

# PROMOTING SYSTEM SAFETY AND RELIABILITY THROUGH RISK QUANTIFICATION/VISUALISATION METHODOLOGY

TAKAFUMI NAKAMURA  
Daito Bunka University, Japan

## ABSTRACT

In this paper, a meta-methodology for holistically examining system failures is proposed to prevent their further occurrence. This methodology was introduced as a meta-methodology called system of system failures (SOSF). SOSF is represented in a three-dimensional space. In addition, a topological method used to monitor failure events within an SOSF space was presented to visualise the trajectory of system failures. A method was developed for quantifying the risk factors for a system failure that enables the factors to be quantified, monitored, and compared among the systems, and whose usage promotes system safety and reliability. The method was introduced using an interaction and coupling (IC) chart based on normal accident theory. An IC chart is used to classify object systems based on an interaction (linear or complex) and coupling (tight or loose); however, its effectiveness is limited by a subjective classification. The proposed method quantitatively (i.e. objectively) measures the risk factors and thus compensates for the subjectivity of an IC chart. Application examples in information and communication technology (ICT) engineering demonstrate that the proposed method applied to quantitatively monitor the risk factors helps improve the safety and quality of various object systems.

*Keywords: risk management, system failure model, normal accident theory, interaction and coupling chart, information and communication technology.*

## 1 INTRODUCTION

In this study, a new methodology is proposed to enhance the safety and reliability of a system by visualising its risk over time. The methodology used for understanding the current state of a failure has numerous shortcomings [1]. The first is the lack of a common language for discussing a failure, making it difficult for stakeholders involved in a failure to have a common understanding of the problem. Consequently, similar failures can occur repeatedly. A meta-methodology called system of system failures (SOSF) was introduced to address this first issue. Second, it is difficult to understand how the nature of a failure changes over time, rather than understanding each failure separately. For the second issue, SOSF is recognised as a failure space, and a topology is introduced into the space allowing individual failures to be recognised with relevance rather than an individual understanding. In this study, the first and second shortcomings were interlinked and organised to improve the safety and reliability of the system. Finally, the new methodology was applied to ICT failures, and its effectiveness was confirmed. ICT was chosen because its environmental changes are severe. These changes were identified in Gartner's study on IT trends [2]. There are three major concerns related to ICT risk: virtualisation, fabric technology, and big data. These three concerns will remain the same for 2022. An applied example shows that the risk quantification/visualisation methodology helps improve the safety and reliability of ICT systems by capturing the evolution and increasing the complexity of ICT technology with virtualisation technology. This application opens up the possibility of an expansion to other social systems.

This study is composed of five sections. Section 2 describes a survey of current methodologies. Section 3 introduces the new SOSF meta methodology, which overcomes the shortcomings of current approaches. Section 4 introduces new metrics into the SOSF space for a topological representation of system failure risk factors. Section 5 provides an



application of ICT system failures. Finally, Section 6, concludes this paper by describing the effectiveness of this methodology and areas of future research.

## 2 SURVEY OF CURRENT METHODOLOGIES

### 2.1 Features of existing structuring methodologies and risk analysis techniques

Two methodologies are widely used, i.e. failure mode effect analysis (FMEA) [3] and fault tree analysis (FTA) [4]. FMEA reveals in a table form the linear relationship between the causes and consequences that lead to the final event in a bottom-up manner. By contrast, FTA is a method for clarifying the causes of a final event in a top-down manner as a logic diagram.

Both methodologies are primarily employed in the design phase. However, they are heavily dependent on personal experience and knowledge. FTA, in particular, tends to miss some failure modes among failure mode combinations, particularly emergent failures. There are two main reasons for missing failure modes. First, FMEA and FTA are rarely applied simultaneously. Accordingly, the sufficiency of the identified elements is not ensured in a mutually exclusive or collectively exhaustive manner. As the second reason, current approaches use a linear link between cause and effect, making it difficult to see complex issues involving many different stakeholders. Other major risk analysis techniques (including FMEA and FTA) have been described in various studies [5]–[7]. Most failure analyses and studies are based on either FMEA or FTA.

The existing structuring methodologies and risk analysis techniques do not sufficiently utilise a holistic approach. Many current methodologies connect cause and effect in a linear manner, and cannot respond appropriately to problems surrounded by many different stakeholders or problems under severe external changes in the environment. Therefore, what is lacking in the current majority of methodologies is the capability to tackle emergent problems caused by the complex relationships between the many stakeholders as well as external environmental changes surrounding the problem. A typical methodology applying a holistic approach is a soft systems methodology (SSM) [8]. SSM can manage emergent properties and thus implement preventative measures. Unlike the other current methodologies, SSM solves the above problem by revealing the nature of the stakeholders surrounding the problem among the customers, actors, transformers, owners, and worldview. Current methodologies tend to lose their holistic view of the root causes of a system failure. In addition, although most of them may be able to clarify the problem structures to confirm the effectiveness of a preventative measure, they do not properly monitor the system failure trends over time. Therefore, systems often exhibit similar failures.

### 2.2 Issues and challenges of current troubleshooting methodologies

All engineering systems were designed to achieve their goals. Events that fail to achieve their goals (i.e. system failures) in such a system can be attributed to an insufficient design. As Turner and Pidgeon pointed out, a system failure may be defined as a characteristic of subsystems that do not contribute to the fulfilment goal of the supersystem. Alternatively, a system failure is the “termination of the ability of an item to perform its required function” [9]. The predominant technology in current IT troubleshooting is a predefined goal-oriented model. In this model, Van Gigch [10] highlighted the main shortcomings of a system improvement as follows:

- Engineers tend to try to find malfunctions inside the system boundary.



- Engineers tend to focus on returning the system to normal. Long-term improvements cannot be achieved through operational improvements.
- Engineers tend to have incorrect and obsolete assumptions and goals. In most organisations, the formulation of assumptions and goals is not explicit. In this context, improvements to fostering systems are senseless.
- Engineers tend to act as “planner followers” rather than as “planner leaders”. In a system design concept, the planner must be a leader planning to influence trends, rather than a follower planning to satisfy trends.

Explanations of system failures in terms of a reductionist approach (i.e. an event chain of actions and errors) are not useful for improving the system designs [11], [12]. In addition, Perrow [13] argued that a conventional engineering approach to ensure safety by building more warnings and safeguards fails because the system complexity makes failures inevitable. The following four key features have commonly been pointed out as limitations of current troubleshooting methodologies in IT system environments.

- (1) Most methodologies have a reductionist perspective. This makes it difficult to understand the real meaning of the countermeasures, whether they are effective or tentative.
- (2) The current mainstream troubleshooting approach applies a cause–effect analysis (or event chain analysis) to determine the real root causes. FMEA or event trees utilise forward sequences, and FTA or fault trees utilise backward sequences [3], [4].
- (3) The speed of intense technological advances creates critical misunderstandings among stakeholders. Current methodologies cannot properly manage the disjunction among stakeholders.
- (4) An improvement of the deviation from the operating norm is bound to fail, and as Van Gigch [10] pointed out, the treatment of a system problem is bound to fail when improving the operation of existing systems.

To summarise these four points, the current methodology focuses on the following three issues:

- The system does not meet the pre-defined goals.
- The system produces no predicted results.
- The system does not work as expected at the design phase.

As the basic assumption of improvement, the goal and operating norm are static and predetermined in the design phase and are based on hard-systems thinking.

These four key features and three issues hinder the examination of system failures from a holistic standpoint, making it difficult to manage the soft, systemic, emergent, and dynamic aspects of such failures.

Based on the discussion described in this section, there are four major shortcomings of the current methodologies.

1. A lack of a methodology covering the world views of multiple stakeholders (Section 2.2, shortcoming (3)).
2. A lack of a methodology covering emergent failures (Sections 2.1 and 2.2, shortcomings (2) and (4)).
3. A lack of a methodology covering a holistic view of system failures rather than a reductionistic view (Section 2.2, shortcoming (1)).
4. A lack of a methodology to monitor system failure trends over time (Section 2.1).



In this study, a new meta-methodology (Section 3) called SOSF is proposed as a countermeasure to the first, second, and third shortcomings. In Section 4, a risk quantification/visualisation method is introduced to address the fourth shortcoming.

3 PROPOSAL OF NEW META METHODOLOGY

SOSF promotes double-loop learning to overcome the first, second, and third shortcomings mentioned in the previous section. Double-loop learning is indispensable for reflecting whether operating norms (i.e. mental models) are effective [14]–[16]. A meta-methodology was used to transform the mental models [10]–[13], as described in Section 2.2. The system of system methodologies (SOSM) proposed by Jackson [17] is a typical and widely recognised meta-methodology. Its main features are the use of a meta-systemic approach (soft system thinking to promote double-loop learning) and complementarianism by encompassing multiple paradigms depending on the state of the problem. FTA [2], FMEA [1], and other analysis types, as discussed in Section 2.2, belong to the simple unitary domain in SOSM.

To overcome the first shortcoming (i.e. covering multiple worldviews of stakeholders), SOSF is designed on the SOSM base that covers multiple stakeholders (i.e. the plural domain in SOSM). In particular, SOSF was developed by placing each type of failure from the system failure taxonomy [18] onto a two-dimensional SOSF (left side of Fig. 1). Notably, the recursive and hierarchical features of SOSF depend on the viewpoint of the system. These features form a system as a structural aggregation of subsystems, where each subsystem has its own SOSF. It is therefore important to note the hierarchical and relative structures (i.e. a technology failure may be an evolutionary failure depending on the viewpoint of the subsystem).

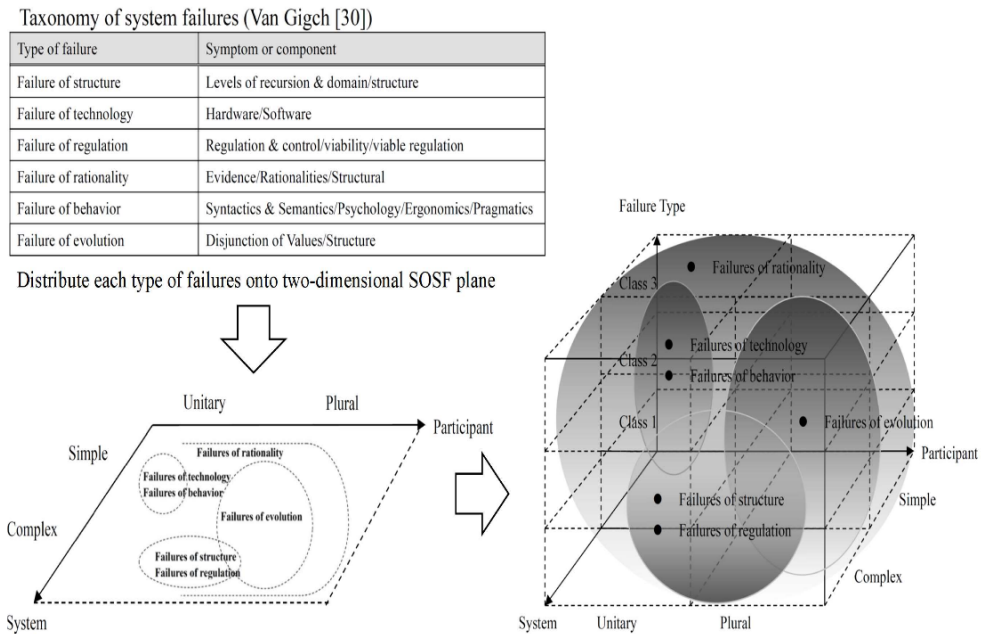


Figure 1: SOSF formulation process.

To overcome the second and third shortcomings (i.e. to cover emergent failures and a holistic view), we introduced a third dimension, namely the failure class. Three failure classes were defined to address the emergent and holistic aspects of a system failure. As Nakamura and Kijima [19] pointed out, failures are classified according to the following criteria.

- Class 1 (deviance failure): The root causes are within the system boundary, and conventional troubleshooting techniques are applicable and effective.
- Class 2 (interface failure): The root causes are outside the system boundary but are predictable at the design phase.
- Class 3 (foresight failure): The root causes are outside the system boundary and are unpredictable during the design phase.

The right-hand side of Fig. 1 (i.e. three-dimensional SOSF) is an expansion of the two-dimensional SOSF (left-hand side of Fig. 1) with the addition of the system failure dimension (i.e. three failure classes).

#### 4 RISK QUANTIFICATION/VISUALISATION METHODOLOGY

The IC chart and closed-code metrics are introduced in this section, including how topological metrics are introduced in the SOSF space to quantitatively monitor the system risk over time.

##### 4.1 Normal accident theory and IC chart

The normal accident theory divides the analysed system into two axes (the coupling axis, i.e. the degree to which the components combine, and the interaction axis, i.e. the degree to which the analysed system interacts with the external environment). The plane represented by these two axes is called an IC chart. An IC chart was first proposed by Perrow [13], and various social systems have been qualitatively laid onto the IC chart plane, as shown in Fig. 2. The IC chart allows a qualitative analysis of the system as two independent variables. Several failures occur, both of which may have a low impact. However, the complex and unexpected interactions of these low-impact failures become fatal owing to a cascading series of minor failures before the safety equipment or alarm comes into effect, which is known as a normal accident against a single-point failure.

To introduce a metric into the IC chart, this qualitative argument should be confirmed using this quantitative measure. As Perrow [13] points out, there is currently no reliable way to measure these two interaction and coupling variables. Therefore, a quantitative discussion is crucial. By introducing this metric into the IC chart, a numerical discussion (change in the position of the failure and time series) can be applied using the IC chart. To address this quantitative issue, the next section explains the method for introducing a metric into the IC chart. A closed-code system of an object system failure is introduced as a metric. If an appropriate topology is added to the IC chart, then a quantified visualisation is possible, enabling us to understand the direction of change in the target system, and thus apply effective countermeasures. The use of this metric enables the safety and quality of a target system to be monitored quantitatively over time.

##### 4.2 Close-code metrics as an example taxonomy of system failures

To introduce a metric into the IC chart described in Section 4.1, we focused on a closed-code system. A close-code system is a type of table that classifies the causes of failures in each

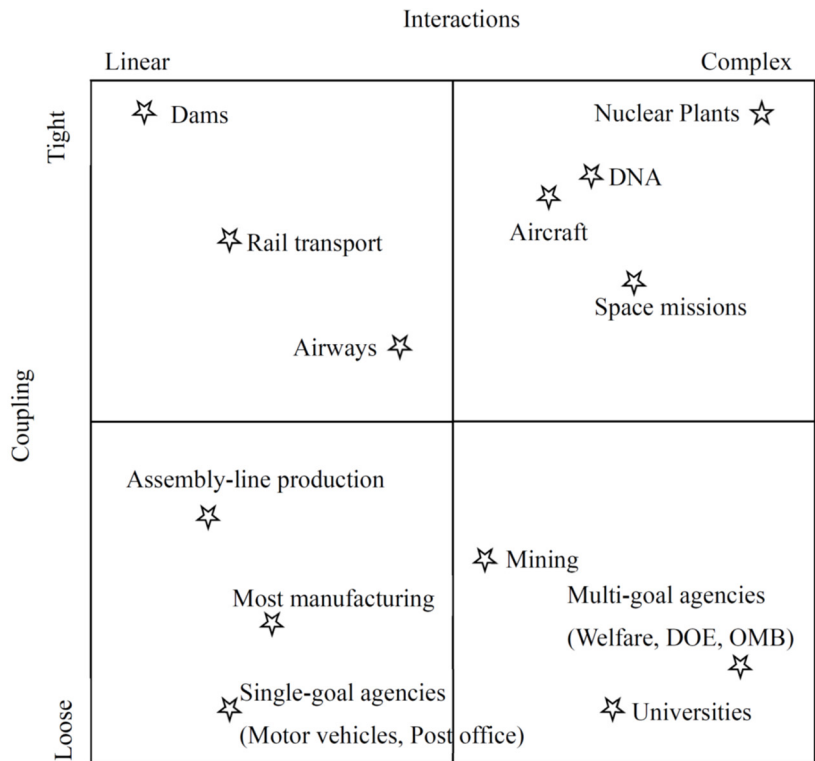


Figure 2: IC chart [13].



industry sector. A close-code is a failure root case classification taxonomy (e.g. hardware, software, or human error) used in a system.

Although the closed-code system varies by system and industry, it is classified as a closed-code matrix with two dimensions. The first dimension consists of phases for creating an object system (i.e. design, configuration, and operation in a time sequence), and the second dimension is the nature of the stakeholders (i.e. simple or complex) responsible for the system failures. A closed-code system is a filter for the root cause of a system failure, and is an example taxonomy of system failures [10], [18]. If a taxonomy of system failures is regarded as a generalised closed-code system, a different closed-code system can be standardised for each industry. This idea enables a quantitative discussion of the nature and changes in the time-series of failures within an industry, enabling comparisons between industries.

Table 1 summarizes an example of mapping a close-code system onto a close-code matrix for the ICT industry, and the relationship between a close-code matrix and an IC chart is clarified. The horizontal (vertical) axis of the close-code matrix corresponds to the interaction (coupling) axis of the IC chart. The coordinates of the close-code matrix in Table 1 are represented by (m, n), where m(n) is the number of horizontal row (vertical column) (i.e. (m = 1: design; m = 2: configuration; m = 3: operation) and (n = 1: simple; n = 2: complex)). For example, (2, 2) indicates configuration-complex area. This (m, n) notation enables risk factors to be quantified. The next section introduces the metric derived from a closed-code matrix used in an SOSF space.



Table 1: Example close-code matrixes in ICT industry.

	Close codes	1 (Design)	2 (Configuration)	3 (Operation)
		Failure of technology and structure	Failure of regulation	Failure of behaviour and evolution
		Failure of rationality, evolution		
1 (Simple)	A (Hardware)	A (1–5) <sup>*1</sup>		A(B) <sup>*2</sup>
	B (Behaviours)		B (A–D, F) <sup>*6</sup>	B(E) <sup>*6</sup>
				
	P (Obsolete)		P	
2 (Complex)	A (Hardware)	A (6) <sup>*3</sup>		A(U) <sup>*4</sup>
	B (Behaviours)			B(G) <sup>*6</sup>
				
	N (Future plan)	N		
Legend: (Causes A and B have subcategories) <sup>*1</sup> A(1)–A(5): CPU, memory, channel, power, and disk failure, respectively. <sup>*2</sup> A(B): hardware setup mistake. <sup>*3</sup> A(6): other IO. <sup>*4</sup> A(U): unknown causes. <sup>*5</sup> B(A)–B(G): network setup, IO setup, parameter setup, installation, operation, application coding mistake, and other mistakes, respectively.				

#### 4.3 Introduction of metric into SOSF space

In Section 3, SOSF was introduced, and in the previous section, a metric capable of a quantitative discussion when using a closed code system was described. Here, the metric detailed in the previous section is introduced into SOSF. Consequently, SOSF is transformed into a topological space, enabling a quantitative discussion. Each failure in an SOSF space is represented by a point representing the system risk location (SRL). The risk of the target system is represented by a three-tuple. This can be expressed as an SRL within an SOSF space. An SRL is represented by three-dimensional coordinates (X, Y, Z) in an SOSF space, where X (Y, Z) represents the system interaction (system coupling and annual call rate (ACR)). ACR is the ratio of incidents per unit of shipment each year. Isomorphism occurs among the two-dimensional SOSF (left side of Fig. 1), closed-code matrix, and IC chart. The isomorphs of these three perspectives with the component attributes are shown in Fig. 3.

There are four steps used to introduce these metrics into an SOSF space, as shown in Fig. 4. The first step (Fig. 4(a)) defines the system failure group at any arbitrary time.

Fig. 4(b) shows that  $\beta$  is the area inside  $\alpha$ ; therefore,  $\beta$  is obtained by dividing the number of system failures in the (3, 2) area by the total number of system failures. The X–Y axes in Fig. 4(b) correspond to the interaction-coupling axes. The quantification of risk factors is achieved using the (m, n) notation in the close-code matrix.

We define  $\gamma$  as shown in Fig. 4(c). The complex and loose risk factors of an object system are represented by  $\gamma = (\alpha, \beta)$ , which is the quantitative coordinate point in the IC chart.

Fig. 4(d) provides a detailed explanation of  $\gamma = (\alpha, \beta)$  in an IC chart for the system failure group at any arbitrary time. Adding a new dimension (i.e. the Z-axis representing ACR) to  $\gamma$  produces the SRL ( $\alpha, \beta, \text{ACR}$ ). Here, ACR is the frequency of system failures and should therefore be incorporated into the system failure metric.

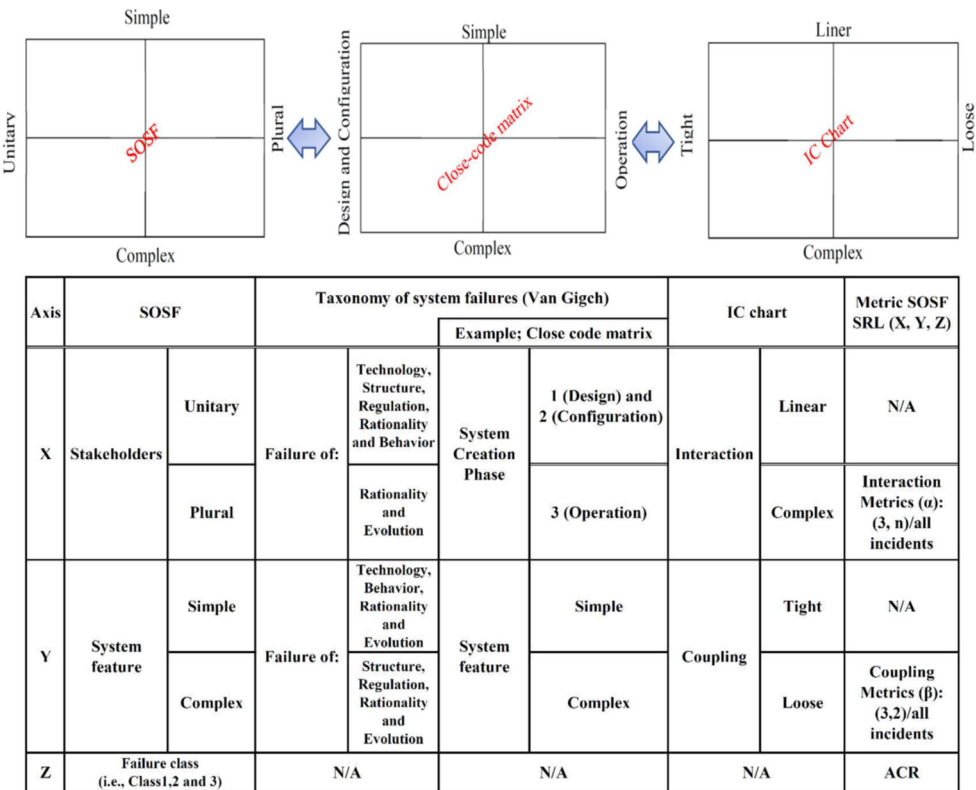


Figure 3: Isomorphic structure of three perspectives and its component attributes.

In Sections 3 and 4, a topological SOSF was introduced, and a system failure can now be discussed numerically. To confirm its effectiveness in the next section, a new methodology is applied to the ICT system shown in Fig. 5. The virtualised ICT systems in Fig. 5 are mainly composed of three technologies (operating systems, networks, and virtualisation platform products). Virtualised ICT systems and other IOTs (i.e. internet-based information architectural devices) are also complexly connected throughout the networks.

## 5 APPLICATION

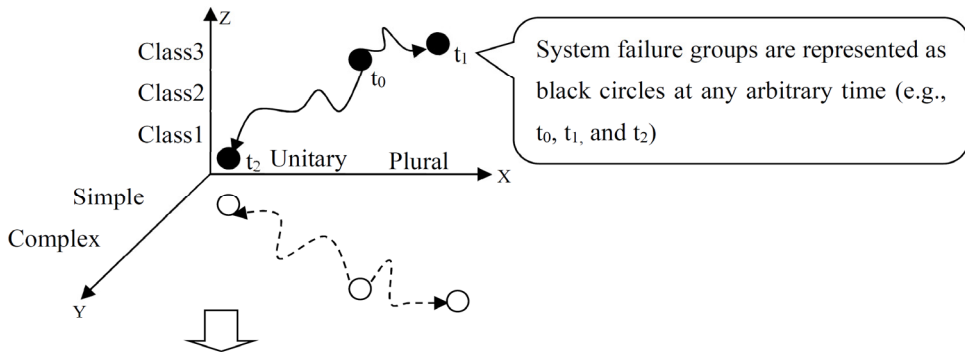
As described in the Introduction, ICT technologies are drastically changing. Complexly connected ICT systems are considered suitable for verifying the effectiveness of the proposed methodology. An overview of the application target system is shown in Fig. 5.

The SRL transition for virtualised ICT systems over a 3 year period is shown in Table 2, and SRL ( $\alpha$ ,  $\beta$ ) was calculated based on the incidents that occurred in the corresponding system components (i.e. OS, virtual platform, and network). Every system failure was identified and gathered from incidents reported by the field operation group and analysed quantitatively by an ICT company. A closed-code matrix (Table 1) was used to formulate a metric within the SOSF space. According to Table 2, the network and virtualisation platforms move towards a complex-loose direction with an increase in the ACR. The OS moves towards a linear-tight direction with a decrease in the ACR.

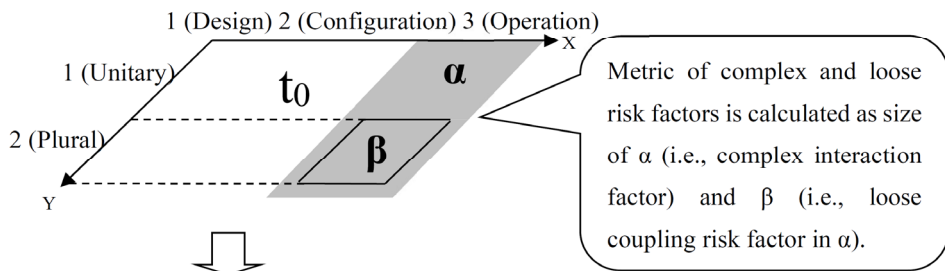




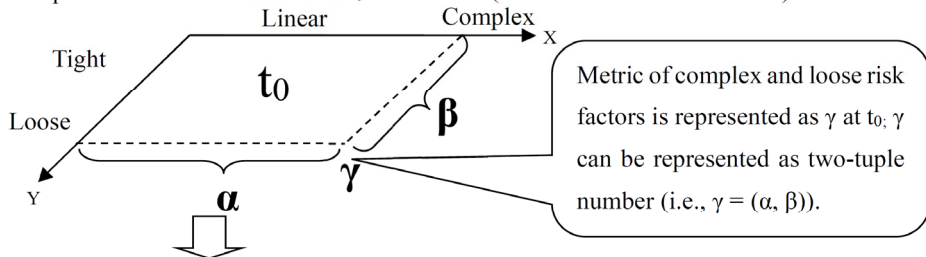
(a) System failure group at  $t_0$  (Cf. Table 3 SOSF column)



(b) Complex and loose risk factors at  $t_0$  in close code matrix (Cf. Table 3 Close code matrix column)



(c) Complex and loose risk factors at  $t_0$  on IC chart (Cf. Table 3 IC chart column)



(d) System risk location at  $t_0$ : SRL ( $\alpha$ ,  $\beta$ , ACR) (Cf. Table 3 Metric SOSF SRL column)

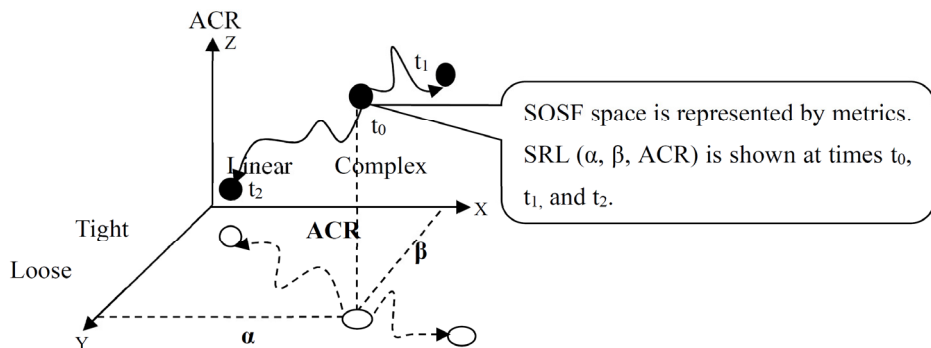


Figure 4: Detailed diagram of metric generation.

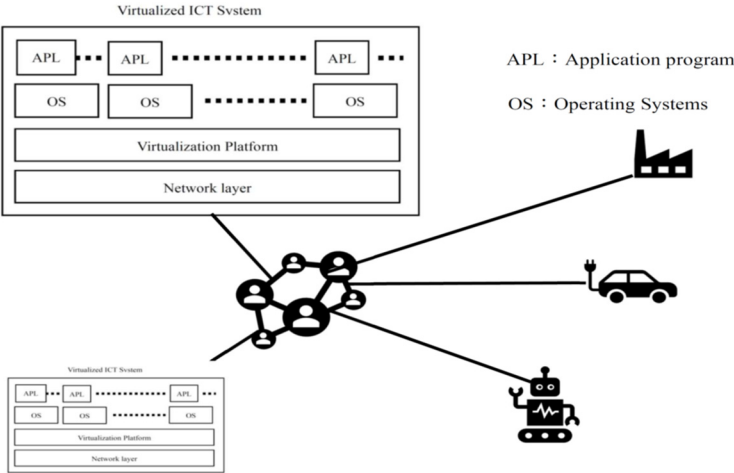


Figure 5: Overview of complexly connected ICT systems.

Table 2: SRL transition.

	2016			2017			2018		
	$\alpha$	$\beta$	ACR	$\alpha$	$\beta$	ACR	$\alpha$	$\beta$	ACR
OS	19.3	19.1	15.5	19.5	19.2	20.6	18.1	18.0	17.6
Virtual	16.7	16.4	31.2	18.0	16.8	39.6	20.3	19.2	41.7
Network	12.6	8.9	44.6	13.0	8.7	57.2	15.9	12.1	52.0

The SRL of the network and virtualisation platform migrates towards the complex-loose direction, and the OS migrates towards the linear-tight direction. The SRL trajectory for each product is shown in Fig. 6. A brief explanation of the key application results is presented below. The network and virtualisation platform migration trends are attributed to changes in the external environment, whereas the OS migration trend is attributed to improved quality and reliability.

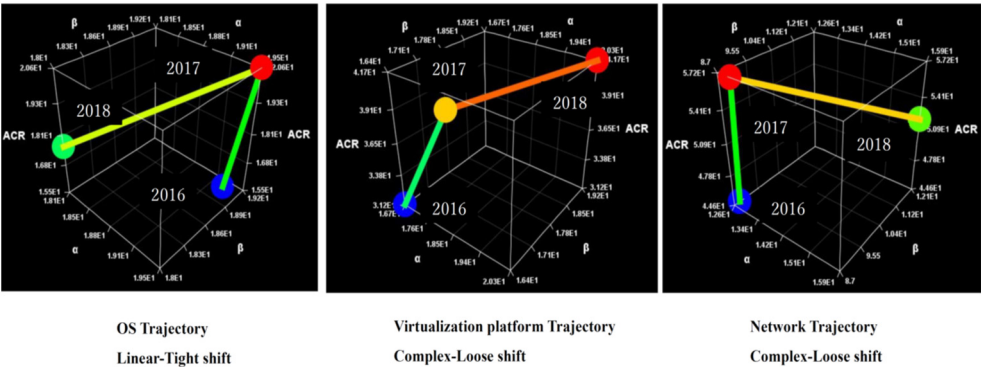


Figure 6: SRL trajectory of each product.

## 6 CONCLUSION

In this paper, a risk quantification/visualisation methodology for an SOSF space was proposed and applied to complexly connected ICT systems. As an application, the risk over time can be clearly visualised using the quantified SRL within the SOSF space.

The findings of this application are as follows:

1. The SRL of the network and virtualisation platforms has been shifting towards a complex-loose domain from a linear-tight domain. In 2018, the ACR increased from its value in 2017.
2. The SRL of the OS platform shifted towards a linear-tight domain from a complex-loose domain. The ACR in 2018 decreased from its value in 2017.

The first result is believed to be due to the diversity of the stakeholders and the complexity of the technology in relation to networks and virtual platforms, particularly complex shifts that deteriorate the risk over time. This change supports Gartner's analysis, as described in the introduction. This loose shift is believed to be due to system redundancy (such as a network or server duplex).

The second result is believed to be a continuous improvement in quality from a relatively small number of OS vendors in comparison to that of network and virtualisation development vendors, and the speed of change in OS technologies is relatively slow compared to that of networking and virtualisation.

For greater safety and higher reliability, the application results suggest the following. If complex shifts are inevitable owing to technological changes, measures such as the introduction of redundancy enhancement equipment contribute towards a movement away from a catastrophic outcome. In other words, measures such as avoiding complex shifts can enhance the reliability and safety of ICT systems. The results of this application in ICT systems suggest that it may be effective for other social systems as well.

Through its application in an ICT system, the proposed methodology shows the effectiveness of quantitatively monitoring the level of risk over time.

Further research is required to expand this approach to other industries with various close-code matrices. This method will lead to a refinement of the proposed methodology and thus contribute to enhancing the safety and security of our society as a whole.

## REFERENCES

- [1] Nakamura, T. & Kijima, K., System of system failure: Meta methodology to prevent system failures. *System of Systems*, ed. A.V. Gheorghe, IntechOpen: London, pp. 31–56, 2012.
- [2] Cooney, M., Gartner: 10 key IT trends for 2012. <https://www.networkworld.com/article/2220899/gartner--10-key-it-trends-for-2012.html>. Accessed on: 17 Mar. 2022.
- [3] IEC 60812:2018, Procedure for failure mode and effect analysis (FMEA and FMECA). <https://webstore.iec.ch/publication/26359>. Accessed on: 17 Mar. 2022.
- [4] IEC 61025:2006, Fault tree analysis (FTA). <https://webstore.iec.ch/publication/4311>. Accessed on: 17 Mar. 2022.
- [5] Bell, T.E., Special report: Managing Murphy's law: Engineering a minimum-risk system. *IEEE Spectrum*, pp. 24–57, 1989.
- [6] Wang, J.X. & Roush, M.L., *What Every Engineer Should Know About Risk Engineering and Management*, Marcel Dekker Inc., 2000.
- [7] Beroggi, G.E.G. & Wallace, W.A., Operational risk management: A new paradigm for decision making. *IEEE Transactions on Systems, Man and Cybernetics*, **24**(10), pp. 1450–1457, 1994.



- [8] Checkland, P. & Holwell, S., *Information, Systems and Information Systems: Making Sense of the Field*, John Wiley, 1998.
- [9] Turner, B.A. & Pidgeon, N.F., *Man-Made Disasters*, 2nd ed., Butterworth-Heinemann: UK, 1997.
- [10] Van Gigch, J.P., *System Design Modeling and Metamodeling*, Plenum: New York, 1991.
- [11] Rasmussen, J., Risk management in a dynamic society: A modeling problem. *Safety Science*, **27**(2/3), pp. 183–213, 1997.
- [12] Leveson, N., A new accident model for engineering safer systems. *Safety Science*, **42**(4), pp. 237–270, 2004.
- [13] Perrow, C., *Normal Accidents Living with High-Risk Technologies*, Princeton Paperbacks: New York, 1999.
- [14] Morgan, G., *Images of Organization*, New edition, SAGE: California, 1997.
- [15] Argyris, C. & Schoen, D., *Organizational Learning II*, Addison Wesley: Massachusetts, 1996.
- [16] Senge, P., *The Fifth Discipline: The Art and Practice of the Learning Organization*, 1st ed., Doubleday: New York, 1990.
- [17] Jackson, M., *System Thinking: Creative Holism for Managers*, John Wiley, 2003.
- [18] Van Gigch, J.P., Modeling, metamodeling, and taxonomy of system failures. *IEEE Transactions on Reliability*, **R-35**(2), pp. 131–136, 1986.
- [19] Nakamura, T. & Kijima, K., System of system failures: Meta methodology for IT engineering safety. *Systems Research and Behavioural Science*, **26**(1), pp. 29–47, 2009.

