Method of Evaluating the Influence Factor of Safety in the Automated Driving System: The Chasm Between SAE Level 2 and Level 3

Masao Ito(✉)

NIL Software Corp., 2-17-7 Kinuta, Setagaya, Tokyo, Japan
nil@nil.co.jp

Abstract. Recently vehicle control system becomes have the automated feature. In this situation, the analysis based on malfunction of a system is not enough. We have to consider other hazard types such as the hazard originated from threats, the hazard that comes from the misinterpretation on using sensor. In this paper, we provide several hazard types that we have to think and explain the consistent approach to analyse the system in the concept phase.

Keywords: Safety · Security · ADS · ISO 26262 · J3016 · J3061
CARDION · SOTIF

1 Introduction

The embedded system of the automobile is improving every year especially from the viewpoint of safety, comfort, energy consumption and so on. Notably, the introduction of automated control, like Advanced Driver Assistance System (ADAS), provides the safe car to the users.

The progress of automated control means that the control of a car moves from the driver to the machine. This movement creates a new problem. Usually, the driver recognises the environment, making decision and control. The automated control implies to include the entire recognition and right decision. Those two of three parts (i.e. recognition and decision) is not deterministic ones because the environment with moving objects like other cars or pedestrians is very complicated and each object moves independently. It implies that we have to think the other elements to keep car safe besides the functional safety provided by the standard ISO 26262.

Currently, the central target of the automated driving system (ADS) becomes beyond the level 2 (partial driving automation) of SAE definition [1]. Between the level 2 and level 3 (conditional driving automation), there is an essential difference on the subtask of the dynamic driving task (DDT), that is who is responsible for the object and event detection and response (OEDR). In the level 2 the driver does the OEDR, but in the level 3, the system will do it instead.

It is difficult to sense object fully under the various circumstances, so the critical characteristic of OEDR is the probabilistic one even if we use the excellent approach to find the objects and events [7, 8]. This non-systematic feature may lead us to safety where the discussion of functional safety is hard to cover.

X. Larrucea et al. (Eds.): EuroSPI 2018, CCIS 896, pp. 161–172, 2018.
https://doi.org/10.1007/978-3-319-97925-0_13

There are several factors to consider other than functional safety when we examine safety. In this paper, we discuss two issues. One is security and the other is multiple ADAS subsystem.

If security problems occur when the system is in charge of control, there might introduce an accident. As for automobile embedded system, there is the cyber security guidebook, SAE J3061 [2]. This guidebook provides the process that is comparable to ISO 26262. So, we can easily understand it if we know the ISO 26262 standard. But both don't have the dedicated concrete method.

Secondly, we have to think the multiple ADAS subsystem. It may cause safety concerns, particularly when we develop them at different timings. To depict the all driving scenes is hard work, and it becomes more difficult for consolidating the behaviour of the multiple ADAS subsystems.

We summarize the influence factors for safety in Table 1.

Table 1. Influence factors

Influence factor	Safety category	Standard
Malfunction	Functional safety	ISO 26262 [5]
Misinterpretation of sensor signals	Safety of the intended functionality (SOTIF)	ISO PAS 21448 [18]
Attack	Automotive cybersecurity	ISO/SAE AWI 21434, SAE J3061 [2]
Decision of ADS	Safety of the decision	N/A

In the survey [19], it says there are three major activities in the safety-relating standardisation. That is the top three records in Table 1. It does not have the last record: "Decision of ADS". The cover of SOTIF is not currently clear, but it is relating to the detection part of OEDR. The 'R' of OEDR is the response to the detection and recognition of objects and events. Also, the system decides to response the recognised them. Currently, there is no existing or planned standard for the decision of ADS.

In this paper, we take account of the four influence factors in Table 1. Note that we do not consider whole security matter. We only think the cybersecurity relating to safety. As for the lifecycle phase, we will focus on the concept phase of system development and explain how to solve safety-related factors based on activities that the ISO 26262 Part 3 says. In Sect. 2 we explain issues we would like to answer. Section 3 describes our approach. Finally, we will summarise them.

We use the following terms. The term 'automated driving system (ADS)' includes 'driver assistance system (DAS)' and 'advanced driver assistance system (ADAS)'. The distinction between these two follows the definition in the eSafety project [3]. The general driving support system is DAS, and if it includes the environmental recognition and judgment, i.e. OEDR, it is called ADAS. We use ADS when referring to both, or when it is hard to say 'driver' support.

2 Issues

2.1 Hazards Coming from Threats

We concentrate on safety, so intrusion for privacy information and physical theft by key unauthorized access are outside the scope of this paper. We first try to find hazards and then find the threats that are relating to hazards. For example, in the adaptive cruise control (ACC) system, we consider a hazard that the time to collision (TTC) with the forward car cannot be maintained properly. This hazard might be caused by the failure of the equipment and the delay of detection of a failure. At the same time, we also think that there may be a security breach to falsify the TTC calculation.

In this paper, we call the hazard caused by the threat H_{THR}. And the normal hazard that comes from the malfunction of the system is called H_{MAL}.

2.2 ADS: Level 2 and Level 3 and Above

First, we consider an example of the car that has multiple ADAS subsystems. From this example, we will propose the principal in ADS of Level 3 or higher.

Example of Multiple ADAS Operation Scenarios
Let's consider an example where multiple ADSs operate (Fig. 1)[1]. It is an example combining the control in the longitudinal direction and the control in the lateral direction. This scenario has modified the case in NHTSA's report [4]. This report shows research results on appropriate HMI when multiple ADSs are used.

The scenario here is as follows: (1) the self-vehicle S follows the forward vehicle F by ACC. Since the preceding vehicle F finds an obstacle O ahead and changes the lane. (2) The driver of the self-vehicle S visually observes that there is an obstacle and at the same time the collision warning and braking (CWB) enacts and know the distance is too short to avoid the collision. (3) The driver of the self-vehicle S intends to change the lane and the active lane change with the assistance of Active Lane Change Assistance (ALCA). However, as the high-speed vehicle X is approaching from the left rear side, so ALCA steers to original lane with the warning to avoid collision with vehicle X. This situation is an ambivalent one if the only escape path is an acceleration to go in front of the vehicle X.

At level 2, this relates to the Human-Machine Interface (HMI) issue how effectively the system notifies the situation to the driver. With multiple warnings from the CWB and ALCA, it is essential to avoid the confusion of the driver. We already described this type of issue with the controllability of the driver in the paper [14]. We define this type of hazard that is relating to HMI as H_{HMI}.

At level 3, there is another problem, since the system is responsible for OEDR. That is, we cannot expect the entirely correct recognition of OEDR. Of course, the driver cannot always identify the object and make a correct judgement, either. But he is responsible for that. If the system is in charge of OEDR, it is necessary to grasp the degree of how OEDR can correctly do its work for keeping the system safe because the

[1] The issue shown in this example is categorized in the last one of Table 1 (i.e. "decision of ADS").

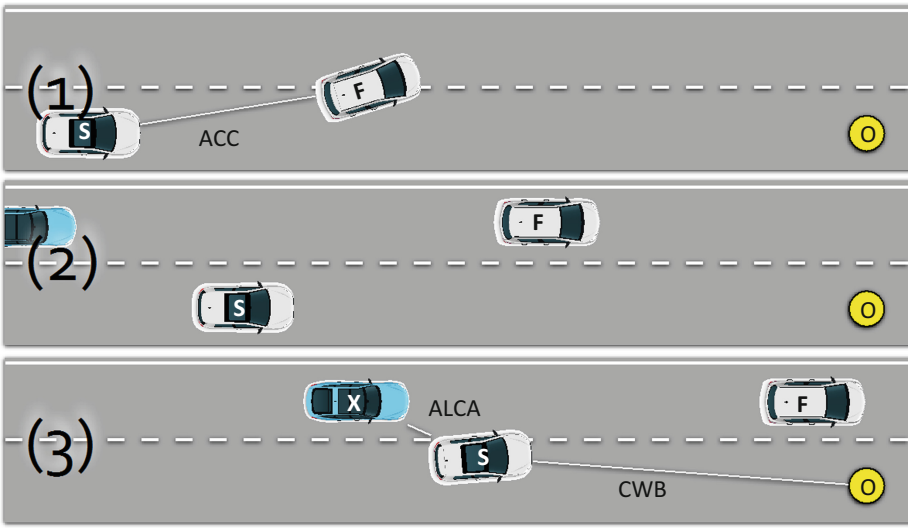

Fig. 1. A scenario shows the difficult situation of the car with multiple ADS systems.

system in charge of judgement. Let the hazard associated with this type relating to OEDR be H_{OEDR}. We describe the principle of ADS in the next clause.

ADS Principles

Substitution Principle

ADS should have the freedom of detection and the freedom of manoeuvre equal to or higher than that of a human being. The degree of detection freedom is the range and precision of detection of the object under the current environment. The degree of manoeuvre freedom refers to longitudinal and lateral available manoeuvring that can be taken in the current vehicle state.

Route Selection Principle

The system should select only the route with sufficient confidence to avoid danger under a current vehicle condition, environmental situation and both degrees of freedom.

At the level 3 or higher, the substitution principle can be considered a reasonable principle, because the machine performs environmental recognition (including other vehicles, etc.) instead of human beings. Route selection principles are principles for clarifying the responsibility of ADS.

Those principles are applied as follows. For example, the self-vehicle runs along the lane, and the nearby lane is opposing one. At this time, the vehicle travels within the range of degree of detection freedom. If there is a relationship between the preceding vehicle and the following vehicle, acceleration/deceleration is performed within the range of manoeuver freedom. If the oncoming car enters the lane of interest, it searches for an appropriate avoidance route. In the above example of multiple ADSs, the system might select the route when the path changing the lanes with acceleration operation has sufficient reliability under the route selection principle.

2.3 Relationship with ISO 26262

ISO 26262 aims at ensuring safety against hazards caused by malfunction of the system. Of course, malfunction is the problem for us to always consider. However, even when the function is working correctly, safety may not be maintained. In other words, we need to think about safety, even where the original ISO 26262 doesn't cover. The hazard H_{HMI} and H_{OEDR} cited above are examples. The latter is a case where there is a gap between demand and realization in the first place if the safety goal says that the system always recognizes target correctly. The realization concerning the recognition of the object itself has a danger, and the software also requires stochastic treatment similar to hardware ("In this case, the consequences are comparable to those of HW and SW failures and may also be safety critical." [7]). Of course, ISO 26262 is a comprehensive approach. We think of a method that can treat other hazards type other than the malfunction of the system.

According to the above discussion, there are two points we have to discuss.

As ISO 26262 covers a single item, we have the special care to the system that has multiple ADSs. For example, consider the previous ALCA as an item in the previous scenario (Fig. 3). ALCA performs control of lateral movement and longitudinal braking. According to the standard, we do the 'item definition (3–5)[2]', 'start the safety lifecycle (3–6)', and 'hazard analysis and risk assessment (HARA, 3–7)'. Suppose that there is a combination with CWB and ACC that are also categories of ADAS. We have to consider those items as the tightly relating subsystems. In the standard, we do this as part of the item boundary description in the item definition stage (3–5.4.2). However, it does not explicitly appear in hazard analysis and risk assessment.

When considering the hazards relating to braking, the braking control of ALCA is different from the control in CWB and ACC. We will think the hazard event that 'unintended braking occurs' in the HARA of item ALCA. For this hazard event, we assess the risk of the target item of ALCA. But, of course, we think the effect of other items: CWB and ACC. It is necessary to perform HARA individually and also including mutual interference. That is, we need to consider not only the influence on other items of ALCA but also whether hazards in ALCA will affect other items that has the braking control [8]. Of course, it is also necessary to consider the influence on the item ALCA from other existing items. In other words, if you have a connection in terms of function, you need to think about the effect mutually.

The second point is that ADS will replace the human recognition function (especially at the level of automation of Level 3 or higher). If this is right, the software safety of ADS truly comes from the consideration of "systematic failure". Regarding faults, as well known, it is treated as the systematic failure concerning software. And regarding hardware, we handle it stochastically [5]. On the other hand, as we saw in the previous principle when machines substitute for humans, the system has to process the information from many sensors under the complicated conditions of the surrounding environment. The judgment there always includes stochastic elements.

---

[2] We use the pointing system to refer a description: P-CH.SC.CL, where P is the part number, CH is a chapter number, SC is a section number and CL is a clause number of ISO 26262.

For example, when recognizing a forward vehicle, whether it is an image or a point group by LIDAR, there is no definitive correct answer. Depending on a certain threshold, it is a forward vehicle or false image unrelated to it [6]. For the recognition of the environment including the target; the system always has to calculate the result stochastically. If the safety goal is systematic only, there will be a gap called 'functional insufficiency' between the safety target, and the design and implementation [7, 8]. We can describe the goal declaratively, but we have to write design and coding procedurally and include some probabilities.

3 Method

We have proposed a method to analysis safety and security in the conceptual phase by the approach called CARDION [9]. We extend this method and deal with the problem mentioned in the previous section.

3.1 CARDION Method

It is a method to support the conceptual stage of ISO 26262. It corresponds to Part 3 of ISO 26262. The difficulty in the conceptual phase is that it is hard to use the safety and security methods normally used in design and implementation. If the structure is clear, Fault Tree Analysis (FTA) or Failure Mode and Effect Analysis (FMEA) can be used, However, for an item which is an abstract representation of the system, its structure and failure mode is not clear, so apply It is difficult. Nevertheless, it is important to analyse intrinsic safety and security by examining hazards and threats at an early stage. Therefore, a method suitable for the concept stage is required. Regarding security, the process of SAE J3061 has the almost same structure of ISO 26262, as it is "(t)he process framework described in this document is analogous to the process framework described in ISO 26262" ([2], p. 6).

The central part of the CARDION method is the next process. Usually, it repeats several times.

- Create an item sketch.
- Create a goal model for the item and simultaneously refine the item sketch.
- Apply the guideword to each goal description.
- Use item sketches to identify hazards and threats of the item.
- Establish safety and security goals and describe the (safety and security) requirements.

First, using the item sketch, we describe the target item according to ISO 26262 3-5.4.11 (function/non-functional of the item and description of the boundary). The item sketch includes static representation Static representation can be given using schema (e.g. UML class diagram, internal block diagram of SysML [10], or specification type representation of CATALYSIS approach [11]).

A part of the example in the ACC is shown in the lower left of Fig. 2 and the left in Fig. 3.

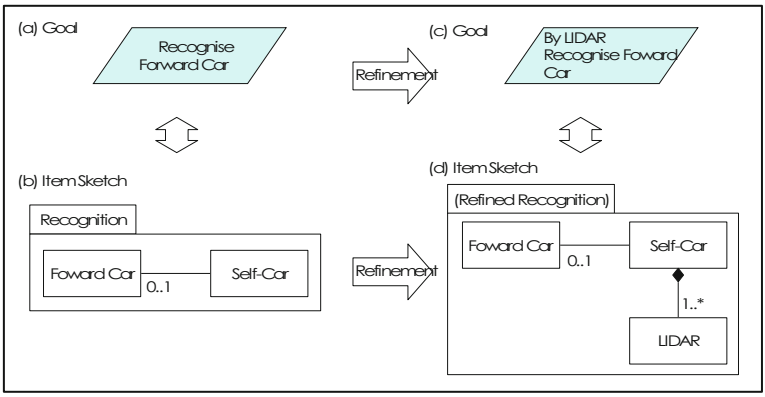

Fig. 2. Item sketch (structure)

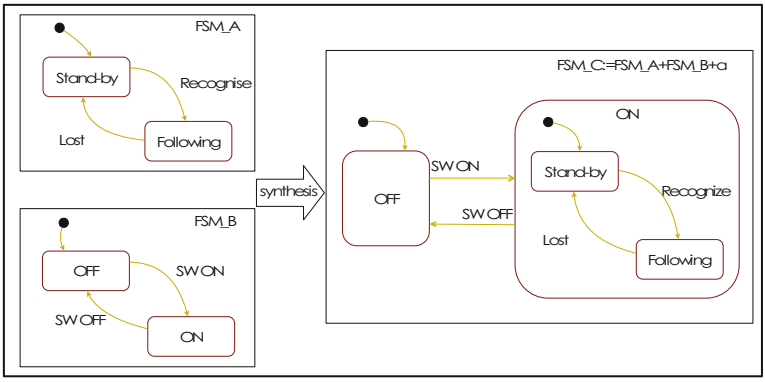

Fig. 3. Item sketch (behaviour)

The goal model of the item corresponds to the function/non-functional requirements that the item should have. As the base modelling method, we use the goal model of the KAOS approach [12] (see Fig. 4), but our approach does not rely only on the KAOS method. Any method can be used as long as we can clearly describe the distinction between the functional requirement and the non-functional requirement, and the relation between the requirements (for example, refinement relationship). Anyway, it is important to describe requests themselves for an item as natural language (NL) sentences. At this stage, natural language description is more suitable than formal description [15]. Moreover, in our method, it is necessary to describe requirements as an NL sentence to manipulate them to be described later. Herein, each request sentence is called a goal description sentence. Also, if necessary, we create new versions by refining or synthesizing the current item sketches. In Fig. 2, the lower right shows the refinement process from the lower left. The right finite state machine can be obtained by synthesizing the two finite state machines shown on the left of Fig. 3.

Next, we apply the guideword of HAZOP [13] to the previous goal description sentence and obtain a hazard candidate. The guidewords are following: NO or NOT (negation), MORE (quantitative increase), LESS (quantitative decrease), AS WELL AS (qualitative increase), PART OF (qualitative decrease), REVERSE (logically inverse), OTHER THAN (fully substitution), EARLY/LATE (clock time) and BEFORE/AFTER (order or sequence).

If the goal descriptive sentence is "system recognizes a forward car" and we apply the guideword NOT, the new sentence is "system does not recognise the preceding vehicle". And this new one is the hazard candidate. Alternatively, by applying LATE, we can obtain "system late recognise the forward car."

The advantage of this operation is exhaustive one by using guidewords extending time and space.

Elements that impede the achievement of the goal are called obstacle nodes in KAOS. Here, we use the four hazard types mentioned up to now: hazard H_{HMI} for HMI, hazard H_{OEDR} for OEDR, and hazard H_{MAL} for failure and H_{THR} for security. We use them to refine the obstacle node as the sub-obstacle node. Then we can select hazard events after getting the sub-obstacle nodes. For example, a sentence applying "OTHER THAN" to "obtain information on a forward vehicle by communication" is "getting the false information on the forward vehicle". As for the H_{MAL}, we can get the malfunction based hazard candidate. And from the viewpoint of H_{THR}, we can get one originated by the communication tampering.

Also, we can use the static item sketch mentioned to find threats. There are various ways to find threats as the threat analysis and risk assessment (TARA), and the appendix of [2] shows several methods as a reference for TARA. But we use another approach in the field of the control system, ISA/IEC 62443 (Security for Industrial Automation and Control Systems)[3] [16]. This standard has a zone/conduit model and defines a security policy. The zone is 'grouping of logical or physical assets', and the conduit is 'logical grouping of communication assets that protect the security of the channels it contains'. The static item sketch created in the conceptual stage includes an abstract structure and data flow. So, we set the zone for resources to be protected and check the data flow through the route among zones. The infringement and unauthorized access to the data that flow through conduit and data stored in the zone is a threat. For example, in the cooperative adaptive cruise control (CACC), the other connected car information should store in a security zone because it regulates the behaviour of the own vehicle, and check authentication of the data flowing through the conduit to/from the security zone.

The safety goal uses the goal description sentence, to which we've already applied the guideword, again. For example, as a result of HARA, it is assumed that "recognizing a forward vehicle is delayed and conflicts" is recognized as a hazard event for securing safety. The safety goal is to invert the meaning and not to be delayed in recognition: "being not in collision with the vehicle ahead due to recognition delay of the forward vehicle". In the safety requirement, for example, the maximum delay time

[3] We use the SAE J3061 to analyse threats, and we think ISA/IEC 62443 is useful to find the safety relating threats.

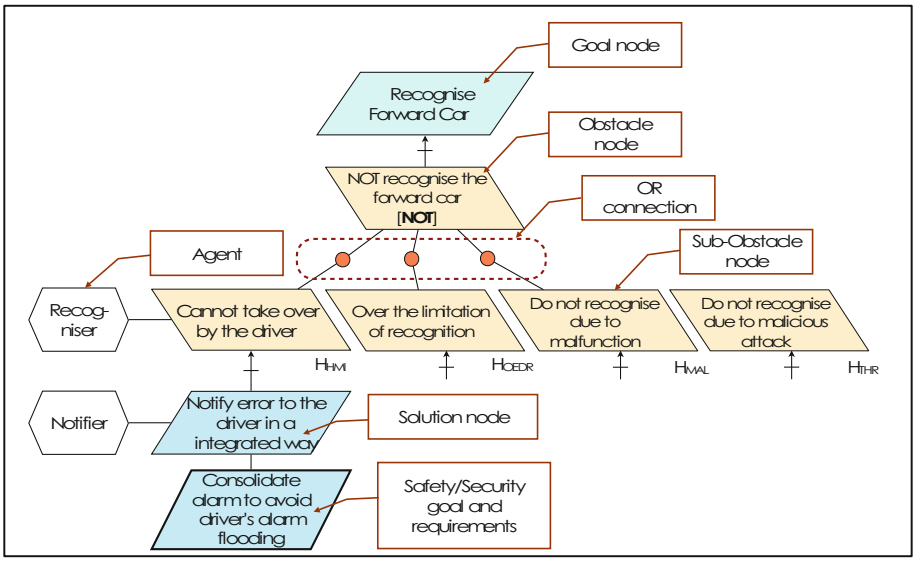

Fig. 4. Goal model and four hazard types

will be specified. It is also necessary to describe the initial architecture at this stage. We base on detailed item sketches and associate them with safety requirements.

The refinement is the top-down direction. We can traverse reversely to make sure whether the refinement is valid. In ISO 26262, we have to create the safety case to assure the validation. The GSN [20] is a candidate for writing safety case. If we can provide the arguments with evidence, we can satisfy the requirement of ISO 26262 (e.g. [21]).

Using the goal model, we can link a series of elements at the concept phase within one model diagram. The goal node indicates a function/non-functional requirement of the item, and the obstacle node is associated with a hazard belonging to any one of the four types. The refined solution nodes can show the safety and security goals. And, since the relationship between the elements in an item can be specified clearly, it is easy to judge the influence of different items.

3.2 Extension of the Approach

In this section, we extend our approach for the situation where multiple systems coexist (2.3.2). There are several points that we have to consider. And in the second half of this section, we will think about the preliminary architecture.

In a single item, it is not enough to identify hazard events, assess ASIL and consider the necessary safety mechanisms. Before defining safety and security requirements, we have to consider the influence of other items. ISO 26262 says in 3–5.4.2 that there is “interactions of the item with other items or elements” in the item definition stage, but we have the mutual influence among items during the concept

phase: finding hazard events stage, risk assessment stage and the stage of setting initial architecture. And we might modify the safety/security goals of other items.

The second point is the recognition of environment. If humans mainly recognise the surroundings, the human has the responsibility. When the system is in charge of OEDR, it is necessary to confirm the degree of recognition. For ensuring safety, self-diagnosis of the system is usually carried out. If the system is in an incorrect state, the system detects the fault and moves to the safety state. There are some important time intervals, the fault reaction time and fault tolerant time interval (FTTI). This interval is for H_{MAL}. And we need the extension for other hazard types.

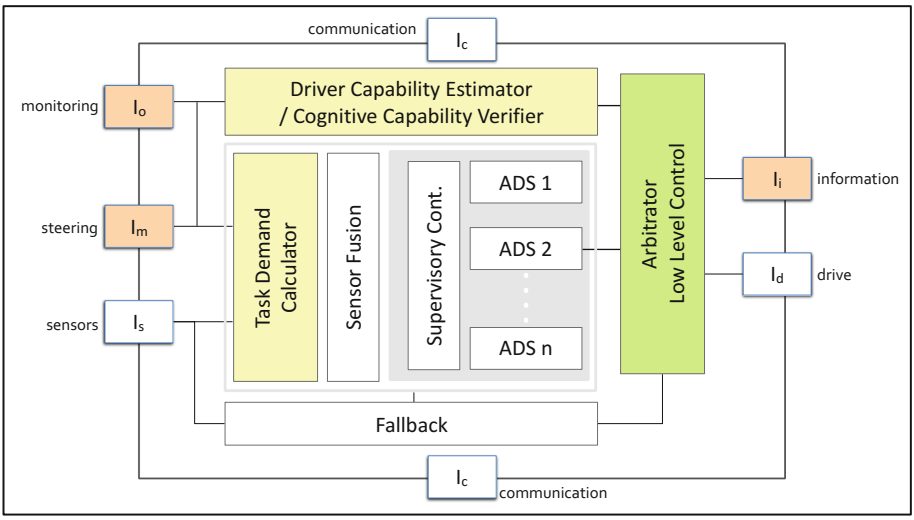

Fig. 5. A base of preliminary architecture

Next, self-diagnosis just closed in the system itself is inadequate. The system recognises the environment and makes a judgment in a stochastic way. So, we need a new type of self-diagnosis for ADS. For example, a recognition module of a system is confirmed the degree of recognition of the moving environment by other modules. When a system does same human activity as OEDR, the system needs such a new self-diagnostic function against changing the environment.

Now we can think about the preliminary architecture. We use the DESH-G (Driver, Environment, Software, Hardware and Goal) model of [14] here. DESH-G has the basic interface types is convenient when considering the control system of a vehicle. The interface types are I_o (interface monitoring a driver), I_m (driver's manoeuvring interface), I_i (interface for information display to a driver), I_s (sensor interface for environment recognition) I_d (interface controlling the actuator) and I_c (communication interface for V2 V and V2I). And also we use the item sketch that we already discussed. Figure 5 shows the base of a preliminary architecture.

We have several options for the recognition and a judgement composition. Figure 5 shows one of them. Synthesis of sensor information (Sensor Fusion) is performed first for multiple ADSs. A calculation results are passed to the Supervisory Controller (SV), and each ADS works from the information from SV. The arbitrator mediates the output of each ADS. In the automation level 3, we need a mechanism for control if a problem occurs (fallback) [1]. Though the fallback is the responsibility of the driver, ADS has to shift the system to the safe state until user ready for control.

Of course, we can consider the different structure for the higher automated level by using motion planner for finding a path and the trajectory control to control the vehicle according to the path. In this case, the item oriented approach of ISO 26262 might not be suitable.

4 Conclusion

In the ADS field, there are several hazard types besides malfunction oriented hazard: H_{THR}, H_{HMI} and H_{OEDR}. To make the vehicle safer, we have to consider those hazard types (The recent introduction of machine learning in the automobile field is another example [17]). We provided the consistent CARDION method for the concept phase. In this paper, we extended this method and finally showed the preliminary architecture. Between the automated level 2 and 3, there is a chasm. In level 3, the system is responsible for OEDR, not human (level 1,2). From the viewpoint of transition of the level, the system has to have both functionalities for OEDR, so the system becomes complicated. We believe the consistent approach is useful in this situation.

References

1. SAE: J3016: SAE international taxonomy and definitions for terms related to on-road motor vehicle automated driving systems. Levels of driving automation (2016)
2. SAE Vehicle Electrical System Security Committee, et al.: SAE J3061-Cybersecurity Guidebook for Cyber-Physical Automotive Systems. SAE-Society of Automotive Engineers (2016)
3. Knapp, A., et al.: Code of practice for the design and evaluation of ADAS. Preventive and active safety applications, eSafety for road and air transport, European Commission Project (2009)
4. DOT, HS 810 905: Integrated Vehicle-Based Safety Systems (IVBSS): Human Factors and Driver-Vehicle Interface (DVI) Summary Report (2008)
5. ISO, ISO26262: Road vehicles-functional safety. International Standard ISO (2011)
6. Yoshida, J.: Tesla's Fatal Crash: 6 Unanswered Questions:1. What did the front-camera actually see just before the crash?. https://www.eetimes.com/document.asp?doc_id=1330060&page_number=2. Accessed 1 Apr 2018
7. Spanfelner, B., et al.: Challenges in applying the ISO 26262 for driver assistance systems. Tagung Fahrerassistenz (2012)
8. Wilhelm, U., Ebel, S., Weitzel, A.: Functional safety of driver assistance systems and ISO 26262. In: Handbook of Driver Assistance Systems: Basic Information, Components and Systems for Active Safety and Comfort, pp. 109–131 (2016)

9. Ito, M.: Finding threats with hazards in the concept phase of product development. In: Barafort, B., O'Connor, R.V., Poth, A., Messnarz, R. (eds.) EuroSPI 2014. CCIS, vol. 425, pp. 277–284. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-43896-1_25
10. Weilkiens, T.: Systems Engineering with SysML/UML: Modeling, Analysis, Design. Elsevier, New York City (2011)
11. D'Souza, D.F., Wills, A.C.: Objects, Components, and Frameworks with UML: The Catalysis Approach. Addison-Wesley Longman Publishing Co., Inc., Boston (1998)
12. van Lamsweerde, A.: Requirements engineering: from system goals to UML models to software. Wiley, Chichester (2009)
13. IEC, B.S. 61882:2001: Hazard and operability studies (HAZOP studies). Application guide. British Standards Institute (2001)
14. Ito, M.: HMI requirements creation, as the collaboration work of human and machine in the safety-critical system. In: Stolfa, J., Stolfa, S., O'Connor, R.V., Messnarz, R. (eds.) EuroSPI 2017. CCIS, vol. 748, pp. 61–71. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-64218-5_5
15. Ito, M.: Cardion.spec: an approach to improve the requirements specification written in the natural language through the formal method. In: Kreiner, C., O'Connor, R.V., Poth, A., Messnarz, R. (eds.) EuroSPI 2016. CCIS, vol. 633, pp. 58–69. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-44817-6_5
16. IEC, IEC 62443-1-1: Security for industrial automation and control systems (2007)
17. Salay, R., Queiroz, R., Czarnecki, K.: An analysis of ISO 26262: machine learning and safety in automotive software, SAE Technical Paper (2018)
18. ISO, ISO/AWI PAS 21448: Road vehicles - safety of the intended functionality (under development)
19. AMASS (Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems), Standardization Survey D8.9 (2017)
20. Kelly, T.: Arguing safety-a systematic approach to managing safety cases. University of York, Department of Computer Science-Publications-YCST (1999)
21. Larrucea, X., Walker, A, Colomo-Palacios, R.: Supporting the management of reusable automotive software. In: IEEE Software, vol. 34, no. 3, pp. 40–47, May–June (2017)