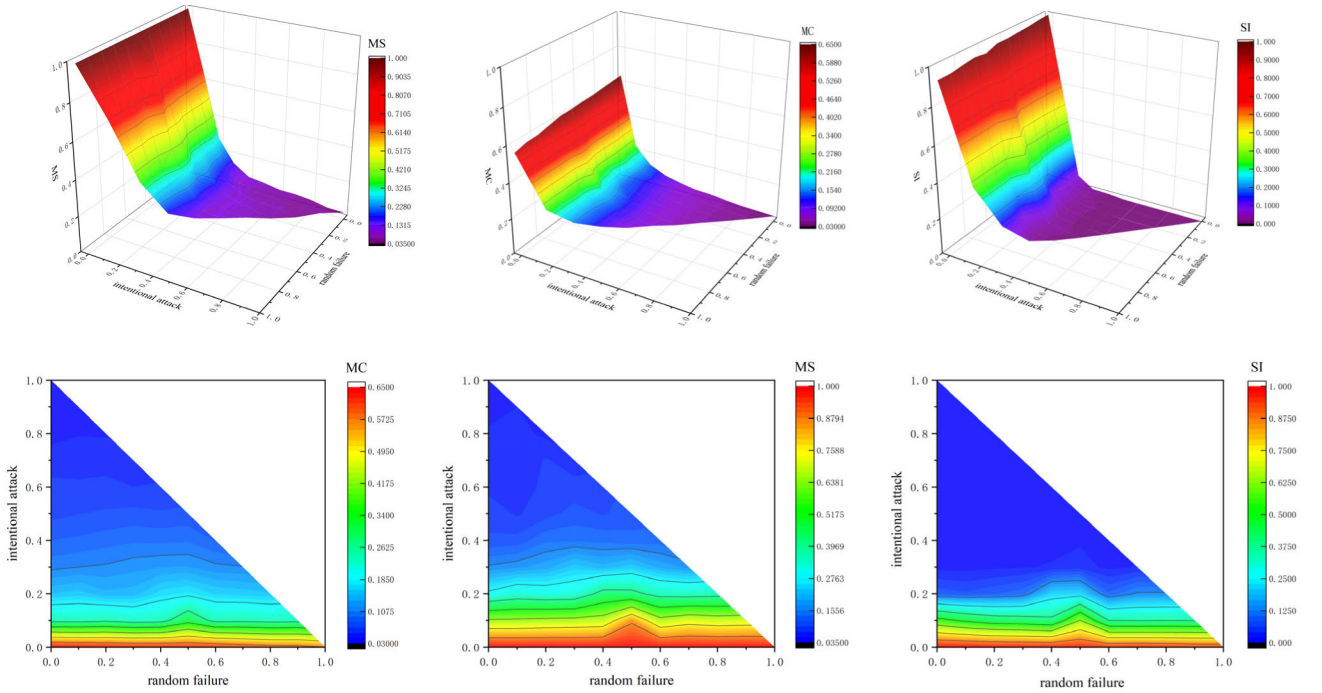


FIGURE 13. The results of cascading failure under combined attack.

D. RESULTS AND ANALYSIS OF CASCADING FAILURE UNDER COMBINED ATTACK

Single node failure, single dependency failure, and single attack mode cannot objectively reflect the mission safety. Therefore, the possible safety results of the mission execution process are simulated through the combined attack (A combination of random failures and intentional attacks). The intentional attack strategies choose CSOL. To avoid contingency, the algorithm runs 1000 times to get the average value, and the node failure rate is set at intervals of 0.1. The calculation result is shown in the figure 13. The simulation process is the same as the figure 9.

The results show that 1) In comparison, the overall MC is low. In the absence of attack, MS and SI are basically maintained at about 1. In other words, there is a great possibility that the UAV will be safely completed in mission phase c, but the possibility that the mission can land safely is not optimistic. 2) With the increase in the proportion of failed nodes, the possibility of safe completion has declined ''cliff-like''. When the node failure ratio is greater than 0.2, the possibility of mission safety execution approaches zero. The effect of random failures on combined attacks is not obvious. 3) The SI is more sensitive to intentional attack and needs special consideration in mission safety assessment.

Based on the above-mentioned tests, the following conclusions can be drawn for the mission safety: 1) The equipment has a high possibility of safely performing a specific mission, but the possibility of safely completing the mission profile is not high. Whether or not to perform this mission requires the commander to make a decision based on its risk preference and comprehensive consideration of mission intentions. 2) Mission safety needs to take into account the completion of the mission, the complete execution of the mission profile and the structural integrity. For the equipment safety assessment of a single mission, it is necessary to consider three aspects of MC, MS and SI. 3) The relationship between the mission safety metrics and the proportion of faulty nodes conforms to the Weibull distribution. 4) The escalating risks will cause serious consequences. When the proportion of failed nodes exceeds 0.2, it can be considered that the mission has failed to complete.

VI. CONCLUSION

This paper first puts forward the concept of mission safety, and constructs a function-structure-based equipment mission safety analysis framework modeling according to ''human-environment-software-hardware''. A numerical simulation method is adopted to analyze the dynamic evolution relationship between nodes and edges. The network damage mode after node failure occurs is obtained under random failures and intentional attacks. Finally, the specific data is used to verify the risk analysis method. The results show that the mission safety assessment needs to consider the completion of the mission, the complete execution of the mission profile and the structural integrity. It is shown that the escalating risks will cause serious consequences. This method can be used to analyze the safety in a specific environment in the design phase and use phase, and has certain reference value for the troops to carry out the risk analysis and research of the equipment system.

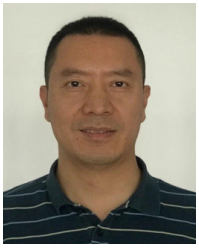
APPENDIX

The authors declare that there is no conflict of interests regarding the publication of this paper.

REFERENCES

[1] S. Wang, "Modeling analysis of complex system accident causation network," *China Saf. Sci. J.*, vol. 23, no. 2, pp. 109–116, Feb. 2013.
[2] Y. Wang, Z. Jin, C. Deng, S. Guo, X. Wang, and X. Wang, "Establishment of safety structure theory," *Saf. Sci.*, vol. 115, pp. 265–277, Jun. 2019.
[3] L. Cui, J. Zhang, B. Ren, and H. Chen, "Research on a new aviation safety index and its solution under uncertainty conditions," *Saf. Sci.*, vol. 107, pp. 55–61, Aug. 2018.
[4] F. Netjasov and M. Janic, "A review of research on risk and safety modelling in civil aviation," *J. Air Transp. Manage.*, vol. 14, no. 4, pp. 213–220, Jul. 2008.
[5] D. R. Insua, C. Alfaro, J. Gomez, P. Hernandez-Coronado, and F. Bernal, "Forecasting and assessing consequences of aviation safety occurrences," *Saf. Sci.*, vol. 111, pp. 243–252, Jan. 2019.
[6] Y. Xue, H. J. Xu, H. Q. Zhu, and J. J. Sheng, "Flight risk probability evaluation in wakes based on multivariate extremum copula," *Acta Aeronautica Astronautica Sinica*, vol. 35, no. 3, pp. 714–726, Mar. 2014.
[7] Y. Xue, H. J. Xu, and M. Q. Hu, "Flight risk probability of pilot-aircraft-environment system under icing conditions," *Acta Aeronautica Astronautica Sinica*, vol. 37, no. 11, pp. 3328–3339, 2016.
[8] J. F. W. Peeters, R. J. I. Basten, and T. Tinga, "Improving failure analysis efficiency by combining FTA and FMEA in a recursive manner," *Rel. Eng. Syst. Saf.*, vol. 172, pp. 36–44, Apr. 2018.
[9] A. Geramian, A. Shahin, B. Minaei, and J. Antony, "Enhanced FMEA: An integrative approach of fuzzy logic-based FMEA and collective process capability analysis," *J. Oper. Res. Soc.*, vol. 71, no. 5, pp. 800–812, May 2020.
[10] F. Franceschini and M. Galetto, "A new approach for evaluation of risk priorities of failure modes in FMEA," *Int. J. Prod. Res.*, vol. 39, no. 13, pp. 2991–3002, Jan. 2001.
[11] Z. H. Qureshi, "A review of accident modeling approaches for complex socio-technical systems," in *Proc. 12th Austral. Workshop Saf.-Rel. Program. Syst.*, 2007, pp. 47–59.
[12] N. G. Leveson, "Applying systems thinking to analyze and learn from events," *Saf. Sci.*, vol. 49, no. 1, pp. 55–64, Jan. 2011.
[13] A. Scarinci, A. Quilici, D. Ribeiro, F. Oliveira, D. Patrick, and N. G. Leveson, "Requirement generation for highly integrated aircraft systems through STPA: An application," *J. Aerosp. Inf. Syst.*, vol. 16, no. 1, pp. 9–21, Jan. 2019.
[14] J. Rasmussen, "Risk management in a dynamic society: A modelling problem," *Saf. Sci.*, vol. 27, nos. 2–3, pp. 183–213, Nov. 1997.
[15] D. S. Castilho, L. M. S. Urbina, and D. de Andrade, "STPA for continuous controls: A flight testing study of aircraft crosswind takeoffs," *Saf. Sci.*, vol. 108, pp. 129–139, Oct. 2018.
[16] W. X. Zhang, "A weapon system of systems safety analysis method based on complex interaction networks," Ph.D. dissertation, Graduate School, Nat. Univ. Defense Technol., Changsha, Hunan, China, 2015.
[17] B. Cai, Y. Liu, Y. Zhang, Q. Fan, Z. Liu, and X. Tian, "A dynamic Bayesian networks modeling of human factors on offshore blowouts," *J. Loss Prevention Process Industries*, vol. 26, no. 4, pp. 639–649, Jul. 2013.
[18] Y. Wang, Z. J. Guo, Y. Sun, and C. Li, "Aircraft safety analysis and simulation based on IDAC-STPA model," *Syst. Eng. Electron.*, vol. 41, no. 5, pp. 125–131, 2019.
[19] S. Kabir, M. Walker, and Y. Papadopoulos, "Dynamic system safety analysis in HiP-HOPS with Petri nets and Bayesian networks," *Saf. Sci.*, vol. 105, pp. 55–70, Jun. 2018.
[20] Q. Zhou, Y. D. Wong, H. S. Loh, and K. F. Yuen, "A fuzzy and Bayesian network CREAM model for human reliability analysis—The case of tanker shipping," *Saf. Sci.*, vol. 105, pp. 149–157, Jun. 2018.
[21] Y. Wang, Y. Sun, X. F. Meng, Y. Qi, and C. Li, "Research on risk transfer GERT of complex equipment systems based on opportunity theory," *Syst. Eng. Electron. Technol.*, vol. 40, no. 12, pp. 92–98, 2018.
[22] Q. Shuang, M. Zhang, and Y. Yuan, "Node vulnerability of water distribution networks under cascading failures," *Rel. Eng. Syst. Saf.*, vol. 124, pp. 132–141, Apr. 2014.
[23] M. Rejzek and C. Hilbes, "Use of STPA as a diverse analysis method for optimization and design verification of digital instrumentation and control systems in nuclear power plants," *Nucl. Eng. Des.*, vol. 331, pp. 125–135, May 2018.
[24] P. R. Garvey and C. A. Pinto, "Introduction to functional dependency network analysis," in *Proc. 2nd Int. Symp. Eng. Syst.*, 2009, pp. 1–17.
[25] L. D. Servi and P. R. Garvey, "Deriving global criticality conditions from local dependencies using functional dependency network analysis (FDNA)," *Syst. Eng.*, vol. 20, no. 4, pp. 297–306, Jul. 2017.
[26] Y. Wang, W. X. Zhang, and Q. Li, "Functional dependency network analysis of security of navigation satellite system," *Appl. Mech. Mater.*, vols. 522–524, pp. 1192–1196, Feb. 2014.
[27] W. Zhang, Z. Li, W. Wang, and Q. Li, "System of systems safety analysis of GNSS based on functional dependency network analysis," *Appl. Math. Inf. Sci.*, vol. 10, no. 6, pp. 2227–2235, Nov. 2016.
[28] Y. Jian, H. Qiwang, and W. Weiping, "Analyzing ballistic missile defense system effectiveness based on functional dependency network analysis," *Open Cybern. Systemics J.*, vol. 9, no. 1, pp. 678–682, Jun. 2015.
[29] B. Drabble, "Information propagation through a dependency network model," in *Proc. Int. Conf. Collaboration Technol. Syst. (CTS)*, May 2012, pp. 266–272.
[30] R. Albert and A. L. Barabasi, "Statistical mechanics of complex networks," *Rev. Modern Phys.*, vol. 27, no. 12, pp. 4622–4623, 2002.
[31] J. X. Gang, H. X. Yuan, and H. Y. Yu, "Capability requirement analysis of aircraft formation based on operational mission," *J. Command Control*, vol. 5, no. 2, pp. 121–127, 2019.
[32] Q. Yu, "Research of complex network risk transmission behavior based on the risk transmission path and node," M.S. thesis, Graduate School, Lanzhou Univ., Lanzhou, Gansu, China, 2017.
[33] C. Guariniello and D. DeLaurentis, "Maintenance and recycling in space: Functional dependency analysis of on-orbit servicing satellites team for modular spacecraft," in *Proc. AIAA SPACE Conf. Expo.*, San Diego, CA, USA, 2013, p. 5327.
[34] C. Guariniello and D. DeLaurentis, "Integrated analysis of functional and developmental interdependencies to quantify and trade-off llities for system-of-systems design, architecture, and evolution," *Procedia Comput. Sci.*, vol. 28, pp. 725–728, 2014.
[35] C. Guarinello and D. Delaurentis, "Dependency analysis of system-of-systems operational and development networks," *Precedia Comput. Sci.*, vol. 16, no. 2, pp. 264–274, 2013.
[36] C. Guariniello and D. DeLaurentis, "Communications, information, and cyber security in systems-of-systems: Assessing the impact of attacks through interdependency analysis," *Procedia Comput. Sci.*, vol. 28, pp. 720–727, Jan. 2014.

YUJIN CHEN received the B.E. degree in computer science from the Dalian University of Technology, in 2015, and the M.E. degree from Air Force Engineering University, in 2017, where he is currently pursuing the Ph.D. degree in management science. His main research interests include rough set and three-way decision making, mission safety, and risk analysis.



JIHUI XU received the B.E., M.Sc., and Ph.D. degrees from Air Force Engineering University, Xi'an, China, in 1996, 2003, and 2011, respectively. He is currently a Professor with Air Force Engineering University. His research interests include uncertainty theory, risk analysis, and aviation safety.

JIAHUI SHI received the B.E. degree from Xi'an Air Force Engineering University, Xi'an, China, in 2019, where he is currently pursuing the M.Sc. degree. His research interests include risk analysis, risk control and aviation safety.

LIJUAN KAN received the B.E. degree in business administration and the M.E. degree in management from the Xi'an University of Technology, in 2005 and 2008, respectively, and the Ph.D. degree in control science and engineering from Air Force Engineering University, in 2019. Her main research interest includes equipment system safety engineering.

WENJIE TIAN received the B.E. degree in safety engineering from Air Force Engineering University, in 2020, where he is currently pursuing the M.E. degree in management science. His main research interests include safety risk assessment and sensitivity analysis.