KIPS Trans. Softw. and Data Eng.
Vol.10, No.8 pp.287~300
ISSN: 2287-5905 (Print), ISSN: 2734-0503 (Online)

https://doi.org/10.3745/KTSDE.2021.10.8.287

Securing Safety in Collaborative Cyber-Physical Systems Through Fault Criticality Analysis

Manzoor Hussain† · Nazakat Ali† · Jang-Eui Hong††

ABSTRACT

Collaborative Cyber-Physical Systems (CCPS) are those systems that contain tightly coupled physical and cyber components, massively interconnected subsystems, and collaborate to achieve a common goal. The safety of a single Cyber-Physical System (CPS) can be achieved by following the safety standards such as ISO 26262 and IEC 61508 or by applying hazard analysis techniques. However, due to the complex, highly interconnected, heterogeneous, and collaborative nature of CCPS, a fault in one CPS's components can trigger many other faults in other collaborating CPSs. Therefore, a safety assurance technique based on fault criticality analysis would require to ensure safety in CCPS. This paper presents a Fault Criticality Matrix (FCM) implemented in our tool called CPSTracer, which contains several data such as identified fault, fault criticality, safety guard, etc. The proposed FCM is based on composite hazard analysis and content-based relationships among the hazard analysis artifacts, and ensures that the safety guard controls the identified faults at design time; thus, we can effectively manage and control the fault at the design phase to ensure the safe development of CPSs. To justify our approach, we introduce a case study on the Platooning system (a collaborative CPS). We perform the criticality analysis of the Platooning system using FCM in our developed tool. After the detailed fault criticality analysis, we investigate the results to check the appropriateness and effectiveness with two research questions. Also, by performing simulation for the Platooning, we showed that the rate of collision of the Platooning system without using FCM was quite high as compared to the rate of collisions of the system after analyzing the fault criticality using FCM.

Keywords : Safety, Composite Hazards Analysis, Fault Criticality, Cyber-Physical Systems, Platooning System

협업 사이버물리시스템의 결함 치명도 분석을 통한 안전성 확보

Manzoor Hussain† · Nazakat Ali† · 홍 장 의††

요 약

협업 사이버-물리 시스템(Collaborative Cyber-Physical Systems, CCPS)은 물리 세계와 사이버 세계가 밀접하게 결합하여 공동의 목표를 달성하기 위하여 협업을 수행하는 시스템이다. 한편, 단일 사이버-물리 시스템(Cyber-Physical System)의 경우에는 ISO 26262 또는 IEC 61508과 같은 표준을 따르거나 다양한 위험 분석 기법을 적용함으로써 그 안전을 확보할 수 있다. 그러나 CCPS에서는 협업을 수행중인 한 CPS의 결함으로 인하여 다른 협업 중인 CPS에게 수많은 결함을 발생시키기 때문에 안전의 확보가 매우 어렵다. 본 논문에서는 이러한 CCPS의 위험을 분석하여 안전을 확보하기 위해 복합적인 위험 분석과 위험 분석 산출물 사이의 관계를 기반으로 하는 위험 치명도 매트릭스(Fault Criticality Matrix, FCM)를 제시한다. FCM에서는 결함, 결함의 치명도, 안전 가드와 안전 가드의 발생 확률, 결함의 영향 및 순위를 나열하여 분석한다. 안전 엔지니어는 이를 통해 시스템의 설계 단계에서 각 결함의 치명도와 영향을 분석하고, 설계된 안전 가드를 통해 식별된 고장을 효과적으로 관리하고 제어함으로써 안전한 CPS를 개발할 수 있다. 제시된 방법의 유용성을 확인하기 위해 CCPS의 대표적 예인 군집주행에 대하여 사례 연구를 수행하였다. 본 연구에서 개발된 도구를 사용하여 군집주행 시스템에 FCM을 적용함으로써 상세한 결함 치명도 분석을 수행하였고, 분석 결과는 적합성과 효과성 관점에서 점검되었다. 또한 군집 주행에 대한 시뮬레이션 수행을 통해 FCM을 사용하여 결함 치명도를 분석한 군집주행 시스템이 발견된 모든 결함을 완화시켜 충돌 가능성을 크게 낮추었음을 보였다.

키워드 : 안전성, 통합 위협분석, 결함 치명도, 사이버물리시스템, 군집주행 시스템

1. Introduction

Collaborative CPSs are tightly coupled with cyber and physical components, massively interconnected, and collaborate by sharing information and resources to achieve common goals. The tight coupling of cyber and physical components has many advantages, but it also creates new challenges like safety issues

※ 이 논문은 과학기술정보통신부 연구재단에서 지원하는 중견연구과제의 지원에 의해 이루어짐(NRF-2020R1A2C1007571).
※ 이 논문은 한국정보처리학회 소프트웨어공학연구회가 주관한 2021 한국소프트웨어공학 학술대회(KCSE 2021)의 우수논문으로 "Safety Assurance in Collaborative Cyber-Physical System through Fault Criticality Analysis"의 제목으로 발표된 논문을 확장한 것임.

† 비 회 원 : 충북대학교 컴퓨터과학과 석·박사과정
†† 종신회원 : 충북대학교 소프트웨어학과 교수

Manuscript Received : April 20, 2021
First Revision : May 31, 2021
Accepted : June 18, 2021

* Corresponding Author : Jang-Eui Hong(jehong@chungbuk.ac.kr)

that have not been considered in the traditional computing domain. The development of smart and context-aware mission-critical systems has drastically risen in recent years. Power grids, water distribution networks, smarts factory, medical CPS, autonomous platoon driving systems are examples of complex cyber-physical systems. The safety assurance of these safety-critical systems becomes very important because any mishap in these systems can cause property loss, environmental damage, and even loss of human lives. As the collaborative CPS's components or subsystems are massively interconnected, their collaboration creates safety concerns because a minor fault in one system can activate many other faults in other collaborating systems.

In our previous work [1], we worked on fault traceability in collaborative CPSs, and we presented a Fault Traceability Graph (FTG) to visualize the impact of faults and their origin. The fault propagation graph also visualizes the fault route as a means of traceability. This traceability graph was based on content relationships among the hazard analysis artifacts. A Fault Traceability Matrix (FTM) was also introduced as a means of another representation of traceability of faults in CPSs. However, in the FTM, fault ranking and the effect of safety guards on a particular fault were not well presented. The criticality calculation of each fault and its rankings before and after supplying corrective actions was not considered. The probability of occurrence of faults and safety guard is the building block of our approach. In previous work [1], we did not present the probability of the occurrence of faults and safety guards. As collaborative CPSs are safety-critical and mission-critical systems, fault traceability is very important to determine the fault routes, their origin and to determine that the system must fulfill the safety goals, and all identified hazards were illuminated [2] or decreased their criticality to an acceptable level. In [3,4], the term traceability is defined by investigating the links between hazard analysis artifacts and design documents. However, the criticality analysis of faults in CPSs was not investigated. The propagation of the fault in collaborative CPSs is very critical for mission-critical systems. In a safety-critical system, even a minor fault cannot afford as it can activate many other faults in the collaborating systems. Therefore, a criticality analysis technique would be required to determine how much a specific fault is a threat to the safety of collaborative CPSs.

Hazard analysis helps safety engineers to identify faults and to resolve the insufficiency of safety- related functions, and some hazard analysis provides information about safety guards as mentioned in [5]. Fault Tree Analysis (FTA), Failure Mode Effect Analysis (FMEA), and Event Tree Analysis (ETA) are the most fundamental hazard analysis techniques used to perform the hazard analysis. The main objective of FTA is to determine the root causes of the specified fault. While on the other hand, the ETA is used to determine whether the safety guards sufficiently control an undesired event or not. However, the FMEA is used to evaluate the failure mode and effect of components, assemblies, and subsystems.

1.1 Motivation

The collaborative nature of CPSs creates safety challenges as CPSs operate in the physical environment. Therefore, fault mitigation methodologies must be considered at design time. As collaborative CPSs are very complex and heterogeneous, a single hazard analysis technique is insufficient to guarantee safety for such kinds of collaborative CPSs. Therefore, we proposed a composite hazard analysis technique in our previous work [6] based on content relationships among the hazard analysis techniques such as FTA, FMEA, and ETA. The composite hazard analysis technique can be used to analyze the hazard for complex systems like collaborative CPSs. Due to these properties such as massively collaboration of cyber-physical systems, fault criticality assessment is a challenging task. We assume that the proposed fault criticality matrix can enable the safety engineers to evaluate faults' criticality and their impact on other collaborating systems. It also enables the safety engineers to manage the fault and its propagation scope. The proposed Fault Criticality Matrix (FCM) ensures that the safety guard can sufficiently control the specified fault. As it is a basic principle of the cyber-physical system, all potential faults that may lead to undesired events must be illuminated to ensure the developed system is safe enough from harmful to humans and its environment.

Intending to present a criticality analysis method to determine the fault criticality in collaborative cyber-physical systems, we first perform the composite hazard analysis of collaborative CPSs by using our developed tool called CPSTracer. Then we generate the FCM for CCPS. By extending our previous work [6] in which we presented only a fault traceability graph, now present the fault criticality matrix to perform criticality analysis of faults in collaborative CPSs.

1.2 Contributions

In this paper, we make the following contributions:

- First, we propose FCM to analyze the criticality of faults that can occur during CPS collaboration. The proposed FCM can demonstrate the criticality of faults with their ranking. It also shows the criticality of faults before and after supplying safety guards. This clear representation of criticality analysis helps to ensure that the supplied safety guards are enough or not.
- The proposed FCM was implemented and integrated with our tool CPSTracer. So, the information about fault criticality can support safety engineers to perform the safety analysis process seamlessly.
- We present a case study on the Platoon driving system (a collaborative CPSs) and generate the FCM to perform the criticality analysis of faults associated with the platooning system.

2. Related Work

The authors in [7] proposed a framework called SafeTrace that can manage the traceability among the safety requirements, design, and safety analysis artifacts in the medical device. The authors investigated the links between the hazard analysis artifacts, requirement artifacts, and design artifacts to see whether the change of requirement or design may affect the safety in the system. More specifically, the authors present the trace links between the design documents and basic events in FTA and between the top event in FTA and the requirement document. However, the criticality analysis of hazard artifacts was not determined in this approach.

The authors proposed a framework to develop safe and secure adaptive collaborative systems with runtime safety guarantees [8]. This framework aimed to develop run-time behavioral models for collaborative adaptive distributed systems and hazard analysis techniques to support continuous safety and ensuring cybersecurity for assumptions made in the model. In this case, it is needed to design the behavioral models, and techniques to analyze and check the safety and cyber-security both at design time and run time. In this approach, the authors considered the platooning system as a case study to validate their approach. However, the fault impact analysis and analysis of criticality of faults were not considered. It also does not support composite hazard analysis as a complex collaborative system's safety can not be guaranteed by a single hazard analysis technique.

An approach was proposed by [9] provides safety in the platooning CPSs within the safeCOP project. The proposed approach aimed to analyze safety-critical systems that need to provide safety assurance. The authors further argued that safety in cooperative CPSs is a challenging task because safety can only be ensured if all the systems cooperate safely during their operations. Continuous safety assurance was handled through the run-time manager during the truck platooning use case. The runtime managers' functionality can fully utilize when the safety contracts based on the safety analysis of the local system and the overall cooperative system are specified. However, composite hazard analysis was not considered, nor the criticality and impact analysis was performed.

A tool called NuDE 2.0 was developed by the authors [10] for safety analysis and verification for safety-critical systems. However, this tool does not support collaboration among multiple CPSs.

The authors in [11] proposed an architecture-based approach to investigate the human-robot collaboration and its capabilities to ensure human safety in a production environment. This architecture considers the shared fenceless working space where robots, humans, and other moving objects may operate. This study aimed to explore the response time needed to detect the human and activate the safety signals to ensure human safety. The shorter the detection time, the more rapid collision avoidance strategies opted. While on the other hand, the longer the response time and activating safety signal, the more hazardous for human safety.

The authors in [12] proposed an approach to address the fault criticality and interaction in complex consumer communications based on fault tree analysis. In this study, the authors outlined the limitation of traditional FTA. The fields and manufacturing failure modes, mechanism, and their interactions are arranged into a unified rank system to determine the criticality. This study also presents a new failure tree to demonstrate the criticality of faults.

3. Proposed Approach: Collaborative Behavior and Composite Hazard Analysis of CPSs

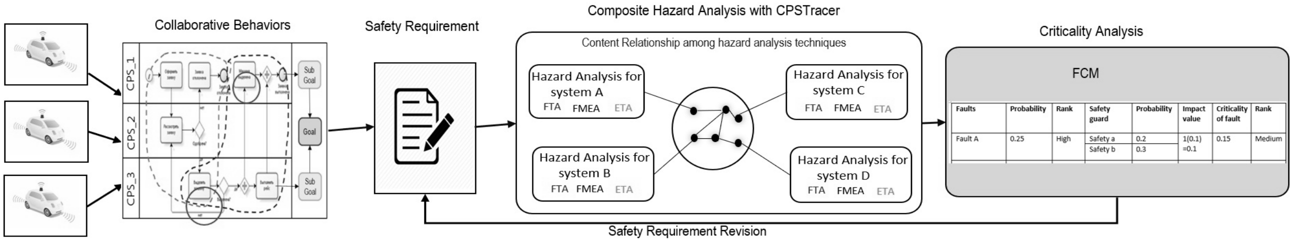
Collaborative CPSs are those systems that perform individual tasks to achieve a common goal. Due to the massively interconnected and heterogeneous subsystem, their collaboration creates safety concerns. However, hazard analysis in CPSs makes it possible for safety engineers to identify the potential hazards and provide the potential safety guards to mitigate the risk of fault from the system. Therefore, we proposed an approach to analyzing the safe operability of the collaborative system supporting the composite hazard analysis for CPSs as a single hazard analysis is not enough to guarantee safety for complex systems as shown in Fig. 1.

In our proposed approach, we first analyze the collaborative nature of the cyber-physical system. As in collaborative CPSs, a fault in one CPS can activate other faults in other CPSs to whom it may collaborate. It means that if any fault occurred in one CPS, it can propagate to other CPS collaborating with that CPSs. Therefore, based on our previous work [1], we used our defined content relationships among the hazard analysis techniques such as FTA, FMEA, and ETA to perform composite hazard analysis for collaborative CPSs. Another reason for this relationship is to envision the relationship among faults coming from different collaborating systems and the impact of specific faults on other systems. As we mentioned earlier that the failure of one collaborative CPS may affect other collaborating CPS. Therefore, to analyze the fault propagation and its routes, we introduced FTG in our previous work. However, the criticality analysis of faults was not considered at that time. Now, extending our previous work, we present a criticality matrix to perform criticality analysis in collaborative CPS. The proposed FCM ranks the criticality of faults based on the probability of occurrence of a fault and its impact on other faults. This means that how many other faults are influenced by a particular fault. There are two rankings in the FCM matrix, in the first ranking criteria of faults, the safety guards are not considered to show the actual criticality of fault without providing a safety guard. After applying the safety guards to the faults, we rank the fault again to check whether the supplied safety guards were enough to cope with the specified fault or not, as well as to check the effect of the safety guard on fault criticality. Definitely, after supplying some safety guards, the criticality of faults is reduced.

3.1 Collaborative Behavior of CPS

To analyze the collaborative behavior of CPSs, we use the platooning system as our case study to justify our approach. The platoon driving system forms a platoon of several vehicles maintaining a short inter-vehicle distance with the preceding vehicles to improve traffic flow, reducing traffic congestions, and reducing fuel consumption as mention in [13]. Fig. 2 shows an example of the platooning system.

In the platooning system, the head of the platoon is called the leader, and the following cars behind the leader are called followers. Cooperative Adaptive Cruise Control (CACC) is used to communicate with each other in the platooning system to create cooperation. Each vehicle in the platooning system


Collaborative Behaviors
CPS_1
CPS_2
CPS_3
Sub Goal
Goal
Sub Goal
Safety Requirement
Composite Hazard Analysis with CPSTracer
Content Relationship among hazard analysis techniques
Hazard Analysis for system A
FTA FMEA ETA
Hazard Analysis for system B
FTA FMEA ETA
Hazard Analysis for system C
FTA FMEA ETA
Hazard Analysis for system D
FTA FMEA ETA
Criticality Analysis
FCM
Safety Requirement Revision

Faults	Probability	Rank	Safety guard	Probability	Impact value	Criticality of fault	Rank
Fault A	0.25	High	Safety a	0.2	1(0.1) =0.1	0.15	Medium
			Safety b	0.3			

Fig. 1. A Proposed Approach for Safety Assurance through Fault Criticality Analysis in Collaborative CPS

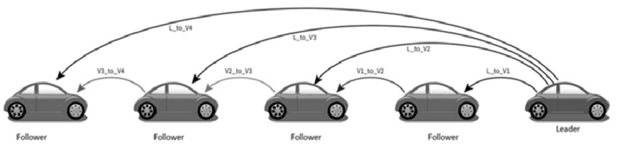


Fig. 2. Example of a Platooning System

is independent CPS and can use the Adaptive Cruise Control (ACC) unit when necessary. As in the platooning system, the distance among the vehicles is supposed to be very short; therefore, a fault in the leader vehicle can be propagated to the other vehicles, as a result, collision among the vehicles may occur. For example, if the leader vehicle sends a wrong command of acceleration to follower vehicles, then the collision will potentially occur. Even though a platooning system developed with safety requirements, it is not possible to consider all possible failure conditions in runtime.

3.2 Safety Requirements

The safety requirements are defined to reduce the risks in systems. These requirements are first specified at a high level, refined, and then supplied to the designer. Safety requirements are generated after analyzing the collaborative behavior of CPSs in our approach to performing composite hazard analysis on specified safety requirements. After extracting the safety requirements from the collaborative behavioral analysis of CPSs, we supply the safety requirement as input to our composite hazard analysis technique to perform hazard analysis. In the next step, we perform the criticality analysis of faults to verify whether identified faults are removed from the system or not. This process is a close loop, and it continues until all potential faults are mitigated from the system or an acceptable level of safety is achieved, and the safety requirements are also revised according to the status of the fault.

3.3 Composite Hazard Analysis of the Platooning System

Usually, in the platooning system, the vehicles form a group and collaborate to reduce traffic congestion by reducing the inter-vehicle distance. However, the shorter distance among the participating vehicles creates safety concerns for vehicles in the platooning system. We assume that the safety of CCPS can be ensured by performing the hazard analysis to explore potential faults in the system. For this purpose, we used our composite hazard analysis tool called CPSTracer to perform hazard analysis. The composite hazard analysis technique is based on four kinds of content relationships, influence, inheritance, supplement, and overlap relationships among the hazard analysis techniques such as FTA, FMEA, and ETA [1]. The definitions of relationships are as follows:

1) Influence Relationship

The influence relationship exists among the faults in participating CPSs in which a fault in one participating system causes the activation of another fault in CCPSs.

2) Inheritance Relationship

This relationship is established when two or more participating systems in CCPS share the same operational and functional constraints. The inheritance relationship also exists among the faults of the participating systems.

3) Overlap Relationship

This relationship exists when the consequences of the failure of one system are the same as the failure consequences of another system. This relationship exists among the outcomes/consequences of the failure of systems in collaborative CPSs.

4) Supplement Relationship

This relationship exists among the safety guards and faults in collaborative CPSs. When a system has a safety guard to cope with identical faults in another system, then this relationship is established between the safety guard and that particular fault. This means that the safety guard for a fault of a system can be supplied to another identical fault of a system in CCPs.

FTA is widely used for hazard and risk assessment in CPSs. In our approach, we first perform an FTA as part of composite hazard analysis to identify the root cause of failure of our case study i.e., the platooning system. We assume that one of the reasons for the platooning system failure is *Car Collision* (i.e., the top event in FTA). The top event in FTA is the failure of the system as a whole which means in the case of the platooning system, the participating vehicles do

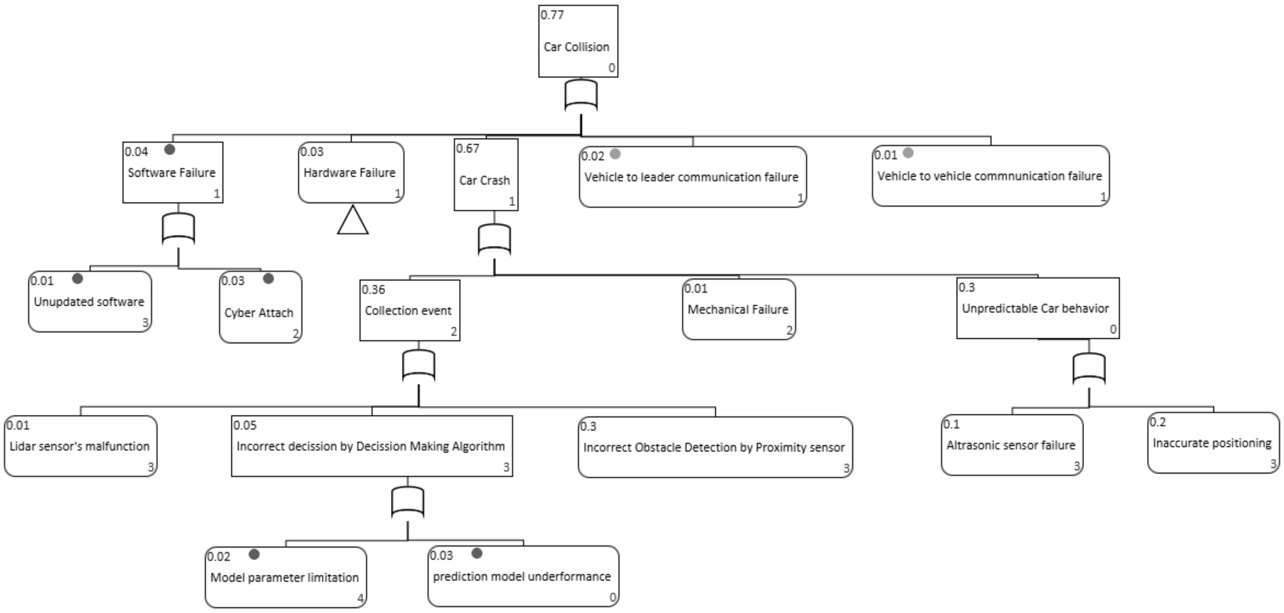
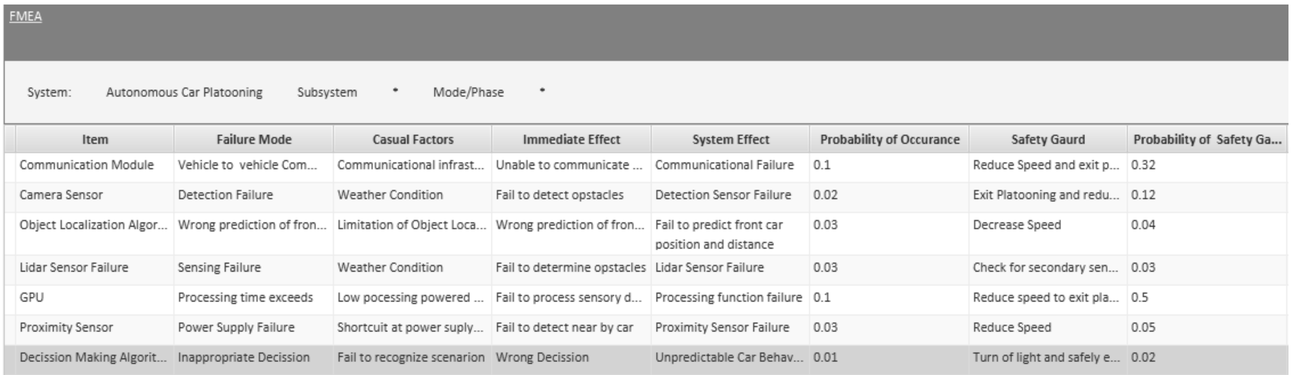

0.77
Car Collision
0
0.04
Software Failure
1
0.03
Hardware Failure
1
0.67
Car Crash
1
0.02
Vehicle to leader communication failure
1
0.01
Vehicle to vehicle commnunication failure
1
0.01
Unupdated software
3
0.03
Cyber Attach
2
0.36
Collection event
2
0.01
Mechanical Failure
2
0.3
Unpredictable Car behavior
0
0.01
Lidar sensor's malfunction
3
0.05
Incorrect decission by Decission Making Algorithm
3
0.3
Incorrect Obstacle Detection by Proximity sensor
3
0.1
Altrasonic sensor failure
3
0.2
Inaccurate positioning
3
0.02
Model parameter limitation
4
0.03
prediction model underformance
0

Fig. 3. Fault Tree Analysis of the Platooning System

not collaborate as a result, the collision has happened. An FTA of five levels for the platooning system is constructed as shown in Fig. 3. The intermediate and the basic events in FTA are root causes of the top event (i.e., *Car Collision*).

The failure mode and effect analysis is a structured method for system safety analysis to identify, evaluate, and score the potential failure for the system and its effect on the system. In our composite hazard analysis technique, we performed FMEA for the platooning system for its potential hazards. We perform FMEA on *Communication Module, Decision Making Algorithm, Object Localization Algorithm, Proximity Sensor, Camera Sensor, and Lidar Sensor*, etc. Fig. 4 shows the FMEA for the platooning system performed in our composite hazard analysis tool. We introduce new columns in FMEA for safety guards and their probability of occurrence and probability of occurrence of failure modes.

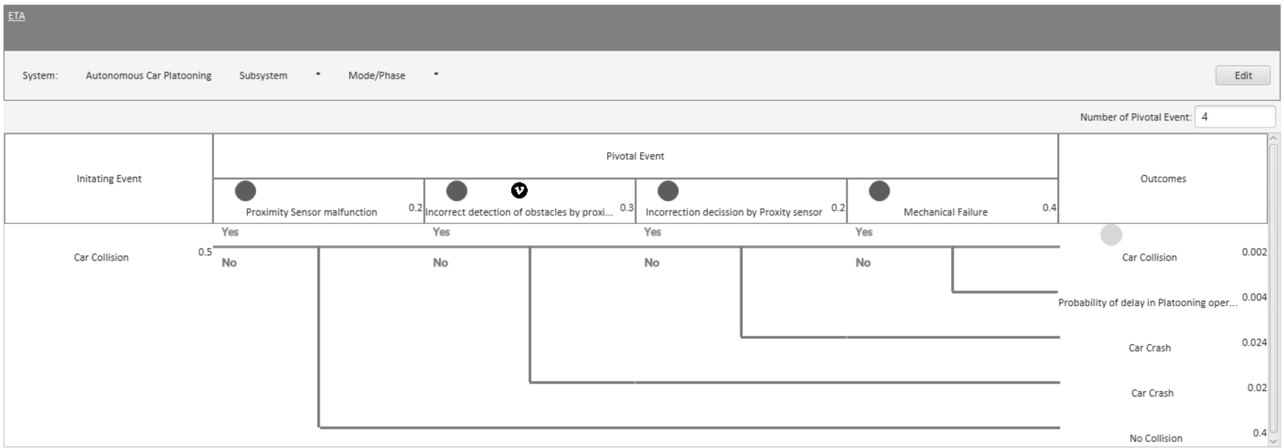
In the last step of our composite hazard analysis technique, we perform event tree analysis for the platooning system. ETA shows all potential outcomes stemming from the undesired event and takes into additional events and factors i.e., whether or not the installed safety barriers are working. ETA is used to identify the possible potential accident scenario and sequences in the complex system. Due to its efficacy in analyzing hazards in complex systems, we assume that it is good for collaborative CPSs as well. Fig. 5 shows the ETA for the platooning system.


FMEA

System: Autonomous Car Platooning Subsystem • Mode/Phase •

Item	Failure Mode	Casual Factors	Immediate Effect	System Effect	Probability of Occurance	Safety Gaurd	Probability of Safety Ga...
Communication Module	Vehicle to vehicle Com...	Communicational infrast...	Unable to communicate ...	Communicational Failure	0.1	Reduce Speed and exit p...	0.32
Camera Sensor	Detection Failure	Weather Condition	Fail to detect opstacles	Detection Sensor Failure	0.02	Exit Platooning and redu...	0.12
Object Localization Algor...	Wrong prediction of fron...	Limitation of Object Loca...	Wrong prediction of fron...	Fail to predict front car position and distance	0.03	Decrease Speed	0.04
Lidar Sensor Failure	Sensing Failure	Weather Condition	Fail to determine opstacles	Lidar Sensor Failure	0.03	Check for secondary sen...	0.03
GPU	Processing time exceeds	Low pocessing powered ...	Fail to process sensory d...	Processing function failure	0.1	Reduce speed to exit pla...	0.5
Proximity Sensor	Power Supply Failure	Shortcuit at power suply...	Fail to detect near by car	Proximity Sensor Failure	0.03	Reduce Speed	0.05
Decission Making Algorit...	Inappropriate Decission	Fail to recognize scenarion	Wrong Decission	Unpredictable Car Behav...	0.01	Turn of light and safely e...	0.02

Fig. 4. Failure Mode and Effect Analysis of the Platooning System


ETA

System: Autonomous Car Platooning Subsystem • Mode/Phase •

Number of Pivotal Event: 4

Edit
Initating Event
Pivotal Event
Outcomes
Proximity Sensor malfunction 0.2
Incorrect detection of obstacles by proxi... 0.3
Incorrection decission by Proxity sensor 0.2
Mechanical Failure 0.4
Car Collision 0.5
Yes
No
Yes
No
Yes
No
Yes
No
Car Collision 0.002
Probability of delay in Platooning oper... 0.004
Car Crash 0.024
Car Crash 0.02
No Collision 0.4

Fig. 5. Event Tree Analysis of the Platooning System

4. Fault Criticality Analysis

The collaborative cyber-physical systems are massively interconnected and heterogenous and collaborate with other CPSs to achieve complex tasks. Due to these properties of CCPS, more and more effective hazard analysis technique providing better traceability, impact analysis, and criticality analysis is required to mitigate all potential hazards from the systems. Determining fault criticality in the CPSs is a very challenging task especially in collaborative CPSs.

Therefore, a matrix called Fault Criticality Matrix is proposed to perform criticality analysis in CCPS with the following definition.

[Definition] Fault Criticality Matrix (FCM) is a tabular form that contains probability, criticality, and other data for a specific fault. The table consists with the tuple of <Fault, P. C, Rank, SG, P(SGs), IV, FC, Rank$_s$>, where each element is explained as follows and also shown in Table 1.

In the proposed FCM, we organize the identified faults in the first column i.e., the "*Fault*" column after performing the composite hazard analysis using our tool. In the second column "*P*", we arrange the probability of occurrence of each fault. In the third column ("*C*"), we calculate the criticality of fault without considering the safety guard to determine the actual fault criticality. The ranking of the faults is done based on the criticality of the fault which is organized in the column "*Rank*".

Table 1. Template for Fault Criticality Matrix

Fault	P	C	Rank	SG	P(SGs)	IV	FC	Rank$_s$
k	X_k	X_k+N	High	i	S_i	N	C	Medium
k	X_k	X_k+N	Medium	i	S_i	N	C	Low
k	X_k	X_k+N	Low	i	S_i	N	C	Negligible

P: Probability, C: Criticality, SG: Safety Guard, P(SGs): Probability of Occurrence of Safety Guards, IV: Impact value FC: Fault Criticality (after safety Guard), Rank$_s$: Rank after the safety guard

The safety guard column (i.e., "*SGs*") in our proposed FCM is consists of all safety guards provided to each fault and the probability of occurrence of safety guard is organized in column "*P(SGs)*". In column "*IV*", the impact value for each fault is placed. Impact value is the number of other potential faults that can be activated by a particular fault. This value is determined by the number of *influence*, *overlap*, and *inheritance* relationships that a fault makes with other faults. One influence relationship or one inheritance relationship or one overlap relationship with other faults have the impact value of 0.1.

For example, if a fault is making influence relationships with two other faults, then the impact value (*IV*) will be 0.2. Same as if a fault is making one inheritance relationship and two influence relationship, then the impact value (*IV*) will be 0.3. The second last column is "*FC*" which is the criticality of fault after supplying the safety guards. This column is introduced to show the effect of the safety guards on fault criticality. The "*Rank*" column ranks the criticality of faults after supplying a safety guard to show the effect of the safety guard on the criticality of each fault. Table 1 shows the template of the pro-

Table 2. Ranking Criteria of the Criticality of Faults

Fault Criticality Value (C)	Ranking
C <= 0.0	No effect
0.0~0.005	Negligible
0.005~0.01	Low
0.01~0.15	Medium
0.15~0.4	High
0.4~0.6	Very High
0.6~1.0	Catastrophic

posed fault criticality matrix. The ranking criteria for determining the criticality of faults are shown in Table 2.

4.1 Fault Criticality Calculation

The criticality of fault is represented by *FC*. Let us consider S (s_1, s_2, s_3, .., s_i) is the probability of safety guards where i=1, 2, 3... and X (x_1, x_2, x_3, ⋯, x_k) is the probability of faults where k=1, 2, 3,⋯ The safety guard S_i is supplied to mitigate the impact of fault X_k from the systems where i is the number of safety guards and k is the number of faults. The impact value of fault is represented by N. The value N is defined to be 0.1 if the fault has an impact on only one fault. If the faults impact multiple faults, then each impact would have a 0.1 value and the final impact value will be the sum of all values. The probability of occurrence of each fault, and safety guard is taken from the respective hazard analysis technique and the impact value is taken from the number of influence, overlap, and inheritance relationship that a fault is making with other faults. Now the value of fault criticality is determined by using equation (1).

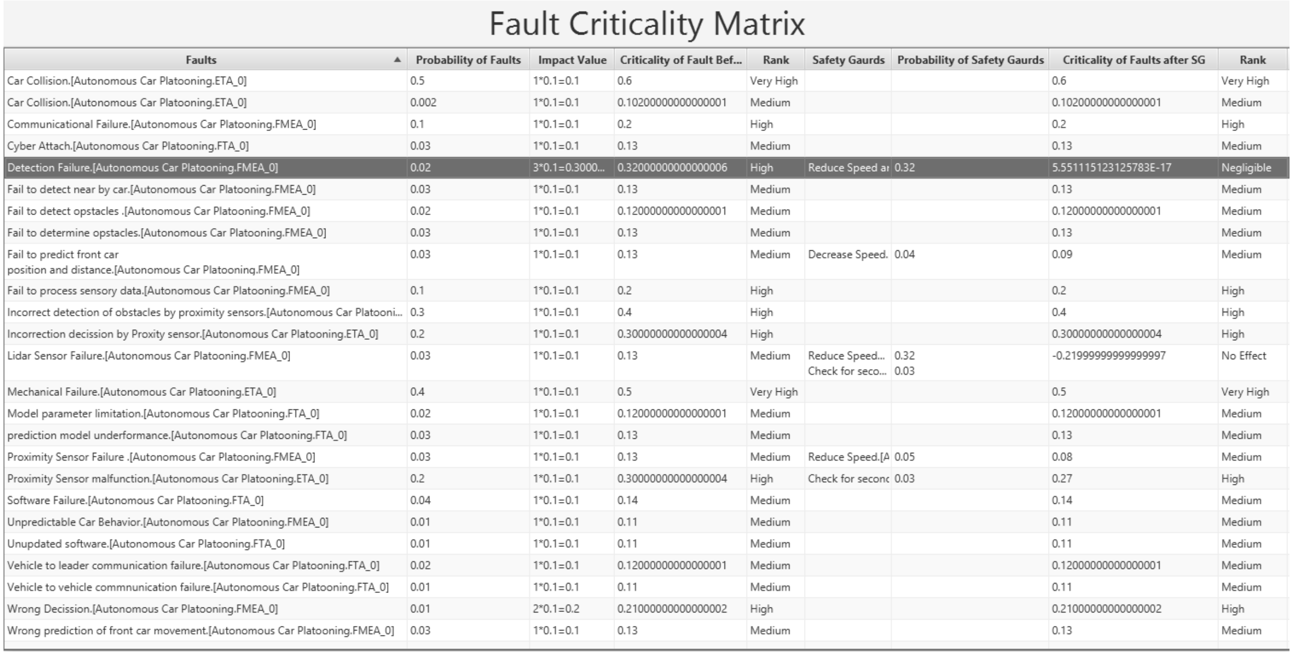
$$FC = (X_k + N)\forall k - \sum_{i \in 1} Si \qquad (1)$$

The minus or zero value of criticality of any fault indicates that the faults do not affect the system as the fault's criticality is mitigated by supplying the safety guards. If the criticality of fault still does not change after supplying the safety guard, then the supplied safety guard is not enough to mitigate the fault of the system. If the criticality of faults does not reduce, safety engineers must focus on such kinds of faults because these faults are more critical for system's safety. Reduction in criticality indicates that the criticality is reduced however, for the failure-free system, all potential faults must be mitigated or should reduce their criticality to zero. For this purpose, after applying the safety guard, safety verification must be done, and then revise the safety requirements. This process must continue until all safety goals are achieved.

4.2 Fault Criticality Analysis of the Platooning System

Cyber-physical systems are safety-critical. Renowned safety-critical CPSs, for example, self-driving cars, manufacturing plants, sensitive medical devices, and nuclear reactors, sometimes exhibit unsafe behaviors and which results in huge loss of properties and environmental damages. Addressing CPSs safety includes designing a safe cyber-physical system and verifying the CPS's safety until the desired safety goal is achieved. A term recently used for safe system designing is safety-guided designing which is used to designing the safe system and its safety verification. Hazard analysis is the crucial activity for safety-guided design which can be a challenging task for complex dynamics systems like collaborative CPSs. As the collaborative CPSs consist of complex, dynamic, and highly interconnected systems, determining the criticality of fault is also a challenging task. It is a very complicated task to determine which faults are more critical to systems safety and which have less criticality to breach the system safety. Therefore, we present the FCM to perform the criticality analysis of fault. This matrix is updated until the criticality of all faults is reduced to zero and the safety goals are achieved.

The platooning driving system is a collaborative CPS in which the leader and follower vehicles collaborate to reduce traffic congestion by maintaining a short inter-vehicle distance. However, the short distance between participating vehicles, relying on sensors creates safety concerns such that any fault in the participant vehicle can cause the whole platooning system collision. After analyzing the platooning system with our composite hazard analysis tool, we established content relationships among the hazard analysis (i.e., FTA, FMEA, and ETA) artifacts using our previously defined relationship [1], and we generated the Fault



Fault Criticality Matrix

Faults	Probability of Faults	Impact Value	Criticality of Fault Bef...	Rank	Safety Gaurds	Probability of Safety Gaurds	Criticality of Faults after SG	Rank
Car Collision.[Autonomous Car Platooning.ETA_0]	0.5	1*0.1=0.1	0.6	Very High			0.6	Very High
Car Collision.[Autonomous Car Platooning.ETA_0]	0.002	1*0.1=0.1	0.10200000000000001	Medium			0.10200000000000001	Medium
Communicational Failure.[Autonomous Car Platooning.FMEA_0]	0.1	1*0.1=0.1	0.2	High			0.2	High
Cyber Attach.[Autonomous Car Platooning.FTA_0]	0.03	1*0.1=0.1	0.13	Medium			0.13	Medium
Detection Failure.[Autonomous Car Platooning.FMEA_0]	0.02	3*0.1=0.3000...	0.32000000000000006	High	Reduce Speed ar	0.32	5.551115123125783E-17	Negligible
Fail to detect near by car.[Autonomous Car Platooning.FMEA_0]	0.03	1*0.1=0.1	0.13	Medium			0.13	Medium
Fail to detect opstacles .[Autonomous Car Platooning.FMEA_0]	0.02	1*0.1=0.1	0.12000000000000001	Medium			0.12000000000000001	Medium
Fail to determine opstacles.[Autonomous Car Platooning.FMEA_0]	0.03	1*0.1=0.1	0.13	Medium			0.13	Medium
Fail to predict front car position and distance.[Autonomous Car Platooning.FMEA_0]	0.03	1*0.1=0.1	0.13	Medium	Decrease Speed.	0.04	0.09	Medium
Fail to process sensory data.[Autonomous Car Platooning.FMEA_0]	0.1	1*0.1=0.1	0.2	High			0.2	High
Incorrect detection of obstacles by proximity sensors.[Autonomous Car Platooni...	0.3	1*0.1=0.1	0.4	High			0.4	High
Incorrection decission by Proxity sensor.[Autonomous Car Platooning.ETA_0]	0.2	1*0.1=0.1	0.30000000000000004	High			0.30000000000000004	High
Lidar Sensor Failure.[Autonomous Car Platooning.FMEA_0]	0.03	1*0.1=0.1	0.13	Medium	Reduce Speed... Check for seco...	0.32 0.03	-0.21999999999999997	No Effect
Mechanical Failure.[Autonomous Car Platooning.ETA_0]	0.4	1*0.1=0.1	0.5	Very High			0.5	Very High
Model parameter limitation.[Autonomous Car Platooning.FTA_0]	0.02	1*0.1=0.1	0.12000000000000001	Medium			0.12000000000000001	Medium
prediction model underformance.[Autonomous Car Platooning.FTA_0]	0.03	1*0.1=0.1	0.13	Medium			0.13	Medium
Proximity Sensor Failure .[Autonomous Car Platooning.FMEA_0]	0.03	1*0.1=0.1	0.13	Medium	Reduce Speed.[A	0.05	0.08	Medium
Proximity Sensor malfunction.[Autonomous Car Platooning.ETA_0]	0.2	1*0.1=0.1	0.30000000000000004	High	Check for seconc	0.03	0.27	High
Software Failure.[Autonomous Car Platooning.FTA_0]	0.04	1*0.1=0.1	0.14	Medium			0.14	Medium
Unpredictable Car Behavior.[Autonomous Car Platooning.FMEA_0]	0.01	1*0.1=0.1	0.11	Medium			0.11	Medium
Unupdated software.[Autonomous Car Platooning.FTA_0]	0.01	1*0.1=0.1	0.11	Medium			0.11	Medium
Vehicle to leader communication failure.[Autonomous Car Platooning.FTA_0]	0.02	1*0.1=0.1	0.12000000000000001	Medium			0.12000000000000001	Medium
Vehicle to vehicle commnunication failure.[Autonomous Car Platooning.FTA_0]	0.01	1*0.1=0.1	0.11	Medium			0.11	Medium
Wrong Decission.[Autonomous Car Platooning.FMEA_0]	0.01	2*0.1=0.2	0.21000000000000002	High			0.21000000000000002	High
Wrong prediction of front car movement.[Autonomous Car Platooning.FMEA_0]	0.03	1*0.1=0.1	0.13	Medium			0.13	Medium

Fig. 6. Fault Criticality Matrix for the Platooning System

Criticality Matrix to perform the criticality analysis of potential faults for the platooning systems as shown in Fig. 6.

Let us take the example of potential fault *Detection Failure.[Autonomous Car Platooning.FMEA_0]* which is discovered using FMEA as part of our composite hazard analysis technique. Its probability is 0.002 and criticality is 0.32 and it has an impact value of 0.3 which means that it is influencing three other faults in the platooning system. After calculating its criticality and before applying the safety guard, we can see that the criticality is "*High*" in the "*Rank*" column. However, after applying the safety guard "*Reduce Speed and exit platooning.[Autonomous Car Platooning.[FMEA_0]]*", we calculated the fault criticality again and the criticality of the fault is reduced to 5.5 $*10^{-17}$ as shown in the column "*Criticality of Fault After SG*". After supplying the safety guard, the criticality rank of the above-mentioned fault becomes "*Negligible*" in the last column that is the "*Rank*" column. This means that the supplied safety guard is enough to mitigate the fault *Detection Failure.[Autonomous Car Platooning System.FMEA_0]*. However, sometimes some faults are so critical that a single safety guard is not enough to mitigate them. They need more safety guards to mitigate or to reduce their criticality from the system. Let us take another example of such a fault that needs more than one safety guard to mitigate its criticality. The probability of fault Lidar Sensor Failure.*[Autonomous Car Platooning.[FMEA_0]]* is 0.03 and it is impacting one other fault due to which its criticality becomes 0.13. As the rank of fault *Lidar Sensor Failure.[Autonomous Car Platooning.[FMEA_0]]* is "*Medium*" before supplying the safety guards. However, a single safety guard was not enough to reduce its criticality, therefore, we supplied two safety guards first one is *Reduce Speed and exit platooning.[Autonomous Car Platooning.[FMEA_0]* and the second one is *Check for the secondary sensor.[Autonomous Car Platooning.[FMEA_0]]*. After supplying these safety guards, we calculated its criticality and the criticality of the fault was reduced to -0.219 and its criticality is now ranked as "*No Effect*" which means the fault has been mitigated and there is no chance of being activated.

However, some faults are very critical that its criticality does not change even after supplying the safety guards. These kinds of faults are dangerous for the system. The criticality of fault *Fails to predict front car position and distance.[Autonomous Car Platooning.[FMEA_0]]* is 0.13 and its rank is "*Medium*" before the safety guard is supplied. However, even after supplying the safety guard *Decrease Speed.[Autonomous Car Platooning.[FMEA_0]]* there was no significant decrease in the criticality of fault is detected as the criticality is still 0.09 which is still huge, and it falls under medium rank, and as we can see that the rank is still "*Medium*" even after the safety guard is supplied. This means that the supplied safety guard is not enough to cope with the fault *Fails to predict front car position and distance. [Autonomous Car Platooning.[FMEA_0]].* Therefore, safety engineers must give significant importance to such kinds of faults as these faults can be activated in the system at any time.

Some faults in the system do not have safety guards. Therefore, their criticality remains always constant and these kinds of faults are the most critical for system safety. Let us take the example of one such kind of fault and analyze its criticality. The fault *Incorrect decision by the Proximity sensor. [Autonomous Car Platooning.[ETA]]* was discovered by applying ETA and its criticality is 0.3 as there was no safety guard available for this fault. Therefore, both criticality ranking (*i.e., Before and after supplying safety guard.*) is the same which is "*High*". This means that this fault is very critical and can be activated at any time causing the whole system failure as its criticality is quite high. Such kinds of faults must be monitored by a safety engineer as these faults do not have any safety guard and could more safety-critical for CPSs like the platooning system.

4.3 Simulation and Evaluation

In order to evaluate the usefulness of our proposed FCM, we consider the following research questions:

RQ1: How to verify that the criticality of faults provided by the FCM is appropriate? This question is formulated to verify the appropriateness of the criticality of faults in the proposed FCM.

RQ2: How effective is the FCM in ensuring safety in safety-critical systems? This question is formulated to check whether proposed safety guards in FCM reduce the risk associated with the faults.

In response to the RQ1, we refer to the guidelines from ISO 26262 standard to verify the appropriateness of the faults and their criticality in our FCM. This is to check that the level of faults criticality in our approach is at a similar level compared to the ISO 26262 standard and [16]. ISO 26262 standard widely used in automotive industries to ensure safety. It defines the three safety attributes such as *severity, exposure, and controllability* to construct criticality levels of faults in safety-critical systems [15]. The authors [16] used the guidelines of ISO 26262 to perform the hazard analysis of the heavy-truck platooning. They used the ranking system of ISO 26262 to rank the criticality of the faults based on their *severity*, exposure, and controllability as shown in Table 3.

In ISO 26262, the severity refers to the extent of harm to an individual in a specific situation. The severity has four levels as mentioned below:

- S0: No injuries; S1: Light to moderate injuries; S2: Severe to life-threatening injuries; S3: Life-threatening to fatal injuries.

Exposure refers to the relative frequency of the operational conditions, in which hazardous events may occur and cause hazards and injuries. Its five levels are as below;

- E0: incredibly unlikely; E1: Very low probability of exposure; E2: Low probability of exposure; E3: Medium probability of exposure; E4: High probability of exposure.

Controllability refers to the prevention of specific damage through the timely reactions of the persons. The levels of controllability are given as follows.

- C0: Controllable in general; C1: Easily controllable; C2: Normally controllable; C3: Uncontrollable

Table 3. Example of Faults Considered in Hazard Analysis and Risk Assessment of Heavy-truck Platooning in [16]

Faults	Accidents	Severity	Exposure	Controllability
Communication failure	Crash	S3	E2	C2
Onboard sensors failure	Platooning system error	S2	E1	C3
Stopped vehicle on the road	Collision with vehicle	S3	E3	C2
Detection failure	Crash	S3	E1	C2
Weather (fog, rain, sleet)	Operation in a dangerous condition	S3	E2	C1

The faults in our FCM to perform criticality analysis of the platooning were carefully considered after following the guidelines from ISO 26262 and other scientific literature such as [14], and [16]. We considered the following faults from the platooning case study to see that the criticality ranking of faults in our FCM is at a similar level compared to the criticality ranking of faults in [16].

- *Communication failure, cyber-attack, detection failure, fail to detect a nearby car, fail to detect obstacles, fail to predict front car position and distance, fail to process sensory data, LIDAR sensor failure, mechanical failure, proximity sensor failure, and software failure, etc.*

As shown in Fig. 6. the FCM shows the criticality of faults before applying safety guards and after applying safety guards. We see that the criticality of faults in Fig. 6 are at a similar level compared to the criticality of faults in [16]. Although, we have ranked the faults into seven categories in the platoon driving system as shown in Table 2. However, the fault ranking falls under the general ranking of faults in ISO 26262. The reason to rank the faults in more categories in our FCM is to define more concrete controllability mechanisms according to the criticality value of faults.

The description of the faults in our FCM (Fig. 6) and the description of faults in existing literature may differ to some extent. However, the types and the class of faults are the same. All faults considered in the FCM are well-known faults associated with the platooning system. For example, the description of the fault "*detection failure*" in our FCM and the description of the fault "*onboard sensor failure*" in [16] is different. However, the type of fault and class of fault is the same and both faults belong to the detection system type with high criticality.

In order to verify the reduction of risk as seen in our FCM (Fig. 6), we used VEhicular NeTwork Open Simulator (VENTOS) [17] to analyze safety for the platooning system. The VENTOS is a closed-loop VANET simulator that combines the capabilities of both communication networks and vehicular traffic simulators. It is an open-sourced simulator that comprises Simulation of Urban Mobility (SUMO) and Objective Modular Network Testbed in C++ (OMNET++). It was designed for vehicular traffic follow analysis and can incorporate new control logic, such as intelligent traffic controller, collaborative driving (e.g., the platooning driving) dynamic routing, and autonomous driving capability.

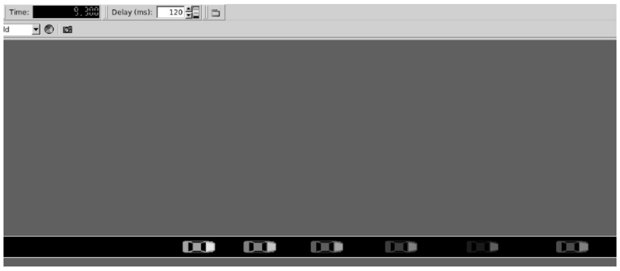


Fig. 7. A Platoon of Six Vehicles

We choose this simulator to verify the safe behavior of the platooning driving because it supports the platooning driving system. In our simulation setup, we first make a platoon of six vehicles (V1, V2....., V6) as shown in Fig. 7. The red-colored car is the leader of the platoon and blue color cars are followers in the platoon. We first present a scenario in which the Platooning system was not analyzed using FCM.

Scenario 1: The platoon was moving towards its destination under normal conditions. However, at some point, suddenly the leader faces a non-platooning car (i.e., the yellow car). The leader sends commands to the follower vehicles to reduce the speed. However, due to the narrow distance between the vehicle in the platoon and less reaction time to react the leader's command which results in a collision of the platoon. Fig. 8 shows the collision of the platoon in the VENTOS simulator.

Scenario 2: In the second scenario, we analyze the criticality of potential faults and make arrangements for safety guards if any critical fault is triggered

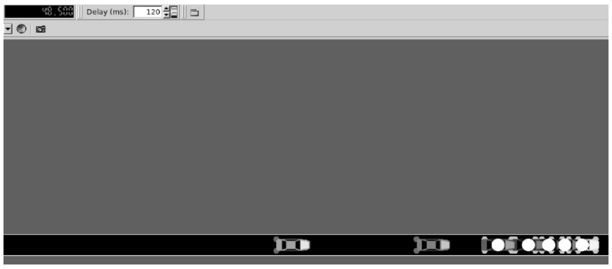


Fig. 8. Platooning Collision without Analyzing Criticality of Faults using FCM

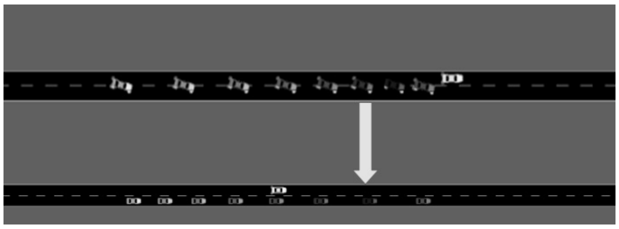


Fig. 9. Activation of the Safety Guards (i.e., Reduction in Speed and Safe Lane Change) in the Platooning Driving

among vehicles in the platoon. Hence, we perform the criticality analysis of faults using our FCM as shown in Fig. 6. This time we simulate a scenario in which we first perform the criticality analysis of faults of the platooning system then we simulate a platoon of 8 vehicles in the VENTOS simulator. The platoon was driving towards its destination under normal conditions. At some point, the leader received a message from Road Side Unit (RSU) about a traffic accident on the lane on which the platoon is driving. The leader transmits a command to activate safety guards such as "*reduce the speed*" and "*change the lane*". Hence, the followers receive the command from the leader and make safe lane change by reducing their speed. Fig. 9 shows the safe behavior of the platooning after activating safety guards discovered during fault criticality analysis.

In response to the research question RQ2, Fig. 6, 8, and 9 show the effectiveness of the FCM. As compared to the scenario in which the platooning driving's fault criticality was analyzed using FCM, it is safer than the scenario in which the fault criticality of the platooning driving was not analyzed with FCM.

It is cleared from the presented scenario 1 in which the platooning system was not analyzed using FCM faces a crash due to the "*detection failure*". In this scenario platoon leader suddenly faces a non-platooning vehicle, due to less reaction time the leader could not apply an emergency brake to stop the vehicle. Therefore, the platooning system crashed. Fig. 8 shows the hazardous scenario of the platooning system without analyzing the criticality of faults using the FCM.

In scenario 2 the platooning system was analyzed by the FCM. Hence, all the potential faults were considered during the criticality analysis and arranged safety guards to cope with the faults in the future. In this scenario, the leader received a message from RSU about a fault i.e., "*traffic accident*" on the lane on which the platoon is driving. Hence the platoon leader activates the predefined safety guards (i.e., *decreasing speed and safe lane change*) and sends the command to activate the safety guards for followers. The followers receive the command from the leader and made a safe lane change by decreasing the speed. Hence, the safety of the platooning in the hazardous scenario is ensured by activating the timely mannered safety guards. Fig. 9 shows the good impact of the FCM on the safety of the platooning in which the platoon system's safety was ensured by activating safety guards (i.e., *decreasing speed and safe lane change*).

To verify the high critical fault's impact on the safety of the platoon system, we present another scenario in which the platooning system faces a fault i.e., *Communication Failure*. The criticality of this fault is *0.2* which is ranked as "*High*" in the rank column in Fig. 6. As we can see that the severity of the *communication failure* is *S2* in the hazard analysis and risk assessment table (i.e., Table 3.) of the platooning system in [16]. The severity of this fault falls under the "*Severe to life-threatening injuries*" rank according to ISO 26262 [15]. Due to *communication failure* between the follower and leader vehicle, the follower vehicle made sudden acceleration, and hence the platoon crash has happened. As we can see the rank of the fault *communication failure* is high and the chance of crash was also higher, hence during the simulation the crash has happened which shows the correctness and effectiveness of calculated criticality of the fault *communication failure* in the FCM.

In the last scenario, we present another important potential fault i.e., *detection failure*. In this scenario, as the *detection failure* rank is *high* in Fig. 6, so there is a huge chance of the platooning collision. The severity of the fault *detection failure* is *S3* in Table 3 of this hazard analysis and risk assessment of the platooning system in [16]. This fault's ranking falls under the "*Life-threatening to fatal injuries*" rank according to ISO 26262. Such kinds of faults are very dangerous to the safety of the vehicle as well as for the safety of human lives. Therefore, we arranged three safety guards for this fault. The platooning was driving in normal condition, at some point, the system detects the decrease in vision system of the leader vehicle due to environmental condition (i.e., dense fog). The system detects high critical faults i.e., *detection failure*, and activates the safety guards. The platooning leader sends the commands to followers to activate safety guards such as *reduce speed, activate ACC mode, and dissolve platooning.* Hence the safety of each vehicle in the platooning increased by activating safety guards.

4.4 Threats to Validity

Internal Validity: Our proposed FCM is based on the four content relationships (influence, inheritance, overlap, and supplement relationship) among the hazard analysis (i.e., FTA, FMEA, and ETA) artifacts only. The first threat to internal validity is that we consider the above-mentioned four types of relationships based on the FTA, FMEA, and ETA in the composite hazard analysis of CCPS. There exists other hazards analysis techniques like System Theoretic Process Analysis (STPA). However, during our study, we found that the most commonly used techniques for hazard analysis are FTA, FMEA, and ETA. And it is enough to find potencial faults from the three techniques.

External Validity: We used the VENTOS simulator to verify the effectiveness of FCM in our case study. VENTOS simulator is only dedicated to the Platooning CPS. Therefore, if any other collaborative CPS's fault criticality is analyzed using FCM, we need to choose another simulator for safety verification. For example, if we analyze the criticality of faults associated with a smart manufacturing system using FCM, then we need to use a simulator related to smart manufacturing CPS. Therefore, this technique poses a threat in terms of the generalizability of the verification part of the approach. However, we minimize this threat by making the FCM part more generalize which is our main contribution. Any collaborative CPS's fault criticality can be analyzed by using our FCM. However, for particular CCPS, we need a separate simulator in order to verify the effectiveness of FCM.

5. Conclusion and Future Work

Collaborative Cyber-Physical Systems (CCPS) are those systems that contain tightly coupled physical and cyber components, massively interconnected subsystems, jointly collaborating and exchanging information and resources, and performing an individual task to achieve a common goal. The failure in one system in collaborative CPSs may lead to activating many other faults in participant systems. Safety assurance in CCPS is a thorny challenge for safety engineers and the determination of criticality of faults is also a challenging task especially in complex systems like collaborative CPSs.

Therefore, we proposed the Fault Criticality Matrix for fault criticality analysis based on the composite hazard analysis technique. This matrix presents the criticality of faults with an impact value of the fault. We analyze the hazards for the platooning system with our composite hazard analysis tool and perform the criticality analysis using FCM. After performing the repeated fault criticality analysis of the platooning system, our simulation results show that safety was assured.

As our project work is going on and in the future, we will work to develop a machine-learning algorithm to provide sustainable and resilient safety for CPSs in on-the-fly situations.

References

[1] D. Horn, N. Ali, and J. E. Hong, "Towards enhancement of fault traceability among multiple hazard analyses in cyber-physical systems," *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, Milwaukee, WI, USA, pp.458-464, 2019. doi: 10.1109/COMPSAC.2019.10249.

[2] J. Cleland-Huang, S. Rayadurgam, P. Mäder, and W. Schafer, "Software and systems traceability for safety-critical projects," Technique Report from Dagstuhl Seminar, 2015.

[3] M. Gleirscher and C. Carlan, "Arguing from hazard analysis in safety cases: A modular argument pattern," in *Proceedings of IEEE International Symposium on High Assurance Systems Engineering*, pp.53-60, 2017, doi: 10.1109/HASE.2017.15.

[4] A. Y. Z. Ou, M. Rahmaniheris, Y. Jiang, L. Sha, Z. Fu, and S. Ren, "SafeTrace: A safety-driven requirement traceability framework on device interaction hazards for MD PnP," in *Proceedings of 33rd Annual ACM Symposium on Applied Computing*, pp.1282-1291, 2018, doi: 10.1145/3167132.3167270.

[5] C. A. Ericsson, "Hazard Analysis Techniques for System Safety," Hoboken, NJ, USA: John Wiley & Sons, Inc., 2005.

[6] N. Ali, M. Hussain, and J.-E. Hong, "Analyzing safety of collaborative cyber-physical systems considering variability," in *IEEE Access*, Vol.8, pp.162701-162713, 2020. doi: 10.1109/ACCESS.2020.3021460.

[7] A. Y.-Z. Ou, M. Rahmaniheris, Y. Jiang, L. Sha, Z. Fu, and S. Ren, "SafeTrace: A safety-driven requirement traceability framework on device interaction hazards for MD PnP," in *Proceedings of 33rd Annual ACM Symposium on Applied Computing*, pp.1282-1291, 2018, doi: 10.1145/3167132.3167270.

[8] A. Čaušević, A. V. Papadopoulos, and M. Sirjani, "Towards a framework for safe and secure adaptive collaborative systems," in *Proceedings of International Computer Software and Applications Conference*, pp.165-170, 2019, doi: 10.1109/COMPSAC.2019.10201.

[9] S. Medawar, D. Scholle, and I. Sljivo, "Cooperative safety critical CPS platooning in SafeCOP," in *the 6th Mediterranean Conference on Embedded Computing (MECO)*, pp.1-5, 2017, doi: 10.1109/MECO.2017.7977210.

[10] E. S. Kim, D. A. Lee, S. Jung, J. Yoo, J. G. Choi, and J. S. Lee, "NuDE 2.0: A formal method-based software development, verification and safety analysis environment for digital I&Cs in NPPs," *Journal of Computing Science and Engineering*, Vol.11, No.1, pp.9-23, 2017, doi: 10.5626/JCSE.2017.11.1.9.

[11] N. Nikolakis, V. Maratos, and S. Makris, "A cyber physical system (CPS) approach for safe human-robot collaboration in a shared workplace," *Robotics and Computer-Integrated Manufacturing*, Vol.56, pp.233-243, Apr. 2019, doi: 10.1016/j.rcim.2018.10.003.

[12] D. E. Verbitsky and P. F. Lucent, "FTA technique addressing fault criticality and interactions in complex consumer communications," *Annual Reliability and Maintainability Symposium. 2001 Proceedings. International Symposium on Product Quality and Integrity (Cat. No.01CH37179)*, Philadelphia, PA, USA, pp.23-31, 2001.

[13] D. Jia, K. Lu, J. Wang, X. Zhang, and X. Shen, "A survey on platoon-based vehicular cyber-physical systems," *IEEE Communications Surveys and Tutorials*, Vol.18, No.1, pp.263-284, Jan. 2016, doi: 10.1109/COMST.2015.2410831.

[14] J. Reich, "Systematic engineering of safe open adaptive systems shown for truck platooning DEIS H2020 View project safeTbox: A model-based safety engineering tool View project," 2016. doi: 10.13140/RG.2.2.27809.61283.

[15] ISO, ISO 26262-2: Road Vehicles-Functional Safety, 2011.

[16] N. Highway Traffic Safety Administration, "Hazard Analysis of Concept Heavy-Truck Platooning Systems," United States, Department of Transportation, National Highway Traffic Safety Administration, May 2021. Accessed: May 25, 2021. [Online] Available: www.ntis.gov

[17] M. Amoozadeh, B. Ching, C. N. Chuah, D. Ghosal, and H. M. Zhang, "VENTOS: Vehicular network open simulator with hardware-in-the-loop support," in *Procedia Computer Science*, Vol.151, pp.61-68, Jan. 2019. doi: 10.1016/j.procs.2019.04.012.

Manzoor Hussain

https://orcid.org/0000-0002-9131-0930
e-mail : hussain@selab.cbnu.ac.kr

He received Bachelor degree at the Department of Software Engineering, The Univeristy of Azad Jammu and Kashmir Muzaffarabard, Pakistan in 2017 Currently, he is a graduate student in the Department of Computer Science Chungbuk National University, Korea sinece 2020. His research interests are in the area of Software Engineering, Deep Learning, Autonomous Car, and Cyber Physical Systems.

Nazakat Ali

https://orcid.org/0000-0002-3875-812X
e-mail : nazakatali@selab.cbnu.ac.kr

He received a Master degree and a Ph.D. degree at the Department of Computer Science, Chungbuk National University, Korea, in 2017 and 2021, repectively. Currently, he is working as a Postdoc in Software Engineering Laboratory, Department of Computer Science, Chungbuk National University. His research interests are in the area of Software requirements engineering, Ontology, software architecture, DevOps, and Cyber-Physical Systems.

Jang-Eui Hong

https://orcid.org/0000-0001-9786-7732
e-mail : jehong@chungbuk.ac.kr

He receivec a Ph.D degree at the Dapartment of Computer Science KAIST, 2001. He served as a researcher in Agency of Defense Development, Korea, and also worked for Solution-Link Co. as a Technical Director. Currently he is a professor in the Department of Computer Science, Chungbuk National University, Korea. His research interests are in the areas of Model-Based Software Engineering. Software Quality, Cyber-Physical Systems.