Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.18576/amis/100625

System of Systems Safety Analysis of GNSS based on Functional Dependency Network Analysis

*Wangxun Zhang**, *Zhifei Li, Weiping Wang and Qun Li*

College of Information System and Management, National University of Defense Technology, Changsha, 410073, China

Received: 2 Mar. 2016, Revised: 25 Jun. 2016, Accepted: 29 Jun. 2016
Published online: 1 Nov. 2016

Abstract: The characteristics of the system of systems (SoS) present great challenges to the safety analysis of Global Navigation Satellite Systems (GNSS). Traditional safety analysis methods and techniques do not work well in a complex SoS, so new safety analysis technologies are needed to adapt to safety problems in SoS. This study first expounds upon the shortcomings of traditional safety analysis methods on GNSS safety and vulnerability study. Then some discussion and works on SoS safety is shown to introduce this new field. In addition, the Functional Dependency Network Analysis (FDNA) method is introduced and an SoS safety modeling and analysis method is proposed, together with detailed processes, which is based on FDNA. Finally, the application of this method is demonstrated through a case study. Based on the case study, it appears that FDNA has great potential and applicability to SoS safety analyses that are otherwise difficult for traditional models or methods to accommodate.

Keywords: Safety Analysis, system of systems, global navigation satellite systems, functional dependency network analysis, modeling

1 Introduction

As the compass of the information age, the Global Navigation Satellite Systems (GNSS) has been increasingly applied to both military and civilian purposes [1], but there are serious inherent shortcomings, which include system-level vulnerabilities, weak signals, etc.[2]. These vulnerabilities may cause not only local or isolated inconveniences for an individual application, but also more serious consequences for safety critical applications. Nowadays, it is widely recognized that the safety and protection issue of GNSS are significant and that there is considerable interest in the vulnerability and safety of GNSS [2,3,4].

To the best of our knowledge, the main research on the vulnerability and safety of GNSS can be classified into three different categories [5,6]: (1) enumeration of the variety of threats to the GNSS and qualitative corresponding defensive measures, (2) quantitative analysis of the effects of those interference methods and anti-jamming technologies from the view of signal or navigation services , and (3) the design of novel anti-jamming or interference detection algorithms.

The above literatures focused on a few specific problems, such as countermoves or anti-interference technologies, rather than the safety problem of the system as a whole. But a GNSS is obviously a kind of SoS according to Geddes's [7] description of System of Systems (SoS):" a system of systems is a collection of interacting systems embedded in a dynamic environment. The behavior of a system of systems is an emergent property of the SoS that results from interactions between the systems within it", because it consists of three interacting segments: the space segment, the ground control segment, and the user segment, and each of the segments also consists of many interacting subsystmes. In addition, for different users of GNSS, the enviroment is different and dynamic. Another equally important reason is that the position, navigation and timing(PNT) service provided by GNSS is emerged by those interacting systems and subsystems. So, it is needed to study GNSS safety from a SoS view, namely the SoS safety view.

Traditional safety and reliability modeling and analysis methods such as Failure Modes and Effects Analysis (FMEA) [8], Fault Tree Analysis (FTA) [9] , Event Tree Analysis (ETA) [10], etc have their own shortcomings and do not work well for complex SoS [11].

There is currently considerable interest in the field of SoS safety and hazards. Leveson [12] states that the changing world and technology make the traditional

* Corresponding author e-mail: zhangwangxun2010@163.com

safety engineering approaches or techniques, which were originally created for first mechanical and then electro-mechanical systems, is no longer applicable to the complex, high-tech systems used today; new accident models and engineering techniques are needed to handle these new complex systems and problems. Bodeau [13] gives an early description of SoS security, recognizing the particularity of SoS security and proposing a security engineering process for SoS that includes legacy systems. Raheja [14], based on his long experience in system safety as a practitioner, trainer and consultant, discusses certain flaws of system safety and proposes new paradigms, the first of which is "system safety must extend to system of systems safety". He argues that "system safety needs to pay more attention to hazard analysis on the structure and architecture of the system-of-systems". Alexander and Kelly [15,16] argue that it is difficult to perform an adequate hazard analysis with traditional hazard analysis techniques, because of the complexity of SoS and the environments that they inhabit. They present a simulation-based hazard analysis method to explore the effects of deviant node behavior within an SoS.

Redmond [17] notes that the emergent properties of SoS can bring new capabilities, but also new hazards. He separates SoS hazards into two distinct categories, hazards from a single system and emergent hazards. The former belongs to the traditional safety domain, while the latter is defined as "any hazard that may occur within a system of systems that is not attributable to a single system". SoS hazard analysis should focus on the latter.

Although the new area of SoS safety has attracted some interest, existing methods only exist for specific systems or domains; there is still no suitable method for all types of SoS. This paper focuses on GNSS safety issues caused by dependency relationships between components. The paper is organized as follows. Section 2 introduces the basic of the Functional Dependency Network Analysis (FDNA) method. Section 3 proposes the FDNA based SoS safety modeling and analysis method together with detailed processes. Section 4 presents a case study, which also serves as an illustration of the method in practice. Finally, Section 5 concludes this paper with a discussion of how well the method meets the SoS safety requirements and it outlines directions for future work.

2 Basic of FDNA

Garvey and Pinto [18,19] originally formulated the Functional Dependency Network Analysis (FDNA) method, which offers the capability to evaluate the effect of both topology and of the possible degraded functioning of one or more systems on the operability of each node in the network [20,21].

According to its definition, there are dependence relationships among the components of a system. This is also the case in a SoS. Therefore, from the view of dependence, components in a dependence relationship can be divided into feeders and receivers. Garvey and Pinto defined two properties of dependency: strength of dependency (*SOD*) between node N_i and node N_j, α_{ij}, and the criticality of dependency (*COD*) between node N_i and node N_j, β_{ij}, to describe each link between a feeder N_i and a receiver N_j. Where *SOD* is a value in the range of 0-1, which captures the effects of relationships that improve baseline operability levels, *COD* is a value in utils (1-100) that capture whether such relationships could involve losses or constraints on these levels.

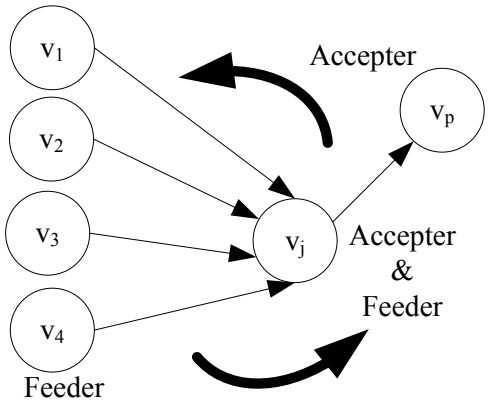

Fig. 1: An FDNA Graph

As shown in Figure 1, the performance of node v_j depends on the performance of feeder

$$P_j = f(\alpha_{ij}, \beta_{ij}, P_i) \tag{1}$$

More generally, for a receiver node v_j that has k feeder nodes $v_1, v_2, \cdots, v_k$, its performance P_j can be expressed as:

$$P_j = F(\alpha_{1j}, \beta_{1j}, P_1, \alpha_{2j}, \beta_{2j}, P_2, \cdots, \alpha_{kj}, \beta_{kj}, P_k) \tag{2}$$

Alothough a new method, After several years of developments, FDNA has been applied to many fields. Drabble applied it to information propagation in the collaboration network [22]. Guariniello and Delaurentis gave some improvements to the method and applied it to maintenance of Aerospace SoS [20], analysis of SoS architecture [21], SoS information and cyber security problems [23].Wang, Zhang and Li applied it to the security analysis of GNSSS [4]. All of these show the great power and potential of FDNA. More over, FDNA can meet the requirements for GNSS safety analysis from the view of SoS well:

1.Nodes and links in FDNA can represent the basic information fo GNSS, including the systems and

interaction and interdependency relationships between them.

2. FDNA uses graphic representation, which makes it very easy to understand and carry out the causal analysis.
3. The calculation method of FDNA allows the representation for combined effects created by multiple failures of any different node and accidents caused by interactions and dependencies among systems that are without faults or errors.

Some of the biggest challenges for SoS safety issues are complex interactions and interdependence. While the FDNA method is proposed to study the potential ripple effects of complex interdependent systems, based on dependency reasoning capability and has above advantages. So it will be suitable for SoS safety study of GNSS.

3 FDNA based Safety Modeling and Analysis

In this section, we propose the SoS safety analysis method based on FDNA, and he SoS safety analysis based on FDNA contains several steps, seen below:

1. Build the basic dependency network model.
   The basic model here contains only basic component systems and basic dependency relations of a SoS, not dependency parameters. The selection of nodes and dependency relations should reflect the demands and focus of the stakeholders, but not so far as to determine every single component.
2. Get dependency parameters.
   For each dependency link in the network, two parameters are needed. The first is strength of dependency (SOD) between node v_i and node v_j, and the second is criticality of dependency (COD) between node v_i and node v_j, which are respectively denoted as α_{ij} and β_{ij}. These parameters can be got by expert opinion, historical data, design documents and so on.
3. Define SoS accidents.
   Abnormity of system states or performances must not be allowed to cause an SoS accident, defined here by the minimum states set $S_i = \{s_{i1}, s_{i2}, \cdots, s_{im}\}$, according to the results of Hazard and Operability Study (HAZOP), FTA, etc. Assuming that there are N accidents in an SoS, which are denoted as $\Omega = \{M_1, M_2, \cdots, M_N\}$, for each accident M_i, a set $S_i = \{s_{i1}, s_{i2}, \cdots, s_{it}, \cdots, s_{im}\}$ is used to describe the minimum state set that will cause the accident. Where, $S_i \neq \emptyset$ and $S_i \subseteq S$, s_{it} is the state of a component system in the SoS, S represents the state set of all the component systems.
   Assuming the state of the SoS at some point in time is $S' = \{s_1, s_2, \cdots, s_n\}$, if this state produces an accident M_i, then $M_i = \{\text{true} | S' \supseteq S_i\}$.
   If an SoS accident occurs, then $\exists S_i$, makes $S' \supseteq S_i$,
   If no SoS accident occurs, then $\forall S_i$ is $(S' \cap S_i) \subset S_i$.
4. Hazard effects analysis.In this step, we will study the effects of performance or state changes of one or more systems in the SoS on other nodes and on the SoS as a whole.
   According to the weakest link principle, Equation (1) can be expressed in Equation (3).

$$P_j = \min(g(\alpha_{ij}, P_i), h(\beta_{ij}, P_i)) \tag{3}$$

where,

$$g(\alpha_{ij}, P_i) = SOD\_P_j = \alpha_{ij} P_i + 100(1 - \alpha_{ij})$$
$$h(\beta_{ij}, P_i) = COD\_P_j = P_i + \beta_{ij}$$

More generally, for a receiver node v_j that has k feeder nodes $v_1, v_2, \cdots, v_k$, Equation (2) can be expressed as following, in Equation (4).

$$0 \leq P_j = \min(SOD\_P_j, COD\_P_j) \leq 100 \tag{4}$$

where,

$$SOD\_P_j = \text{Avg}(SOD\_P_{j1}, SOD\_P_{j2}, \cdots, SOD\_P_{jk})$$
$$SOD\_P_{ji} = \alpha_{ij} P_i + 100(1 - \alpha_{ij})$$
$$COD\_P_j = \min(COD\_P_{j1}, COD\_P_{j2}, \cdots, COD\_P_{jk})$$
$$COD\_P_{ji} = P_i + \beta_{ij}$$

Therefore, according to the above equations, the effects of the performance changes of one or more nodes on their receivers can be analyzed.
Garvey did not consider circularity links in the network, however. For instance, accurately running satellites are dependent on the performance of ground upload antennas, while the performance of an antenna is dependent on the download data history from the satellites. A circularity dependency is, thus, created between satellites and the ground antenna.
The problem of where to start and stop the calculation if there is a circularity link in the network is addressed in the following algorithm.

Algorithm 1 An FDNA Algorithm

```
for each node in FDN do
    rootCause(v)=v;
end for
Denote the start nodes set as SP = {v_i};
while SP ≠ ∅ do
    Denote the receiver set of v_i as FP_i = {v_ik};
    for each FP_i do
        for each v_ik do
            if v_ik ∈ rootCause(v_i) then
                FP_i = FP_i − v_ik;
            end if
        end for
```

if $FP_i \neq \emptyset$ **then**
for each v_{ik} **do**
if *deal*(v_{ik})=false **then**
Calculate the performance of v_{ik} based on dependency relationships, which is denoted as p'_{ik}, where, the performance without dependency relationships is p_{ik};
deal(v_{ik})=true;
end if
if $p'_{ik} \leq p_{ik}$ **then**
rootCause(v_{ik})=*getRoot*;
end if
end for
else if $FP_i = \emptyset$ **then**
$SP = SP - v_i$;
end if
end for
$SP = \bigcup FP_i$;
if $SP = \{MCS\}$ && Valudeschanged=false **then**
STOP.
end if
end while

In this algorithm, we use a node set *SP* to represent the beginning nodes. Here, nodes in *SP* must be of the same kind; namely, the nodes in *SP* are all master control stations, all satellites, all antennas or all monitoring stations. For example, we constrict the master control station (MCS) to be the starting point, then calculate other nodes, step by step, until back to the MCS (termed a calculation cycle), then we compare the values of each node in the network before and after this calculation cycle. If all of the values are the same before and after (*values changed*=false), then the calculation is over; otherwise, we go on to the next calculation cycle.

rootCause(v) represents the root node set that causes performance or state variety of node v. For example, if the performance of v_1 changes, then it will directly impact the performance of v_j, and, indirectly, v_p. Here v_1 is the root cause of v_p. The key here is that *rootCause*(v) is a node set and may contain one or more nodes.

Different feeder nodes may have the same receiver node. For instance, $v_1 \sim v_4$ has the same receiver v_j. *Deal*(v_{ik}) is used to denote whether the node v_{ik} has been calculated by its feeder v_i. Then, when dealing with another feeder of v_{ik}, it does not need to be calculated again.

5.Accident reason investigation.

Step (5) is a bottom-up analysis, which links reason to effect, i.e., investigating the reasons for each accident. Assuming that an abnormal SoS state is $S_N = \{s_1, s_2, \cdots, s_i, \cdots, s_n\}$, the work in this step is to find the root cause for each system state s_i. The process of accident reason investigation is the reverse of effect analysis. From the state changed nodes, according the dependency links, the feeder nodes must be found and the states of these feeder nodes must be analyzed, until the real root cause nodes are discovered.

4 Case Study

This section applies the method and process proposed above to a GNSS safety analysis.

4.1 Background and Assumptions

A team of researchers will complete a mission, which may only last for a month, under the support of GNSS. There are some constrains, however; they can only do the work from nine to twelve o'clock every day. There are six visible satellites during this time, STL1∼STL6, which communicate with the ground upload antennas, ATN1 ATN3, and monitor stations, MNT1∼MNT3. The communication links between these systems are not arbitrary, however, STL1 and STL2 can only communicate with ATN1 and MNT1, STL3 can only communicate with ATN2 and MNT2, and STL4 STL6 can only communicate with ATN3 and MNT3.

The information transmission process occurs when the monitor stations collect information about the satellites and then transmit it to the MCS, where the information is processed in order to determine satellite clock and orbit status. Processed information is then uploaded to the satellites via the antennas. Thus, the whole GNSS network, as described above, is shown in Figure 2, and GNSS user (UR) here is the research team.

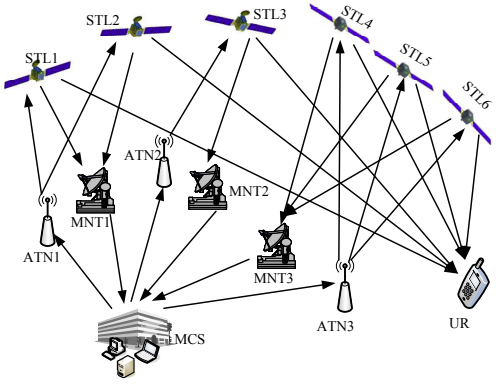

Fig. 2: GNSS network demonstration

4.2 Safety Modeling and Analysis

1.Build the basic dependency network model.

Table 1: Dependency parameters

i/j	1/2	1/3	1/4	2/5	2/6	3/7	4/8	4/9	4/10	5/11	6/11	7/12
α_{ij}	1.00	1.00	1.00	0.70	0.80	0.75	0.85	0.80	0.85	0.35	0.25	0.80
β_{ij}	0	0	0	20	20	30	10	15	20	45	55	15
i/j	8/13	9/13	10/13	11/1	12/1	13/1	5/14	6/14	7/14	8/14	9/14	10/14
α_{ij}	0.35	0.40	0.25	0.60	0.30	0.65	0.80	0.75	0.85	0.30	0.25	0.20
β_{ij}	60	50	70	20	30	45	15	20	15	65	75	85

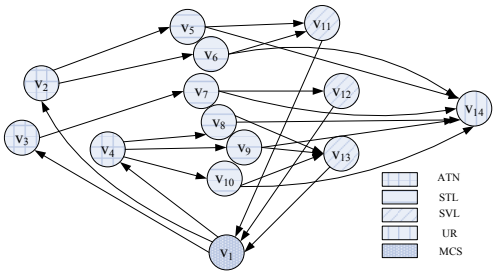

Fig. 3: Basic dependency network model

The basic dependency network contains nodes and dependency links. The nodes here are the six satellites STL1∼STL6, the three monitor stations MNT1 MNT3, the three upload antennas ATN1∼ATN3, an MCS, and a user. The dependency links are communication links between these nodes. The basic dependency network model is shown in Figure 3.

2.Get dependency parameters.
Based on specialist experience, historical data, design documents and so on, the dependency parameters for the GNSS dependency network are shown in Table 1.

3.Define SoS accidents.
Six accidents for the GNSS are defined in Table 2.

Table 2: GNSS accidents list

Number	Definition
I	$P_{14} < 90$
II	More than 3 satellites are seriously degraded
III	MNT2 v_{12} is seriously degraded
IV	ATN2 v_3 is seriously degraded
V	MNT1 v_{11} and MNT3 v_{13} are seriously degraded
VI	ATN1 v_2 and ATN3 v_4 are seriously degraded

The definition for a seriously degraded satellite is:

$$\text{State}_{STL} = \{\text{serious}|P_{STL} \leq 90\} \quad (5)$$

The definition for a seriously degraded monitor station or upload antenna is:

$$textState_{\text{Ground}} = \{\text{serious}|P_{\text{Ground}} \leq 80\} \quad (6)$$

where, P_{STL} and P_{Ground} are the performance values of satellites and ground stations, which contain the monitor station and upload antenna.

4.Hazard effects analysis.
In this step, we will analyze the effects of different hazards (abnormal states of systems), and judge whether the hazards caused the SoS accidents defined in Table 2. Two kinds of analyses (deterministic analysis and stochastic analysis) can be conducted here.

(a)Assuming that a single node of the network is degraded by 20 for some reason, while other nodes are all work normally, then its effects on the user node v_{14} and SoS accidents in the GNSS network are shown in Table 3.

Table 3: Effects of $P_4 = 20$ on the network

	V14	**I**	**II**	**III**	**IV**	**V**	**VI**
v1=20	67.70	Y	Y	Y	Y	Y	Y
v2=20	55.00	Y					
v3=20	55.00	Y	Y	Y	Y	Y	
v4=20	89.89	Y	Y				
v5=20	35.00	Y	Y				
v6=20	40.00	Y					
v7=20	35.00	Y	Y	Y	Y	Y	Y
v8=20	85.00	Y					
v9=20	93.98						
v10=20	96.43						
v11=20	75.77	Y	Y	Y	Y	Y	Y
v12=20	79.81	Y	Y	Y	Y	Y	Y
v13=20	85.87	Y	Y	Y	Y		Y
v14=20	20.00	Y					

In Table 3 “Y” means that the accident happened, and blank means that the accident did not happen. It is clear that the GNSS network has different sensitiveness to the degradation of each node: single degradation of node v_1, v_7, v_{11} and v_{12} caused all the accidents defined in Table 2. And degradation of v_5, v_6 and v_7 affect the user node v_{14} most. While degradation of v_9 and v_{10} has hardly any effects on the whole GNSS network because they caused no accidents and affected scarcely any on the user node. Therefore, such network could be preferable if node v_9 and v_{10} are prone to be attacked or failures. And v_7 may be

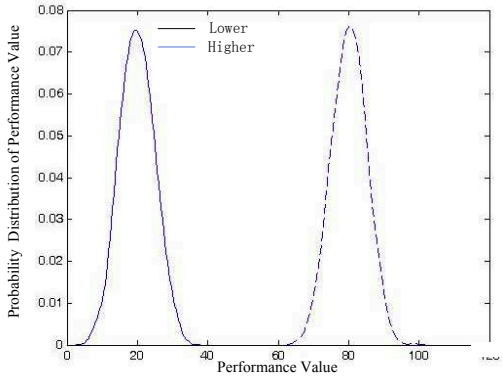

Fig. 4: Shift of probability distribution of v_4 from lower to higher performance

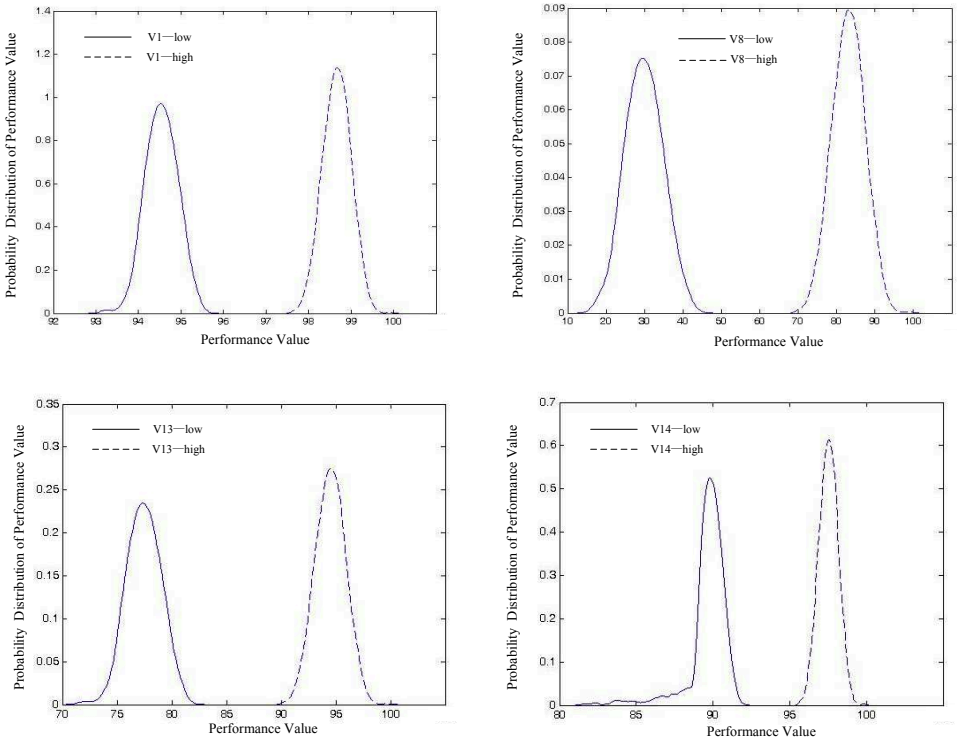

Fig. 5: Effects of v_4 performance change on nodes v_1, v_8, v_{13} and v_{14}

the most vulnerable node because its degradation caused all the accidents and affected the user node most.

(b)The stochastic analysis can capture more information and resilience or sensitivity of the network. Two pairs of simulations are designed here, one team runs 1,000 times when v_4 operates at a lower performance level ($\mu = 20$, $\sigma = 5$), while the other runs 1,000 times when v_4 is at a higher performance level ($\mu = 80$, $\sigma = 5$). After each pair of simulations, the probability distribution curve of each node in the network can be drawn according to the simulation results. Figure 4 shows the shift of the probability distribution curve of v_4, and Figure 5 shows the shift of v_8, v_{13}, v_{14} and v_1.

This simulation also counts the numbers of each type of SoS accident (Table 4) that have occurred, the results of which show that when v_4 is at lower performance level ($\mu = 20$, $\sigma = 5$), the statistical

number of accident I is 563, and II is 1,000, while that of other accidents are all 0. When v_4 is at a higher performance level ($\mu = 80$, $\sigma = 5$), accident II happened 929 times.

Table 4: Effects of v_4 performance change on SoS accidents

Type	I	II	III	IV	V	VI
Lower v4	563	1000	0	0	0	0
Higher v4	0	929	0	0	0	0

Figure 5 shows that the direct receiver node v_8 (v_9 and v_{10} are nearly the same with v_8) is the most sensitive node to a variety of v_4 values. Its mean performance value changes from about 29.9 to 83.1 when that of v_4 changes from 20 to 80. The more sensitive nodes are v_{13} (from 77.5 to 94.5), v_{14} (from 89.7 to 94.5). Node v_1 is robust (from 94.5 to 98.7 and the variances are very small) to the changes in v_4 because it is far from v_4.

In addition to the promulgation of single system failures in the network, emergencies of combined failures or degenerations are also possible. A single failure has been studied above, but, in the following, a combined problem will be analyzed assuming that there is some degeneration of satellite v_7 ($\mu = 30$, $\sigma = 5$), together with a fault in v_4 ($\mu = 20$, $\sigma = 5$). Also, the simulation is run 1,000 times, and we then study the combined effects on the GNSS network. The frequency of each type of SoS accident is shown below in Table 5.

Table 5: SoS accident times under combined degeneration of v_4 and v_7

Type	I	II	III	IV	V	VI
f	1000	1000	1000	919	1000	919

It is apparent in the table above that there are more SoS accidents under combined faults or degeneration compared with simple single failures, encompassing all of the SoS accidents. Figure 6 compares mean performance values of each nodes between only v_4 degeneration and two nodes degeneration.

It can be seen that there is nearly no difference between single degeneration and combined degeneration for v_8, v_{10} and v_{13}; great difference for v_{12} and v_{14}; obvious difference for other nodes. There are two possible reasons for this. First of all, the dependency relation parameters SOD and COD could be involved. Secondly the distance from the root cause node to the effect node can also play a part. v_8, v_{10} and v_{13} have little

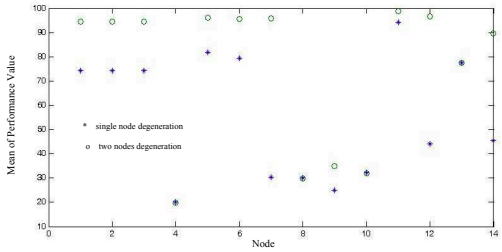

Fig. 6: Comparison of only v_4 degeneration to v_4 and v_7 combined degeneration

dependency on v_7, so there are scarcely any differences for their probability distribution curves in Figures 6 and 7. v_{12} and v_{14} is directly dependent on v_7, and they both have a strong dependency ($\alpha_{7,12} = 0.80$, $\alpha_{7,14} = 0.85$) on the root cause node v_7.

5.Accident reason investigation.

Take accident IV (No.5 simulation) as an example, assuming that the reason for the accident is not clear. The investigation process is shown below.

The definition of accident IV is "ATN2 v_3 is seriously degraded". Thus, the investigation will start at v_3. According to the basic node information that $I_3 = \{v_1\}$, together with the GNSS network in Figure 4 and dependency parameters in Table 1, we know that the performance of v_3 wholly depends on v_1 ($\alpha_{1,3} = 1$), so we go to node v_1.

Results show that the performance value of v_1 is 72.08. There is no failure or fault in v_1 itself, however, so the reduced performance value may be caused by its feeder nodes $I_1 = \{v_{11}, v_{12}, v_{13}\}$. Therefore, we then go to v_{11}, v_{12} and v_{13}.

The final performance values of v_{11}, v_{12}, and v_{13} are 93.79, 42.08, and 79.86, respectively. These three nodes, themselves, are also without fault or failure, so we go on.

For node v_{12}, after 1 step of this converse reasoning, we get to the root cause node v_7. After 2 steps, the root cause node of v_{13} is also found, v_4. But after 3 steps for v_{11}, we go back to node v_1, which means that we enter a cycle. We can, of course find the curve and correct the root cause node(s) with detailed calculation and careful judgment, but it is difficult to do solely by hand and takes time. Fortunately, we have considered the cycle in a network and designed an algorithm that can be automatically run by computer beforehand. The algorithm can automatically compute the hazard effects, but also find accident reasons.

Thusthe root nodes for accident IV of the No. 5 simulation are found, which contains v_4 and v_7. The manual investigation process is shown below in Figure 7.

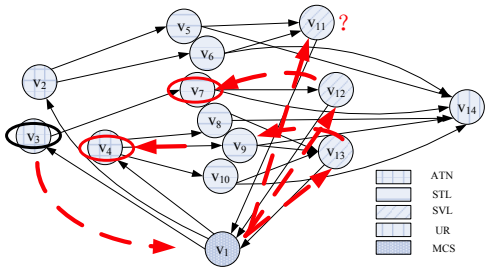

Fig. 7: Accident reason investigation process

After finding the accident reasons or root cause nodes, we can strive to improve the SoS safety through enhancing the protection ability of those nodes, or adjusting the dependency relationships of correlative links by optimizing the SoS architecture.

5 Conclusions and future work

A main problem causing vulnerability of GNSS that is facing systems engineering and management professionals, and the key challenge for this issue, is that GNSS is a type of SoS. Although still a relatively new area of research, SoS safety has attracted considerable interest.

This paper first discussed the limitations of research on GNSS safety, and briefly introduced some work on SoS safety.

Then, to address the questions and requirements for a SoS safety method, this paper presented the FDNA-based method for SoS safety analysis of GNSS, together with detail modeling and analysis processes. Alos, an algorithm that can be run automatically by computer was designed.

In the case study, a GNSS dependency network model was built and simulation results showed that the method is well suited for GNSS safety analysis. The method can (1) describe the basic information of GNSS, (2) model the accidents caused by interactions among component systems, (3) analyze hazard effects, (4) investigate the cause of accidents, and (5) and be executed by a computer.

The advancement of the model, however, is still under continuous development. Further work is indicated, which must include two aspects:

1. **Expand and refine the node attributes.** In this paper, only one comprehensive performance value is considered without other parameters. However, the states or behaviors of a node need more parameters for a general description.
2. **Define other relationships between nodes.** Dependency relationships only exist in a single direction in FDNA, but in an actual system or SoS, other relationships must be accounted for, such as bidirectional interactions between two components.

Acknowledgement

This research was supported by the Natural Science Foundation of China (61074107, 91024015).

References

[1] Kaplan, E.D. and C.J. Hegarty, Understanding GPS: principles and applications. 2006, Massachusetts: Artech House Inc Press.
[2] Thomas, M., et al., Global navigation space systems: reliance and vulnerabilities. 2011, The Royal Academy of Engineering: London.
[3] Zhang, W. and H. Hou, Study on Safety & Protection Ability of GNSS Receiver from the View of Main Materiel System. Applied Mechanics and Materials, 2014. **511-512**(2014): p. 1048-1052.
[4] Wang, Y., W. Zhang, and Q. Li, Functional Dependency Network Analysis of Security of Navigation Satellite System. Applied Mechanics and Materials, 2014. **522-524**(2014): p. 1192-1196.
[5] Wangxun, Z., H. Hongtao, and W. Weiping, Research on GNSS's security-protection. Computer Engineering & Science, 2013. **35**(4): p. 60-64.
[6] Wangxun, Z., H. Hongtao, and W. Weiping, MATE based design for protection of GNSS. Systems Engineering and Electronics, 2013. **35**(6): p. 1231-1235.
[7] Geddes, N.D., D.M. Smith, and C.S. Lizza. Fostering collaboration in systems of systems. in Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on. 1998. IEEE.
[8] Stamatis, D.H., Failure mode and effect analysis: FMEA from theory to execution. 2003: Asq Press.
[9] Ericson, C.A. and C. Ll. Fault tree analysis. in System Safety Conference, Orlando, Florida. 1999.
[10] Ericson, C.A., Event Tree Analysis. Hazard Analysis Techniques for System Safety, 2005: p. 223-234.
[11] Leveson, N.G., Engineering a safer world: Systems thinking applied to safety. 2011: MIT Press.
[12] Leveson, N.G., A new accident model for engineering safer systems. Safety Science, 2004. **42**(4): p. 237-270.
[13] Bodeau, D.J. System-of-systems security engineering. in Computer Security Applications Conference, 1994. Proceedings., 10th Annual. 1994. IEEE.
[14] Raheja, D. and B. Moriarty, New paradigms in system safety. Journal of System Safety, 2006. **42**(6).
[15] Alexander, R.D., Using simulation for systems of systems hazard analysis. 2007, University of York.
[16] Alexander, R. and T. Kelly. Hazard Analysis through Simulation for Systems of Systems. in Proceedings of the 24th International Systems Safety Conference. 2006.
[17] Redmond, P., A system of systems interface hazard analysis technique. 2007, Naval Postgraduate School: Monterey, CA.

[18] Garvey, P.R. and C.A. Pinto. Introduction to Functional Dependency Network Analysis. in Second International Symposium on Engineering Systems, Massachusetts Institute of Technology. 2009. Cambridge, Massachusetts.
[19] Garvey, P.R. and C.A. Pinto, Advanced Risk Analysis in Engineering Enterprise Systems. 2012, Boca Raton,FL: CRC Press.
[20] Guariniello, C. and D.A. DeLaurentis. Maintenance and Recycling in Space: Functional Dependency Analysis of On-Orbit Servicing Satellites Team for Modular Spacecraft. in AIAA SPACE 2013 Conference and Exposition. 2013. San Diego, CA.
[21] Guariniello, C. and D. DeLaurentis, Dependency Analysis of System-of-Systems Operational and Development Networks. Procedia Computer Science, 2013. **16**: p. 265-274.
[22] Drabble, B. Information propagation through a dependency network model. in Collaboration Technologies and Systems (CTS), 2012 International Conference on. 2012. Denver,CO: IEEE.
[23] Guariniello, C. and D. DeLaurentis, Communications, information, and cyber security in Systems-of-Systems: Assessing the impact of attacks through interdependency analysis. Procedia Computer Science, 2014. **28**(2014): p. 720-727.

Wangxun Zhang is currently working towards the Ph.D. degree in College of Information System and Management in National University of Defense Technology. His current research interests include satellite navigation system simulation and system of systems safety.

Zhifei Li is pursuing a Ph.D. at the College of Information Systems and Management Science at NUDT. His main areas of research include complex systems analysis, agent-based modeling, and simulation.

Weiping Wang received a Ph.D. in systems engineering from the National University of Defense Technology, Changsha, China. His research area covers system of systems engineering, system modeling and simulation.

Qun Li is Professorof National University of Defense Technology. He received his Ph.D.degree in control science and engineering from National University of Defense Technology in 1999.His current research interests include ?exible simulation theory, simulation for operational effectiveness evaluation and system of systems engineering.