An approach for hazard analysis of multiple-cooperative systems considering dynamic configuration uncertainty

Sejin Jung
Div. of Computer Science and Engineering
Konkuk University
Seoul, Republic of Korea
jsjj0728@konkuk.ac.kr

Junbeom Yoo
Div. of Computer Science and Engineering
Konkuk University
Seoul, Republic of Korea
jbyoo@konkuk.ac.kr

***Abstract*—The cooperative systems, such as cyber-physical systems (CPS) and system of systems (SoSs), are systems that operate with collaborations between numerous heterogeneous systems to accomplish common goals of the systems. These systems are often safety-critical systems since they are increasingly used to perform safety-related activities. Therefore, it is one of the important behavior to assure the safety of such systems by identifying potential hazards that lead to the accident by appropriate hazard analysis techniques. Due to the nature of the cooperative systems, such as dynamic changing structures or the existing multiple number of configurations during operations, it is necessary to consider such possible dynamic structures in the hazard analysis. However, there are several limitations to identifying hazards associated with such uncertainties through the existing hazard analysis approaches. This paper proposes an approach of hazard analysis considering dynamic configurations uncertainty for cooperative systems. The proposed approach constructs a variability information unfolding model from several system specifications and traceability analysis results and provides the process of using such information for hazard analysis. We also performed a case study to show the feasibility of the proposed approach with two systems.**

***Index Terms*—Hazard analysis, Cooperative systems, STPA, FMEA, Dynamic structures**

I. Introduction

The cooperative systems, such as cyber-physical systems (CPS) or system of systems (SoSs), are systems that share information and tasks to achieve common goals of the systems [11]. These systems also have multiple collaborative and interactive behaviors between numerous heterogeneous systems during operations. The cooperative systems are often safety-critical systems [18], for example, autonomous vehicles, unmanned aerial vehicles, or health-care CPSs are representative sorts of safety-critical cooperative systems. Since these systems are increasingly used to perform safety-related activities, it is an essential activity to demonstrate that the whole system is acceptably safe from identified hazards [19], [25] with the help of hazard analysis techniques. Hazard analysis is a systematic way to identify potential sources of harm and derive safety requirements to mitigate/eliminate the effects of failures/hazards [4].

There are several challenges related to dynamic natures to be considered in the hazard analysis of cooperative systems. Analyzing hazards and deriving safety requirements from an integrator's or cooperative perspectives is necessary for hazard analysis of cooperative systems [3], [17]. In some cases, the system structures may appear as the constitution of multiple instances, and their collaborations at runtime [12]. Infinite number of configurations during operation [8] is also possible scenarios. In other words, the characteristics of cooperative systems like "*dynamically changing structure*," and "*possibility of the multiple numbers of configurations*" [2], [8] can lead to various operation circumstances with multiple dynamic structures. These various circumstances during operation may show various contexts that are emergent or potential hazardous behaviors.

These dynamic features can cause variable structures of system configurations (compositions) during operation, and these variabilities of system configuration structures can also be a factor causing uncertainty, which this paper regarded as a dynamic configuration uncertainty for hazard analysis of cooperative systems.Therefore, hazard analysis for cooperative systems should also consider features that can generate various situations at a higher level of structures. There exist several studies for hazard analysis of collaborative/cooperative systems considering variability [11], [18], or dynamic safety assurance for the safety of collaborative CPSs at runtime [8]–[10]. The previous study proposed an information unfolding process and model [2] to perform STPA thoroughly with an unfolded control structure that considers the variable cases of control structures. However, it needs an additional method to consider such uncertainties in creating a process model of the STPA. Most existing studies do not directly cover the uncertainties about dynamically changing structures or configurations of multiple systems in hazard analysis. Furthermore, it is difficult to thoroughly consider various situations from multiple configuration structures in a typical hazard analysis

This paper was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.2021R1F1A1047246).

2640-0715/22/$31.00 ©2022 IEEE
DOI 10.1109/APSEC57359.2022.00040

approach.

This paper proposes an approach of hazard analysis considering dynamic configuration uncertainty for cooperative systems. The proposed approach provides an intermediate model, variability information unfolding model, which is based on the previous studies [2], to identify and extract possible configuration structures including operation states in cooperative systems during operation thoroughly. And next, we also guides to the use of this extracted information in the hazard analysis with guidewords. We use both STPA [15], and FMEA [4], which are representative hazard analysis techniques. Hazard analysis with the two techniques is supported using certain circumstances that can be created by the combination of the captured configuration structures and the guidewords. The proposed approach can help analysts identify diverse cases of hazards under multiple different configuration structures of cooperative systems. We also applied the proposed hazard analysis approach to the two systems, the vehicle platooning system, and the incident detection system, to show the feasibility and applicability of the proposed approach.

The remainder of this paper is as follows. Section 2 introduces hazard analysis techniques and related works as a background. Section 3 shows a concept of the uncertainty that would be handled in this paper. Section 4 explains the proposed hazard analysis approach and section 5 shows the case study and discussions of the paper. Finally, Section 6 concludes the paper and shows future directions.

II. Background

A. Hazard Analysis

Hazard analysis is a systematic way to identify potential hazards, their effects, and mitigation methods for assuring the safety of systems [4]. It is importantly applied for safety-critical systems since identifying potential hazards and mitigating the hazards by safety requirements is required by the international standards such as [19], [26]. There are a lot of hazard analysis techniques to use [4] in current, for example, failure mode and effect analysis (FMEA), hazard and operability (HAZOP) study, fault tree analysis (FTA), or system-theoretic process analysis (STPA).

FMEA is a hazard analysis technique that identifies the effects of potential failure mode of components, functions, or assemblies [4]. It is a valuable tool for evaluating the failure mode and failure rates. Performing FMEA typically starts with identifying the functions or components of the system, which are the analysis target, and their possible failure modes. The FMEA uses a worksheet table that consists of system-subsystem-function-failure mode-cause-effect-hazard-recommend action, and analysis proceeds to fill the worksheet table. There are also several variations for worksheet tables according to the purpose or scope of the analysis. However, the traditional process of the FMEA has limitations in identifying multiple failures.

STPA [15], on the other hand, is also a hazard analysis technique based on a system-theoretic accident model and process [27]. In the STPA, the system is viewed as a hierarchical structure in which higher-level components (controller) control lower-level components (controlled process). And the controlled process also generates feedback to higher-level components. Then, STPA focuses on identifying hazardous controls, called unsafe control actions (UCA), between the controller and controlled process, as shown in the ⟨Fig. 1⟩. UCAs are identified with four types which can occur in controls between components. Finally, it analyzes the causes of UCA.

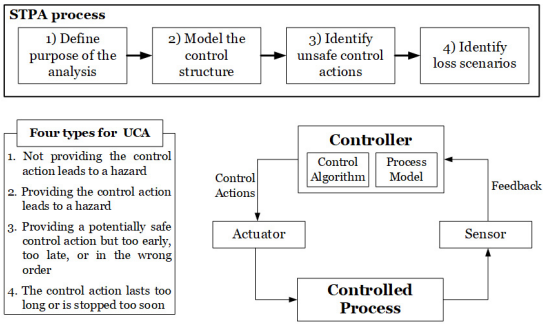

Fig. 1: A typical STPA process and the control structure [15]

STPA has received much attention for analyzing collaborative and cooperative systems such as the SoSs and CPS [25], [29], because the STPA has suitable concepts for analyzing the hazardous behavior between components in SoSs. However, cooperative systems consist of various dynamically changing elements [8], [12], so we need to consider the situations that occurring by such circumstances of structures in the hazard analysis.

B. Related Works

Several studies have tried to perform hazard analysis for SoSs or CPSs that are collaborative and cooperative systems. Ali et al. [18] propose a method for safety analysis of CPSs considering variability. The authors divided the variability, which causes uncertainties in systems, into four categories that are environmental, infrastructural, spatial, and temporal. Jaradat et al. [3] propose a modular approach for safety cases for safety assurance of industry 4.0 with considering the key characteristics for safety in industry 4.0 - *Modular and Cooperative*, *Continuous* and *On-demand*. They propose a modular safety assurance case model by the combination of the different actor's responsibilities that are system integrators and things or infrastructure providers. The authors also discussed the importance of "*identifying emerging hazards due to expected, yet unpredictable, reconfigurations or redeployments of the architecture in multiple contexts* [3]." Kabir et al. [8] proposed a framework for providing dynamic safety assurance for the cooperative system of systems to address the several challenges of cooperative SoSs, such as uncertainty and multiple sets of configurations. The framework is formed as a distributed multiagent system that has responsibilities to ob-

serve and monitor the dependability of the system at runtime, and an intelligent reasoning engine can make decisions. It can consider the dynamic configuration changes in runtime, and they do not focus on the hazard analysis of such situations.

The authors of [17] presented an extension of the ISO 26262 standard process from a single vehicle to a cooperative vehicle. The extended process allows safety analysis of cooperative driving architectures as proposing a methodology for analysis of functional safety of cooperative driving architectures. They performed hazard analysis and risk assessment process of the ISO 26262 as two perspectives, a single vehicle scenario, and cooperation vehicle scenarios. Therefore, the authors considered the hazardous events at a cooperative perspectives in vehicles with operational modes and situations. The authors less dealt with situations about multiple configuration structures with surrounding systems in the hazard analysis. The composite safety analysis approach for the system of CPSs (SoCPS) was introduced by the authors of [11]. They called the approach as SafeSoCPS. They analyzed potential hazards for the network of CPSs and traced the faults among the whole of participating CPSs. The traceability analysis among faults in consittuent systems is connected by the relationships that are proposed by the authors.

In this regard, hazard analysis for SoS has also been tried and studied in several ways [28], [30], [31]. Baumgart et al. [28] proposed a hierarchical process to document system-of-systems specifications with their interactions in behaviors. They also provide some guidewords, such as '*No*,' '*More*,' '*Late*' that are possible to apply to the safety analysis for each level of SoSs structure. However, these studies lack consideration of the variability which could be derived from the dynamically changing structures in hazard analysis. Axelsson et al. [29] proposed a hazard and risk analysis method for SoSs based on system thinking theory. It aims at coping with risks for SoSs and deriving safety requirements on the constituent system for reducing the emergent risks of the SoSs. The paper provides control diagram concepts that consist of a constituent system and coordinator to represent and handle the SoSs level model. Risk analysis with the STPA process first starts at the SoS level loss and hazard analysis. Interface hazard analysis is also importantly applied to hazard analysis for SoSs [32]. The various way of hazard analysis for SoSs and CPSs, which are used in cooperative systems, has been studied, nevertheless, there still need for method to identify hazards in dynamic configuration uncertainties.

III. Dynamic configuration uncertainties from Variability of dynamically changing structures

Uncertainty in CPSs is usually defined as "*the lack of certainty (i.e., knowledge) about the timing and nature of inputs, the state of a system, a future outcome, as well as other relevant factors* [1]." There are many factors that create uncertainties, such as dynamic and unpredictable environments, inaccuracy or indeterminism of dynamic systems, limited knowledge of other systems, and temporal variability factors [8], [14], [18]. Among the numerous variability factors of uncertainties that can be found in cooperative systems, this paper focuses on the uncertainty that can occur by dynamic features cooperative systems such as "*dynamically changing structure* [2]," and "*possibility of the multiple numbers of configurations*" during operation. Multiple instances of the same type systems are also possible cases [12] in cooperative systems. These features can cause multiple structures of system configurations (compositions) including surrounding systems, and they result in various operation circumstances about multiple configuration structures and system states during operation. Consequently, these various circumstances are a kind of variability factor causing uncertainty, which this paper regards it a dynamic configuration uncertainty, which should be considered in the hazard analysis.

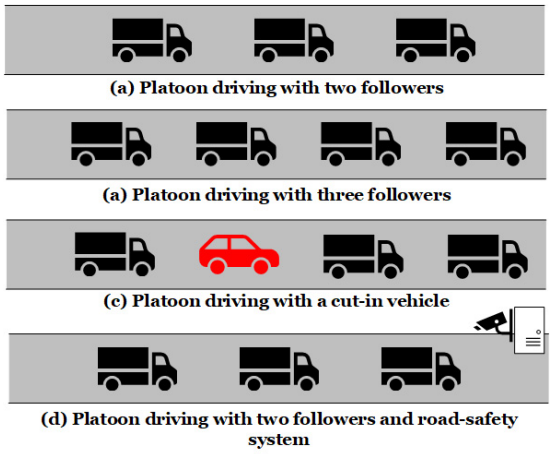

Fig. 2: Four examples of possible configuration structures of the platooning systems on the road

⟨Fig. 2⟩ shows four examples of possible structures that can show features of dynamic configuration uncertainties occurring by the variability of dynamically changing structures. The operation states of platoons might be varied according to the changes in the composition of vehicles and surrounding systems. ⟨Fig. 2 (a)⟩ and ⟨Fig. 2 (b)⟩ show a normal scenario of configuration structures in platoon driving that consists of one leader and multiple follower vehicles. On the other hand, ⟨Fig. 2 (c)⟩ ⟨Fig. 2 (d)⟩ represent circumstances with surrounding systems like non-platoon cut-in vehicle or incident detection system. System nodes in such variable configuration structures can be classified as intrasystem and intersystem. Intrasystem means the system nodes belong to the same system, and the intersystem is represented by other external systems [7]. According to the composition and scope of the cooperative systems, intra- and inter-systems nodes can exist under dynamic operation circumstances when the analysis target is selected.

The variable configuration structures and their changes can be a hazardous state (*i.e., hazard*) itself, or they can be considered as a triggering condition that leads to the hazards. These cases also can be considered in combination with

conditions like guidewords HAZOP (hazard and operability) [4]. An identical activity of system function, in other words, can be hazardous in particular circumstances of itself and surrounding systems. The normal behavior of acceleration of automotive vehicles with platooning systems may lead to an accident when the surrounding incident alarm system on the roadway sends incorrect information. Therefore, considering the dynamic configuration uncertainty is also an important issue for hazard analysis of cooperative systems. It also needs to identify hazards from a cooperation perspective of systems [16], [17].

IV. The proposed hazard analysis approach

This section introduces the hazard analysis approach considering the dynamic configuration uncertainty for cooperative systems proposed in this paper. The approach supports performing the hazard analysis by providing supplementary information about various configuration structures and application perspectives as guidewords. Section A shows an overview of the proposed approach and Section B introduces the extended intermediate model, called variability information unfolding model. Section C explains hazard analysis using the extracted variability information.

A. Overview of the proposed approach

⟨Fig. 3⟩ shows an overview of the proposed approach. Identifying the scope and purpose of the target system and surrounding systems is necessary before the analysis, like other typical hazard analysis techniques. In this step, various information about elements in dynamically changing structures of systems should be identified and collected from available system specifications such as plans or software requirements specification (SRS). Individual systems or multiple instances of systems are all possible examples of multiple elements.

Relationships such as controlling relations, interactions, or connections between these components should also be identified because those are important information to reveal the dynamically changing structures. Such relations can also be revealed between different modes in multiple instances of the same systems with different responsibilities. We use the traceability/connectivity relationship analysis for CPSs proposed in the [22] to set and identify the scope/range of the systems.

(Step 1) This step constructs an intermediate model to extract/identify the possible configuration structures and their states in cooperative systems during operation. We extend the information unfolding model (IUM), which is proposed in the [2] previously, to encompass the surrounding systems. The extended model, called the variability information unfolding model (VIUM), represents the possible structural changes of configurations. The model also contains a hierarchical finite state machine (FSM) internally because the FSMs behave differently depending on the current state or mode, so they could play a role in finding various combinations of configuration structures.In this step, analysts have to consider the identification of multiple composed system elements and changed structures. Various relationships between systems, system elements, and development artifacts, such as traceability link types about *internal interaction node*,' '*multi-mode item*,' or '*external interaction node*' in the [22], can serve as the basic foundation for identifying artifacts that are used to establish uncertainty information for supporting hazard analysis.

Before proceeding the step 2, unfolding the model and capturing each structure, which appears to changes of configuration structures and states, from the VIUM is required. It is before a "*combine captured structures and guidewords*" substep, as shown in the middle of the ⟨Fig. 3⟩. These captured structures and states will be employed in the hazard analysis with guidewords to provide various contexts that are possibly hazardous scenarios.

(Step 2) We suggest four ways use the circumstances, which are the combination of captured structures and guidewords, for two hazard analysis techniques, FMEA and STPA. As shown in the ⟨Fig. 3⟩, several guidewords could be combined with the extracted structures to provide a division of such situations in this step. The combination of captured structures and guidewords (*i.e.* a certain operation circumstances) can help analysts consider the hazards under such circumstances additionally and thoroughly according to the method of usage. The red rectangular labeled in each hazard analysis technique in the ⟨Fig. 3⟩ are the point of using the information in four ways proposed in this paper. The extracted circumstances play a role in providing supplementary contexts for hazard analysis to help determine whether the situation can be hazardous or not with these circumstances. It will help analysts identify hazardous behavior in intended functionality or hazardous function itself.

B. Constructing the variability information unfolding model

This section introduces the variability information unfolding model (VIUM) proposed in this paper. The VIUM extends from the IUM (information unfolding model) proposed in the [2], as mentioned earlier. It is also based on the hierarchical finite state machine (FSM) to represent variability information as constructing variable configuration structures of systems. While the IUM is a model for helping STPA analysts list all possible combinations of control structure thoroughly, we extend it to be able to represent system configuration changes, including surrounding systems with some additional identifiers.

A simplified definition of the variability information unfolding model (VIUM) is as follows:

$\texttt{VIUM} = \langle N, I, n_i \rangle$, where

- N : a set of individual systems
- I : a set of transitions, $N \times N$
- n_i : an intra system (a target system for hazard analysis), an element of N

The model consists of systems represented as a node, N, and their interactions as transitions, where the n_i is a target system of hazard analysis. All systems except n_i, are surrounding

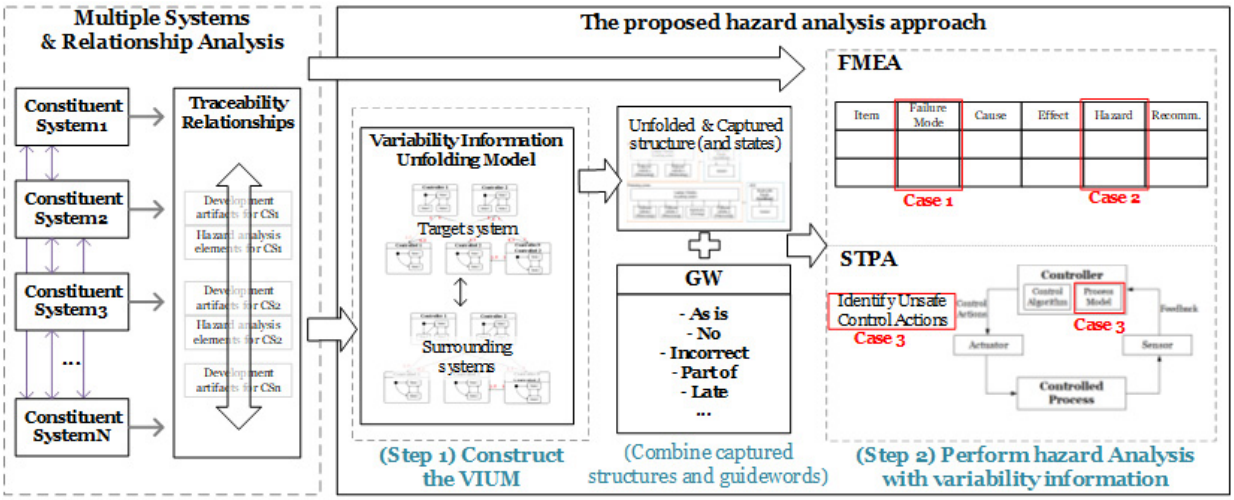

Fig. 3: An overview of the proposed hazard anaylsis approach

systems possibly connected/interacted in a dynamic environment. Transitions I are any connections/relations/interactions between system nodes. This transition element can also be a point of interface hazard analysis of system of systems. The N would also have various internal structures. We define these structures as a model based on the FSM with IUM. A definition of the system node is as follows:

$N = \langle \mathtt{E}, \mathtt{C} \rangle$, where

- $\mathtt{E} = \langle S, L, T \rangle$
  - S : a finite set of states (Modes)
  - L : a set of transition labels
  - T : a set of transitions, $S \times L \times S$
- $\mathtt{C} = (T, M, c)$
  - T : a set of transitions, $\mathtt{E} \times M \times c \times \mathtt{E}$
  - M : a set of pairs of multiplicities
  - c : a label for control relationship, {T, F}

The principal notation does not differ from the IUM [2]. The node N is a structural model based on FSM definitions consisting of nodes and transitions, whereas the node $\mathtt{E}$ is also FSMs of modes/states. In some cases, the $\mathtt{E}$ can be both entities for system instances or component entities in systems. And transitions $\mathtt{C}$ are controlling relations or interaction/connection relations between nodes, therefore, it has a label for marking whether the transition is in a controlling relationship. If the control relationship of c is true, it can be used to construct various control structures, as explained in the [2]. Otherwise, this transition means two entities have interactions or connections relationships, and they can also be a candidate for interface hazard analysis at cooperative perspectives. A multiplicity in the transition label means that the configuration structure can have multiple elements of system/component entities.

⟨Fig. 4⟩ is an example of the VIUM. In the figure, the 'System 1' is a target system that is n_i in the VIUM, and this system will have various configuration structures in operation. For example, 'System 1' operates under the structures that have (Controller 1), which has three modes, and other elements such as controls 1..3 (Controlled 1) and 0..3 (Node 2) according to the multiplicity symbols. 'System 2' also has nodes represented in the $\mathtt{E}$ elements of the model. The important thing in modeling the configurations by using the VIUM is that the elements, $\mathtt{E}$ in the N, mean individual instance elements of systems.

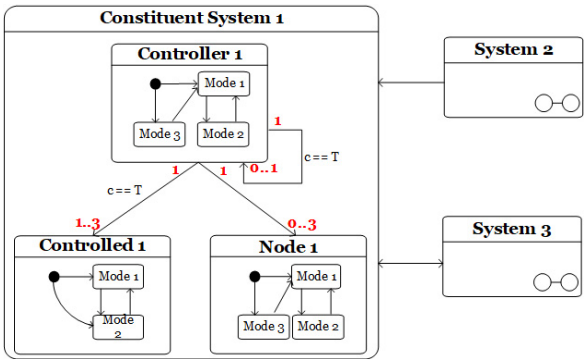

Fig. 4: An example VIUM

Before performing the hazard analysis, as mentioned earlier, we should extract and identify various structures and states in the dynamic operations of systems from the VIUM. For example, we construct 24 cases of structures of the 'System 1' from the VIUM in the ⟨Fig. 4⟩. Among the various structures, one operating structures in which the system structures consist of one 'Controller 1,' two 'Controlled 1,' and one 'Node 2' is possibly identified. And each system node in these various configuration structures can also be distinguished by the internal states of the model. The VIUM is used to identify various configuration structures and states by unfolding all

possible structures of cooperative systems. As a result, the analysts can capture structures/states thoroughly in dynamic operations for hazard analysis. Generating unfolding structures of one system is the same as the algorithm introduced in the [2], and this paper applies the algorithm as many times as the system exists in the model.

C. Hazard analysis with variability information

This step performs hazard analysis with the variability information that can be represented as various circumstances with guidewords. The basic principle of performing hazard analysis in this step is the same as the typical process of the STPA and FMEA. The only difference is the additional thought about the possibilities of hazards under the identified circumstances. Then, the guidewords are used so that analysts can consider various scenarios with the identified configuration structures in the hazard analysis. It would be able to broaden the thinking of the hazards under various situations from configuration changes.

The guidewords have been used in several hazard analysis techniques like FMEA or HAZOP. The traditional HAZOP analysis uses several guidewords, such as 'No,' 'Less,' 'Part of,' Etc. [4]. Various types and contents for guidewords also have been proposed, for example, accuracy, security, or functionality in software hazard analysis [5], [6] and the guidewords also used in hazard analysis of SoSs [28]. The followings are examples of guidewords that are possibly combined with the identified configuration structures in this paper, as shown in the ⟨Fig. 3⟩.

- **As is**: This guideword applies a captured circumstance, structures, and states of the composed system nodes to the hazard analysis as they are.
- **No (Fail)**: This guideword assumes that a system node in a captured configuration structure does not operate or fails to operate/perform.
- **Incorrect (Behavior, value)**: The intention of the '*Incorrect*' guideword is to assume that a system node does not operate correctly for its purposes, such as unintended behavior or incorrect output.
- **Part of**: The '*Part of*' is used to assume situations which only a part of the system nodes (entities) in the captured configuration structure operates correctly.
- **Late**: This means that a certain system node's behavior operates too late.

A one example of combination is about when combining '*Part of*' guidewords and ⟨Fig. 2 (d)⟩, analysts can get a circumstance about "platoon driving vehicles with two followers and road-safety system exist then part of vehicles does not operate correctly." Other guidewords can also apply to the various structures generated from VIUM. Hazard analysis concretely uses these circumstances as various ways to identify additional/supplementary hazardous scenario, behavior or controls.

This paper proposes four possible points to apply the identified circumstances to hazard analysis. Case 1 $\sim$ 3 of the ⟨Fig. 3⟩ on the FMEA/STPA in the (STEP 2) correspond to it. Case 1 in the ⟨Fig. 3⟩ uses the circumstances, which are guidewords and configuration structure, as the failure mode of the analysis target system. For example, identifying which effects occur if the connected systems (both intra- or inter-system nodes) in the captured configuration structures fail to operate using the 'No (Fail)' guidewords. This usage is similar to other hazard analysis approach for SoSs [28]. The application method of Case 2 uses the circumstances as context information to provide additional information to judge whether the failure mode and analyzed effects of the target item are hazardous. This usage is similar to the context of unsafe control action (UCA) in STPA.

The next two usages are involved in the two steps of STPA, which are the steps of identifying unsafe control action and constructing a control structure. They both relate to the context of UCAs. In the case of the process model, the variability information is applied to construct the process model, which relates to the context with multiple control structures [2], while the case of UCA uses the variability information as the context directly for identifying unsafe control actions.

Then, analysts should significantly choose the guidewords according to the application point and captured situations. Not all the guidewords fit every situation and system relationship. A distinction between intra-system and inter-system is also critical to determine to guidewords applying. The 'As is' guideword is unsuitable for identifying failure mode in the FMEA, on the other hand, it may provide helpful information to identify hazards from the effects of the failure mode. We introduce a case study about performing hazard analysis with the proposed approach in the next section.

V. Case Study

We performed a case study to show the applicability of the proposed hazard analysis approach with two systems. The target system is a vehicle platooning system [20], [21] which operates in automotive vehicles on the road, and the road contains an automatic incident detection system (AIDS) [24] to detect and provide alarms of the incidents. This section introduces the target system and analysis results of the case study. ⟨Fig. 5⟩ shows a conceptual overview of the platooning system and AIDS in the case study.

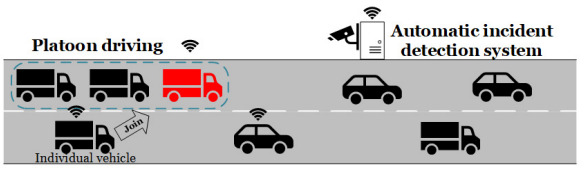

Fig. 5: A conceptual overview of the platooning system and AIDS

A. Target Systems: vehicle platooning systems with automatic incident detection system

We use the vehicle platooning system example [22] and the automatic incident detection system example [24] in the

case study. The case study uses as system specifications preliminary functional requirements and software requirements specification in [22]. The vehicle platooning system is a kind of cooperative automotive system for enhancing traffic capacity and energy efficiency, and AIDS detects several incidents on the road and sends alarms to the vehicles. A one-leader vehicle leads and controls the following vehicles composing the platoon as described in ⟨Fig. 5⟩. It also has many functions such as *create/join/leave platoon*, *merge*, *split*, *acceleration/deceleration*, *leader change*, and so on. We did not include the functions and details related to automotive driving itself in the case study. The *join/leave* function of the platooning system is 'joining an external vehicle into the platoon' and 'exporting the following vehicle from the platoon,' respectively. The *merge* function combines two platoons into one platoon with one leader vehicle, on the other hand, *split* function divides one platoon into two platoons. Other functions also have their own operations.

AIDS is a kind of support system on the road for safety. We use the requirements of AIDS in Korea [24] for the case study. The basic functional requirements of AIDS are detecting sudden incidents on the road, distinguishing and determining that the detected incident is a valid case, and sending alarms through various media such as electronic displays, infrastructure network messages, or radio. It consists of two sub-systems, which are the roadside equipment subsystem, including sensor, radar, or camera, and the main-control sub-system that stores incident data and manages them.

B. A summary of the case study

This section shows a summary of the hazard analysis for the vehicle platooning system with the proposed approach. We first construct the VIUM from various system specifications and their traceability analysis results. ⟨Fig. 6⟩ shows a part of traceability relationships for the platooning system and AIDS, which are proposed in [22], [23]. It shows various relationships between development artifact elements, such as functional requirements in software requirements specifications, and interaction-related relationships are also identified. We also checked the existence of various modes of the platooning system (e.g., leader, follower, external mode) and several subsystems in the AIDS.

We, consequently, identified three elements *leader*, *follower*, and *non-platoon (external) vehicle* in the platooning system and two elements *road-side equipment* and *Center* in the AIDS from the specifications manually. The three elements of the platoon system are individual entities of systems that are different modes and have responsibilities, and the two elements of the AIDS are component entities. They also have interactions, as shown in the ⟨Fig. 6⟩. For example, link types such as '*External interaction node*' and '*Internal interaction node*' represent these connections.

⟨Fig. 7⟩ shows the VIUM of the platooning system and AIDS in the case study. The relationships between elements and an individual FSM for each element are also identified. The arrow between the vehicle platooning system and

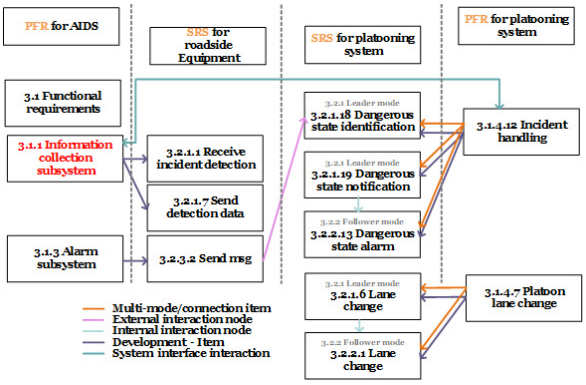

Fig. 6: A part of traceability analysis results [22], [23]

AIDS means they have interactions in operations. Elements in each system exist in multiple instances during operations, for example, the leader can have multiple followers and can also interact with multiple external vehicles while controlling relations exist between identified elements with multiplicities. AIDS also has similar multiplicities, as shown in the figure.

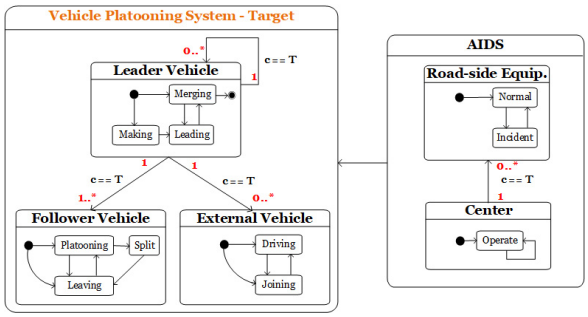

Fig. 7: A VIUM for platooning system and AIDS in the case study

According to the proposed approach, we need to unfold and identify the possible configuration structures from the VIUM and combine the guidewords to generate circumstances. Unfolding the model is based on the algorithms introduced in the previous study [2] by exhaustive search and cartesian product of the model. ⟨Fig. 8⟩ shows three cases of configuration structures generated from the VIUM while assuming * is up to 5. The inside text of parenthesis of each box (node) means states of the system respectively. ⟨Fig. 8 (a)⟩ represents a structure that is a case of ⟨Fig. 2 (d)⟩. And ⟨Fig. 8 (b)⟩ represents a operation state that a one leader leads the platoon driving that consists of two followers and one other leader exist near the platoon. Then, one follower is leaving the platoon and the other leader is performing the merge function.

⟨Fig. 9⟩ and ⟨Fig. 10⟩ are analysis results of FMEA and STPA for the vehicle platooning system, respectively. We did not include the further step of the hazard analysis, which is identifying causes and losses of unsafe control actions, in

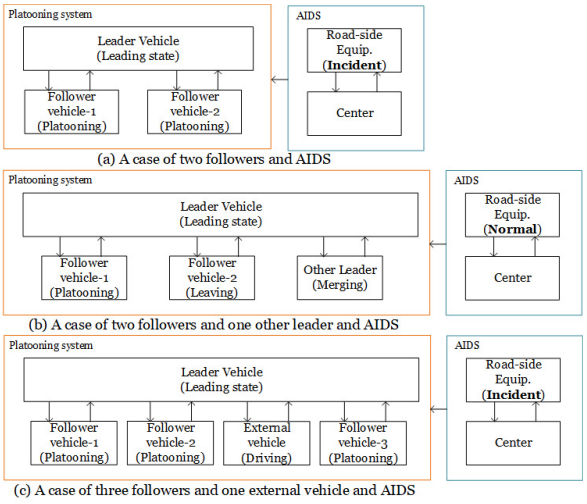

Fig. 8: Three examples of configuration structures generated from ⟨Fig. 7⟩

this case study. Using the configuration structure described in the ⟨Fig. 8⟩ combined with guidewords, analysts, consequently, get help to identify several additional possibilities of unsafe/hazardous events, failure modes, or situations which are underlined and bold texts in the ⟨Fig. 9⟩ and ⟨Fig. 10⟩. For example, the failure mode of '*Part of followers fail to operate (Fig. 8 (a) + part of)*' means a specific situation in the ⟨Fig. 8 (a)⟩ and then part of followers fail to operate when analyzing the lane change function of the leader mode vehicle. The third row of '*Merge*' function in the ⟨Fig. 9⟩ is not quite different from hazard analysis results of fail to merge request behavior. However, it has been meaningful in providing different aspects of analysis like identifying failure effects of leader vehicles, which receives the merge request, when other leader fails its function.

The results of STPA also shows several UCAs with the identified circumstances. UCA No. 4, 10, 11, and 12 are examples of UCAs under certain circumstances, which are further identified using the VIUM and guidewords. The UCA 11 is almost the same as the UCA introduced in the previous study [2] except for the differences between acceleration and deceleration, meanwhile, the UCA 10 and 12 are added scenarios that consider the structures and states of surrounding systems. Hazardous controls also occur in various situations in intra-/inter-system relationships.

As a result, we could identify several hazardous failure modes, hazards, and UCAs that can emerge under the various circumstances about variable configuration structures of multiple-cooperative systems. The proposed approach can help analysts check and find hazards occur by dynamic configuration uncertainties in dynamic operations. So it offers opportunities to identify various scnearios/cases of hazardous/unsafe events and failure modes thoroughly. These contexts are not easy to elicit in typical hazard analysis process. It, nevertheless, is not that straightforward to find all of such possible structures and situations, especially in cooperative systems that show dynamic configuration uncertainties. It would provide the basic foundation for identifying various structures for supporting hazard analysis.

C. Discussions

The case study shows that the proposed approach can support the hazard analysis to identify hazardous failure modes or UCAs for cooperative systems with supplementary contexts, as shown in the ⟨Fig. 9⟩ and ⟨Fig. 10⟩. The identification of hazards under such features is sometimes hard to analyze with the typical approach of hazard analysis. The hazard analysis for SoSs mainly focuses on identifying hazards in communication or collaborative behavior at SoSs-level architecture according to the characteristics of SoSs [28]. The interface, interoperability, resource, and proximity hazards are representative categories for hazards in SoSs [30]. The transitions in VIUM, between platooning system and AIDS in the case study example, also can be a target item for interface hazards analysis. The authors of [17] introduces several hazards of cooperative function of the platooning system such as "Platoon does not merge with other platoon when this is desired," "Vehicles in platoon do not keep enough distance," and so on. We think the proposed approach of this paper may show another aspects of cooperation hazards under various circumstances in dynamic operations. And hazards, which are identified/assumed in this paper, are helpful to develop safety requirements for cooperative functions.

However, the proposed approach cannot always cover all possible structures, of course, because it depends on constructing the VIUM thoroughly. The feature, *Dynamic*, has many more issues that need to be considered in safety analysis, except as discussed in this paper. There exist several studies for dynamic safety assurance of CPSs or cooperative systems of systems [8], [9], or the safety case for assurance [3]. It is usually considered that foreseeing all potential configurations and scenarios of cooperative systems at design time is certainly not possible [8]. The VIUM may be extended with such reasoning and a dependability model for representing dynamics in the nature of cooperative systems. Therefore, the proposed approach of this paper should also consider such possibilities of changes in constructing the VIUM.

Another issue is the complexity of the VIUM. The more complex the VIUM becomes, the more difficult is increased because the purpose of the proposed approach produces all possible structures from searching the VIUM mechanically. In the worst case, it may be impossible to predict the size of results from the exponential size of elements of VIUM. To reduce the complexity of the VIUM and generated structures, appropriate method and studies should be performed. And it might rarely seem possible to apply the proposed approach without a CASE (computer-aided software engineering) tool. A systematic and automatic way to extract and construct variability information and model would help perform hazard analysis efficiently.

System: Platoon system, Sub-system: Platoon controller, Component: Leader mode					
Function	**Failure mode**	**Causes**	**Immediate effect**	**System effect**	**Hazard**
Lane change	Fails to operate lane change function	Network error of leader	Leader vehicle fails to control the follower's driving lane change	Followers do not change the lane	Followers drive in a different lane than the leader Platoon driving operates in incorrect status
	Incorrect lane change	TBD	Leader send the incorrect lane change command to followers Follower try to change driving lane	Followers change to a driving lane different to the leader's	Followers drive in a different lane than the leader Platoon driving operates in incorrect status
	Part of followers fails to operate (Fig. 8 (a) + Part of)	TBD	-	Part of the followers do not change the driving lane	Platoon driving operates in incorrect status
Deceleration	Fails to operate deceleration	Network error of leader	Leader vehicle fails to occur deceleration to followers	Followers does not decelerate	Vehicle-to-vehicle distance below safe distance
					Platoon may not maintain safe velocity **under incident (Fig. 8 (a) + As is)**
	Incorrect value of deceleration	TBD	Leader vehicle operates decelerate function at incorrect speed	Followers decelerate incorrect speed according to the leader operation	Vehicle-to-vehicle distance does not maintain appropriately
Merge	Fails to merge	TBD	Leader fails to merge function	Two platoons drive without merging respectively	-
		TBD	Leader does not operate the recognition of the combined vehicles correctly	Leader vehicle does not update the merging vehicle	Platoon driving operates in incorrect status
	Incorrect behavior of merge operation	TBD	Leader merges with incorrect platoon	Merging two platoons does not occur correctly	The platoon which desires to merge is driving separately without merging
		TBD	Leader merges other platoon vehicles partially	Merging two platoons does not complete successfully	Platoon driving operates in incorrect status
	Other leader fails to operate (Fig. 8 (b) + No (Fail))	TBD	-	Merging two platoons does not complete successfully	Platoon driving operates in incorrect status

Fig. 9: Analysis results of FMEA for the vehicle platooning system (excerpt)

Control Action	Not providing causes hazard	Providing causes hazard	Too late, Too soon, Out of order	Stopped too soon, Applied too long
Lane change	[UCA1]Leader mode platoon controller does not provide a lane change command to followers when leader changes the driving lane	[UCA2]Leader mode platoon controller provides a lane change command to followers when leader drives with maintaining lanes	[UCA3]Leader mode platoon controller provides a lane change command to followers too late when leader changes the lane	
			[UCA4]Leader mode platoon controller provides a lane change command too soon **when leaving and merging function has not been completed (Fig. 8 (b) + Late)**	
Deceleration	[UCA5]Leader mode platoon controller does not provide deceleration command to followers when the leader decelerate under emergency situation	[UCA6]Leader mode platoon controller provides deceleration command to followers without emergency situation [UCA7]Leader mode platoon controller provide deceleration command to followers when the vehicle-to-vehicle distance is under safe distance	[UCA8]Leader mode platoon controller provides deceleration command to followers too late when the leader decelerate under emergency situation	[UCA9]Leader mode platoon controller stop the deceleration command too soon when the follower did not decelerate enough
	[UCA10]Leader mode platoon controller does not provides deceleration command to followers **while AIDS fails to operate its behavior under incidents (Fig. 8 (a) + No (Fail)**	[UCA11]Leader mode platoon controller provides deceleration command to followers **while a non-platooning (external) vehicle is driving in cut-in the platoon (Fig. 8 (C) + as is)**		[UCA12]Leader mode platoon controller stop the deceleration command too soon **while AIDS is under an incident state (Fig. 8 (a) + as is)**
Merge	[UCA13]Leader mode platoon controller does not provide merge command to the other leader when desired	[UCA14]Leader mode platoon controller provides merge command to unrelated platoon	[UCA15]Leader mode platoon controller provides merge command to the other leader too late than requested	

Fig. 10: Analysis results of STPA for the vehicle platooning system (excerpt)

VI. Conclusions and Future Works

This paper proposes a hazard analysis approach for cooperative systems, which shows the dynamic configuration uncertainties. We provide a variability information unfolding model, which extended from the information unfolding model [2] in this paper. The proposed immediate model can model the variability of structures and thoroughly identify possible configuration structures of systems by unfolding the model. This paper also provides guidewords that can generate a situation corresponding to the captured structures. The circumstances, which is identified by combining a captured configuration and guidewords, is used for the hazard analysis in four points according to the hazard analysis process. They are the failure mode and hazard clause of the FMEA table and identification of the context of UCA in the STPA. We also performed a case study with two systems that are the vehicle platooning system and automatic incident detection system. The case study shows that identifying some UCAs and failure modes that are possible in certain circumstances. The proposed approach will help safety analysts consider and identify hazardous failure modes and UCAs of cooperative systems that are possibly generated from various configuration structures under dynamic configuration uncertainty. We are now planning the development of CASE tool for applying the proposed approach (semi-)automatically in hazard analysis, and also have a plan to develop the method of using several reasoning models like other dynamic safety assurance methods [8] to construct VIUM effectively.

References

[1] Zhang, Man and Selic, Bran and Ali, Shaukat and Yue, Tao and Okariz, Oscar and Norgren, Roland, "Understanding uncertainty in cyber-physical systems: a conceptual model," European conference on modelling foundations and applications, pp.247–264, 2016.

[2] Jung, Sejin and Kim, Eui-Sub and Yoo, Junbeom, "Unfolding Hidden Structures in Cyber-Physical Systems for Thorought STAP Analysis," IEICE Transactions on Information and Systems, Vol.E105-D, No.5, PP.1103-1106, 2022.

[3] Jaradat, Omar and Sljivo, Irfan and Hawkins, Richard and Habli, Ibrahim, "Modular Safety Case for the Assurance of Industry 4.0," Safety-Critical Systems Symposium, York, 2020.

[4] Ericson, CLIFTON A. "Hazard Analysis Techniques for System Safety," Wiley, 2015.

[5] Lawrence, J Dennis, "Software Safety Hazard Analysis," Nuclear Regulatory Commission, 1996.

[6] Jung, Sejin and Yoo, Junbeom and Lee, Young-Jun, "A practical application of NUREG/CR-6430 software safety hazard analysis to FPGA software," Reliability Engineering & System Safety, Vol.202, 2020.

[7] Betz, John W. and Titus, Bryan M. "Intersystem and intrasystem interference with signal imperfections," PLANS 2004. Position Location and Navigation Symposium (IEEE Cat. No.04CH37556), pp.558-565, 2004.

[8] Kabir, Sohag and Papadopoulos, Yiannis, "Computational intelligence for safety assurance of cooperative systems of systems," IEEE Computer, Vol.53, No.12, pp.24–34, 2020.

[9] Pop, Paul and Scholle, Detlef and Šljivo, Irfan and Hansson, Hans and Widforss, Gunnar and Rosqvist, Malin, "Safe cooperating cyber-physical systems using wireless communication: The SafeCOP approach," Microprocessors and microsystems, Vol.53, pp.42-50, 2017.

[10] Kabir, Sohag, "Internet of thIngs and safety assurance of cooperatIve cyber-physIcal systems: opportunItIes and challenges," IEEE Internet of Things Magazine, Vol.4, No.2, pp.74-78, 2021.

[11] Ali, Nazakat and Hussain, Manzoor and Hong, Jang-Eui, "SafeSoCPS: A Composite Safety Analysis Approach for System of Cyber-Physical Systems," Sensors, Vol.22, No.12, 4474, 2022.

[12] Daun, Marian and Brings, Jennifer and Bandyszak, Torsten and Bohn, Philipp and Weyer, Thorsten, "Collaborating Multiple System Instances of Smart Cyber-physical Systems: A Problem Situation, Solution Idea, and Remaining Research Challenges," 2015 IEEE/ACM 1st International Workshop on Software Engineering for Smart Cyber-Physical Systems, pp.48-51, 2015.

[13] Kim, Eui-Sub and Yoo, Junbeom, "A Study on Application of STPA in Safety Analysis of Platoon System," 2020 Korea Conference on Software Engineering, pp.193-196, 2020. (In Korean)

[14] ISO, "Road vehicles - Safety of the intended functionality," International Organization for Standardization (ISO/PAS 21448), 2019.

[15] Leveson, N. G., "Engineering a Safer World: Systems Thinking Applied to Safety," The MIT Press, Cambridge, 2016.

[16] Kochanthara, Sangeeth and Rood, Niels and Saberi, Arash Khabbaz and Cleophas, Loek and Dajsuren, Yanja and van den Brand, Mark, "A functional safety assessment method for cooperative automotive architecture," Journal of Systems and Software, Vol.179, pp.110991, 2021.

[17] Kochanthara, Sangeeth and Rood, Niels and Cleophas, Loek and Dajsuren, Yanja and van den Brand,"Semi-automatic architectural suggestions for the functional safety of cooperative driving systems," 2020 IEEE International Conference on Software Architecture Companion (ICSA-C), pp.55-58, 2020.

[18] Ali, Nazakat and Hussain, Manzoor and Hong, Jang-Eui, "Analyzing safety of collaborative cyber-physical systems considering variability," IEEE Access, vol.8, pp.162701–162713, 2020.

[19] ISO, "Road vehicles - Functional safety (ISO 26262)," 2018.

[20] Fakhfakh, Faten and Tounsi, Mohamed and Mosbah, Mohamed, "Vehicle platooning systems: Review, classification and validation strategies," International Journal of Networked and Distributed Computing, Vol.8, No.4, pp.203-213, 2020.

[21] Jia, Dongyao and Lu, Kejie and Wang, Jianping and Zhang, Xiang and Shen, Xuemin, "A survey on platoon-based vehicular cyber-physical systems," IEEE communications surveys & tutorials, Vol.18, No.1, pp.263-284, 2015.

[22] Jung, Sejin and Kim, Eui-Sub and Yoo, Junbeom, "A Traceability Analysis for Integrated Relationship Analysis of Development/Safety Artifacts of Cyber Physical Systems," Journal of KIISE, Vol.48, No.1, pp.107-118, 2021. (In Korean)

[23] Jung, Sejin, "A Comprehensive Relationship Analysis for Heterogeneous Artifacts in Multiple-Collaborative Safety-Critical Systems," PhD thesis, Konkuk University, 2022.

[24] ITSK-00103-1, "Standard for Automatic Incident Detection System - Part 1. Basic requirements," ITS Korea, 2015 (In Korean).

[25] Bolbot, Victor and Theotokatos, Gerasimos and Bujorianu, Luminita Manuela and Boulougouris, Evangelos and Vassalos, Dracos, "Vulnerabilities and safety assurance methods in Cyber-Physical Systems: A comprehensive review," Reliability Engineering & System Safety, Vol.182, pp.179-193, 2019.

[26] IEC, "Functional safety of electrical/electronic/programmable electronic safety-related systems (IEC 61508)," 2010.

[27] Leveson, N. G., "Safeware: System Safety and Computers," Addison-Wesley, 1995.

[28] Baumgart, Stephan and Fröberg, Joakim and Punnekkat, Sasikumar, "A Process to Support Safety Analysis for a System-of-Systems," 2020 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW), pp.61-66, 2020.

[29] Axelsson, Jakob and Kobetski, Avenir, "Towards a risk analysis method for systems-of-systems based on systems thinking," 2018 Annual IEEE International Systems Conference (SysCon), pp.1-8, 2018.

[30] Baumgart, Stephan and Fröberg, Joakim and Punnekkat, Sasikumar, "Analyzing hazards in system-of-systems: Described in a quarry site automation context," 2017 Annual IEEE International Systems Conference (SysCon), pp.1-8, 2017.

[31] Muram, Faiz Ul and Javed, Muhammad Atif and Punnekkat, Sasikumar, "System of systems hazard analysis using HAZOP and FTA for advanced quarry production," 2019 4th International Conference on System Reliability and Safety (ICSRS), pp.394-401, 2019.

[32] Redmond, Patrick J and Michael, James Bret and Shebalin, Paul V, "Interface hazard analysis for system of systems," 2008 IEEE International Conference on System of Systems Engineering, pp.1-8, 2008.