# Integrating Autonomous Vehicle Safety and Security Analysis Using STPA Method and the Six-Step Model

Giedre Sabaliauskaite, Lin Shen Liew, and Jin Cui

Centre for Research in Cyber Security (iTrust)
Singapore University of Technology and Design
Singapore 487372
Email: giedre@sutd.edu.sg, linshen_liew@sutd.edu.sg, jin_cui@sutd.edu.sg

*Abstract*—**Safety and security are two inter-dependent key properties of autonomous vehicles. They are aimed at protecting the vehicles from accidental failures and intentional attacks, which could lead to injuries and loss of lives. The selection of safety and security countermeasures for autonomous vehicles depends on the driving automation levels, defined by the international standard SAE J3016. However, current vehicle safety standards ISO 26262 do not take the driving automation levels into consideration. We propose an approach for integrating autonomous vehicle safety and security processes, which is compliant with the international standards SAE J3016, SAE J3061, and ISO 26262, and which considers driving automation levels. It incorporates the System-Theoretic Process Analysis method into autonomous vehicle safety analysis, and uses the Six-Step Model as a backbone for achieving integration and alignment among safety and security processes and artefacts throughout the entire autonomous vehicle's life-cycle.**

***Keywords–Autonomous vehicle; safety; security; Six-Step Model; STPA.***

## I. Introduction

Autonomous Vehicles (AVs), the self-driving vehicles, are safety-critical Cyber-Physical Systems (CPS) – complex engineering systems, which integrate embedded computing technology into physical phenomena. Safety and security are two key properties of CPSs, which share the same goal – protecting the systems from undesirable events: failures (safety) and intentional attacks (security).

Ensuring the safety of autonomous vehicles, i.e., reducing the number of traffic crashes to prevent injuries and save lives, is a top priority in autonomous vehicle development. Safety and security are interdependent (e.g., security attacks can cause safety failures, or security countermeasures may weaken CPS safety and vice versa), therefore they have to be aligned in the early system development phases to ensure the required level of protection [1][2][3].

Although AVs could be considered to be smaller and/or less complex systems as compared to other CPSs, such as, power plants or water treatment systems, they face some unique challenges, which have to be taken into consideration when analyzing their safety and security.

Firstly, there are six different levels of driving automation ranging from no driving automation (level 0) to full driving automation (level 5), as described by the international standard SAE J3016 [4]. The levels describe who (human driver or automated system) performs the driving tasks and monitors the driving environments under certain environmental conditions. Thus, AV safety and security depend on the driving automation levels and the environmental conditions.

Secondly, the AV domain is relatively new, and therefore, there are no international standards developed specifically for AV safety and security yet. Currently, the ISO 26262 standard, which describes functional safety of road vehicles, is being used for AV safety analysis [5]. However, it is not sufficient for AVs, as argued in [6] and [7]. ISO 26262 addresses the safety of each function, or item, of the vehicle separately, since the driver is responsible for everything what falls outside the item. However, in AV, it is necessary to ensure safety at all times, especially at the high automation levels, when there is no driver in the vehicle.

Moreover, the ISO 26262-recommended hazard analysis techniques, such as Fault Tree Analysis (FTA) and Failure Modes and Effects Analysis (FMEA), are ineffective in identifying the systemic and interaction related problems of complex software intensive E/E systems [8]. In view of that, some recent works (e.g., [9][10]) employ a relatively new hazard analysis technique, System-Theoretic Process Analysis (STPA) [11], to complement the ISO 26262 for a more comprehensive AV safety analysis.

To address vehicle security needs, the SAE J3061 standard has been developed [12]. It defines cyber-security lifecycle of cyber-physical vehicle systems. However, the security lifecycle, defined in SAE J3061, is analogous to the vehicle safety lifecycle described in ISO 26262, and therefore, it is not sufficient for AV cyber-security analysis. ISO and SAE are currently jointly developing vehicle standard ISO 21434 [13], which will replace SAE J3061 in 2019.

How can we analyze AV safety and security throughout its entire life-cycle in a consistent way, and provide required level of protection? In our previous work, we proposed a Six-Step Model for modeling and analysis of CPS safety and security [15][16]. It incorporates six dimensions (hierarchies) of CPS, namely, functions, structure, failures, safety countermeasures, cyber-attacks, and security countermeasures. Furthermore, it uses relationship matrices to model inter-dependencies between these dimensions. The Six-Step Model enables comprehensive analysis of CPS safety and security, as it utilizes system functions and structure as a knowledge base for understanding the effect of failures and attacks on the system. Furthermore, we presented an initial approach

TABLE I. Driving automation levels [4][14].

| Level | Name | Execution of steering/ acceleration/deceleration | Monitoring of driving environment | Performance of DDT fallback | Driving modes (environment and conditions) |
|---|---|---|---|---|---|
| 0 | No automation | Human driver | Human driver | Human driver | N/A |
| 1 | Driver assistance | Human driver and system | Human driver | Human driver | Some driving modes |
| 2 | Partial automation | System | Human driver | Human driver | Some driving modes |
| 3 | Conditional automation | System | System | Human driver | Some driving modes |
| 4 | High automation | System | System | System | Some driving modes |
| 5 | Full automation | System | System | System | All driving modes |

for applying the Six-Step Model for AV safety and security analysis in [1].

In this paper, we extend the initial approach, proposed in [1], to enable a comprehensive analysis of AV safety and security using STPA method and the Six-Step Model, which is compliant with the international standards SAE J3016, SAE J3061, and ISO 26262.

The remainder of this paper is structured as follows. Section II describes the preliminaries. The proposed approach is explained in Section III, and a Six-Step Model example is included in Section IV. Finally, Section V concludes the paper and describes our future work.

## II. Preliminaries

This section describes the AVs, their safety and security analysis, the Six-Step Model, and STPA method.

### A. Autonomous Vehicles' Main Terms and Definitions

The real-time operational and tactical functions required to operate the vehicle in on-road traffic include lateral and longitudinal vehicle motion control, monitoring the driving environment, object and event response execution, maneuver planning, and enhancing conspicuity via lighting, signaling, etc. [4]. These functions are collectively called the Dynamic Driving Task (DDT) [4]. An automated driving system of an AV performs entire or part of DDT, depending on AV driving automation level. In addition to DDT, AVs implement DDT-fallback – a response mechanism, which enables a human driver or an automated system to take over performance of the entire DDT in case of unexpected situations, e.g., during traffic jams on freeways.

SAE International (SAE) has developed an international standard, SAE J3016 [4], to describe the levels of vehicle driving automation. The standard has been widely adopted by international organizations, such as the National Highway Traffic Safety Administration (NHTSA) [14].

There are six driving automation levels (see Table I):

- Level 0 – the human driver performs entire DDT.
- Level 1 – an automated system on the vehicle can assist the human driver to perform either the lateral or the longitudinal vehicle motion, while driver monitors the driving environment and performs the rest of DDT.
- Level 2 – an automated system performs the lateral and the longitudinal vehicle motion, while driver monitors the driving environment and performs the rest of DDT.
- Level 3 – an automated system can perform entire DDT, but the human driver must be ready to take back control when the automated system requests.
- Level 4 – there is no human driver; an automated system conducts the entire DDT, but it can operate only in certain environments and under certain conditions (driving modes).
- Level 5 – there is no human driver; an automated system performs entire DDT in all environments and under all conditions that a human driver could perform them.

Level 3-5 vehicles are called the highly automated vehicles, since their automated systems (not a human driver) are responsible for monitoring the driving environment [14]. Furthermore, level 1-4 vehicles are designed to operate only in certain environments and under certain conditions, while level 5 vehicles - in all environments and under all conditions.

AV functions can be grouped into three main categories: perception (perception of the external environment/context, in which vehicle operates), decision & control (decisions and control of vehicle motion with respect to the external environment/context that is perceived), and vehicle platform manipulation (sensing, control, and actuation of the vehicle, with the intention of achieving desired motion) [17][18]. An international standard for describing AV functions and functional interfaces, SAE J3131, is currently under development.

AV structural architecture consists of two main systems: a) cognitive driving intelligence, which implements perception and decision & control functions, and b) vehicle platform, which is responsible for vehicle platform manipulation [17]. Each system consists of components, which belong to four major groups: hardware, software, communication, and human-machine interface [18][19]. See Section IV for more details.

### B. A Six-Step Model

In our earlier work [15][16], we proposed a Six-Step Model to enable comprehensive CPS safety and security analysis. The model is constructed using the following six steps (see Figure 1):

1) The first step is aimed at modeling the functional hierarchy of the system. The functions are defined using the Goal Tree (GT), which is constructed starting with the goal (functional objective) and then defining functions and sub-functions, needed for achieving this goal. A relationship matrix, F-F, is used to define the relationships between functions, which can be high, medium, low, or very low.
2) In the second step, system's structural hierarchy is defined using the Success Tree (ST) to describe system's structure as a collection of sub-systems and units. Furthermore, the relationships between structure and functions are defined using a relationship matrix S-F, as shown in Figure 1.

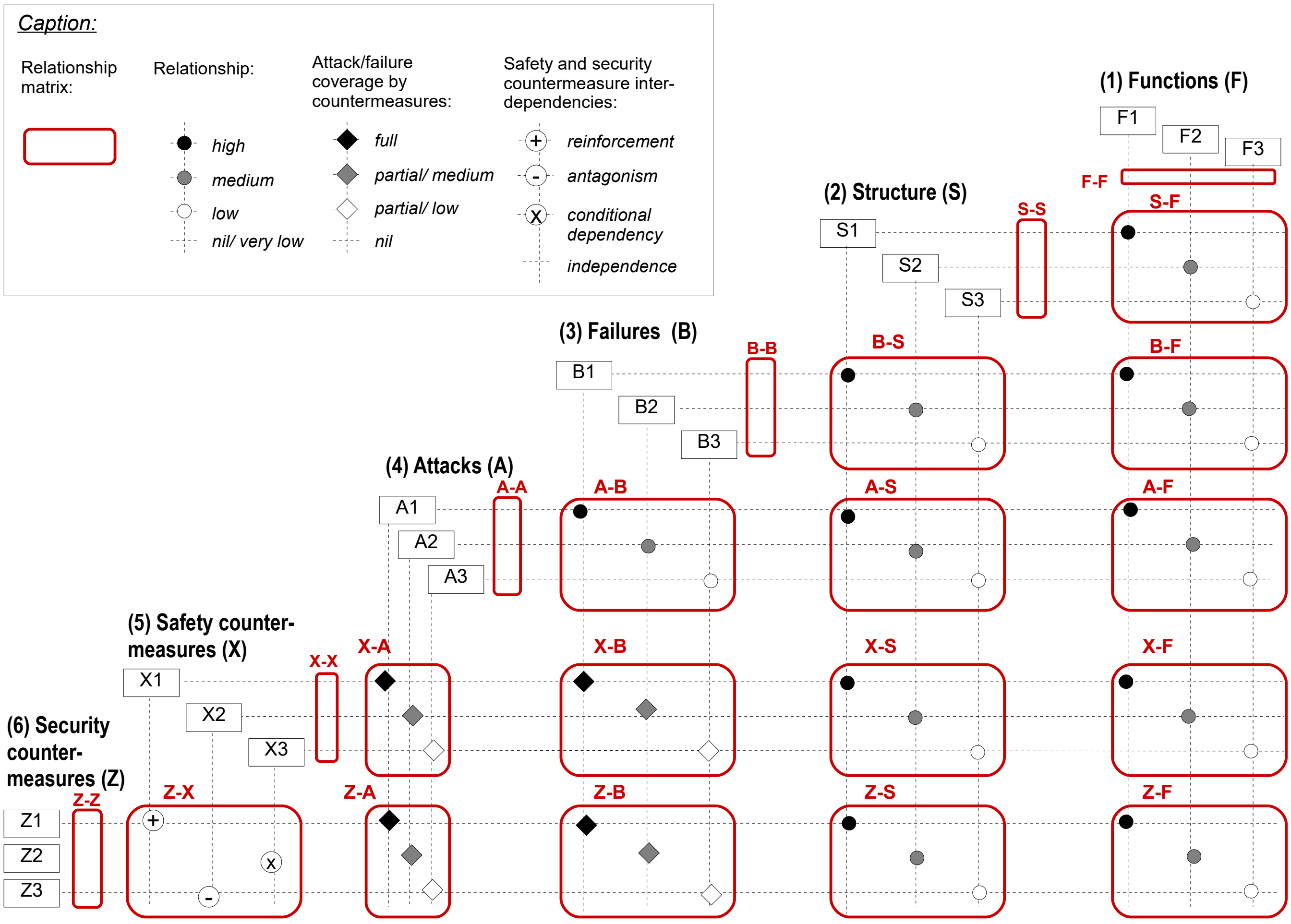

Figure 1. The Six-Step Model.

3) The third step is focused on safety hazard analysis. In this step, system's failures are identified and added to the model. In addition, the relationships between failures, system structure and functions are identified, and the corresponding relationship matrices – B-B, B-S, and B-F – are added to the model.

4) The fourth step focuses on security threat analysis. In this step, attacks are identified and added to the model along with the relationship matrices to describe relationships between attacks, failures, structure and functions. Relationship matrix A-B (attacks – failures) is used to determine the failures, which could be triggered by a successful attack.

5) In the fifth step, safety countermeasures are added to the model and their relationships are identified. Matrices X-A and X-B show the coverage of attacks and failures by safety countermeasures, where white rhombus indicates that the countermeasure provides low protection from attack/failure; gray rhombus - medium protection; black rhombus - full protection (see Figure 1).

6) Finally, in the last step, security countermeasures are added to the model and their relationships are established. Similarly to matrices X-A and X-B from the previous step, two new matrices Z-A and Z-B are added to define the coverage of attacks and failures by security countermeasures. The security countermeasures, added in this step, could be used to protect the system from attacks and failures, not covered by the safety countermeasures. Furthermore, matrix Z-X is used to capture the inter-dependencies between safety and security countermeasures, such as reinforcement, antagonism, conditional dependency, and independence, as defined in [20].

After completion of steps 5 and 6, it is important to analyze if there were any changes made to system's structure, as some countermeasures might require the use of additional components, e.g., sensors or controllers. If the changes occur, it is necessary to return to the step 2 to add the new components and then repeat steps 3-6.

The Six-Step Model, constructed throughout steps 1-6, interconnects six hierarchies of the systems (functions, structure, failures, attacks, and safety and security countermeasures) by forming a hexagon-shaped structure of their relationships, as shown in Figure 2. The relationships help to ensure alignment between these hierarchies, and they have to be maintained throughout the entire system's life-cycle.

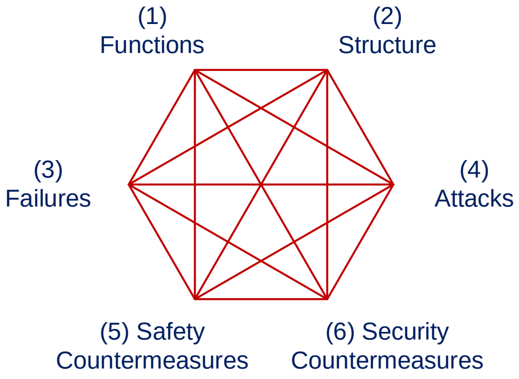

Figure 2. Relationships among hierarchies of the Six-Step Model.

### *C. AV Safety Analysis*

The ISO 26262 standard [5] defines functional safety for automotive equipment applicable throughout the life-cycle of all automotive Electronic and Electrical (E/E) safety-related systems. It aims to address possible hazards caused by the malfunctioning behavior of E/E systems. The safety process consists of several phases, such as concept, product development, and production, operation, service and decommissioning. Hazard Analysis and Risk Assessment (HARA) is performed during the concept phase, where hazardous events, safety risks, and safety goals are identified. These goals are further refined into the safety requirements, and the safety countermeasures are designed and implemented. Fault Tree analysis [19] can be used during HARA to identify the conditions and events that could lead to high-level hazardous events. Fault tree refines top-level hazardous event into intermediate events and basic events, which are interconnected by AND and OR logical operators.

However, as ISO 26262 requires the presence of the human driver inside the vehicle to respond to unexpected environments and conditions, it is likely to be unfit for AVs where humans have little or no part in the driving. In fact, various means have been proposed to complement the ISO 26262 standard in ensuring the safety of complex software intensive E/E systems like AVs. For instance, [6][21][22] highlight the importance of having an adequate item definition; note that the *item definition* is a prerequisite for Hazard Analysis and Risk Assessment (HARA) process as per ISO 26262 standard.

Traditionally, the *item* delivers only one function like steering and braking, and malfunctions caused by interactions between *item* and other entity are simply eliminated by design [21]. In contrast, the *item* for AVs may deliver multiple functions (e.g., a complex braking system, which includes regenerative braking), and thus defining it would require more careful consideration. In view of that, Ibarra et al. [21] model the item definition based on Goal Structuring Notation [23].

Inadequate *item definition* would eventually result in inadequate Safety Goals (SGs). Such SGs could be violated even when the system is fault-free (e.g., having no sensor failure at all). There is an ISO work-group called Safety of the Intended Functionality (SOTIF) aiming to provide a guidance on handling such violations, but its specification is yet to be published [24]. In [6], Warg et al. suggest that the *item definition* should be a product of an iterative process, which comprises three steps: (1) perform HARA where generic operational situation and hazard trees are used to generate a list of Hazardous Events (HEs), and the trees get updated according to the Scope and Requirements of the Function (SRF) updated on last iteration; (2) identify the dimensioning HEs based on a set of rules as described in [25]; (3) refine the SRF according to the dimensioning HEs. The iterative process ends once the HEs and SRF are mature (based on certain criteria) for creating the safety goals and an *item definition*, which are then inputted to the *functional safety concept* phase.

How an inadequate *item definition* would lead to inadequate safety requirements (e.g., Functional Safety Requirements (FSRs) and Technical Safety Requirements (TSRs)) is exemplified in [22], where Bergenhem et al. claim that there is a substantial gap between any adjacent levels of safety requirements (e.g., between SG and its corresponding FSRs, and between FSR and its corresponding TSRs). Such gap necessitates a complex rationale for verification of the completeness and correctness of these requirements. Therefore, they recommend that each safety requirement level be refined to a certain extent - e.g., prior to deriving its FSRs, a high-level SG is translated into multiple lower-level SGs to reduce the gap and subsequently ease the rationale and hence the verification.

In addition, another challenge lies in specifying the HEs for the AVs. Conventionally, in the context of automotive industry, the HEs were identified by using hazard analysis techniques recommended by the ISO 26262 such as Fault Tree Analysis (FTA), and Failure Modes and Effects Analysis (FMEA). A brief review on the effectiveness of these techniques with respect to modern complex E/E systems can be found in [8]. In essence, these traditional techniques are based on simple linear chain-of-event accident causality models, originally intended for systems where the safety issues are mainly caused by random hardware failures; they are unfit for AVs, which could be compromised by software error, dysfunctional interaction, etc. apart from hardware failures.

Recently, a relatively new hazard analysis technique known as STPA has gained popularity among the researchers and practitioners in engineering the safety of complex systems in various domains [26][27][28][29][30]. STPA [11] is developed based on STAMP (Systems-Theoretic Accidents Model and Process) - a novel accident causality model that consider the safety of a complex system as a system control problem rather than a component failure or reliability problem. It aims to identify inadequate control scenarios, which could result in unwanted losses/accidents, and then develop detailed safety constraints to avoid/mitigate such scenarios. Note that the inadequate controls can occur owing to human error, dysfunctional interaction, software failure, etc. Arguments on why traditional safety engineering approaches (including traditional accident causality models) are inadequate for addressing the safety of complex systems can be found in [11]. In fact, some works such as [26] and [27] have demonstrated that STPA is able to identify not only all the hazardous/unsafe scenarios, which FTA identifies, but also those that FTA fails to identify.

To address the safety needs of AV more comprehensively, both [9] and [10] have proposed to integrate STPA into the concept phase of ISO 26262. Figure 3 illustrates the integration between STPA and ISO 26262 concept phase, which consists of five main stages:

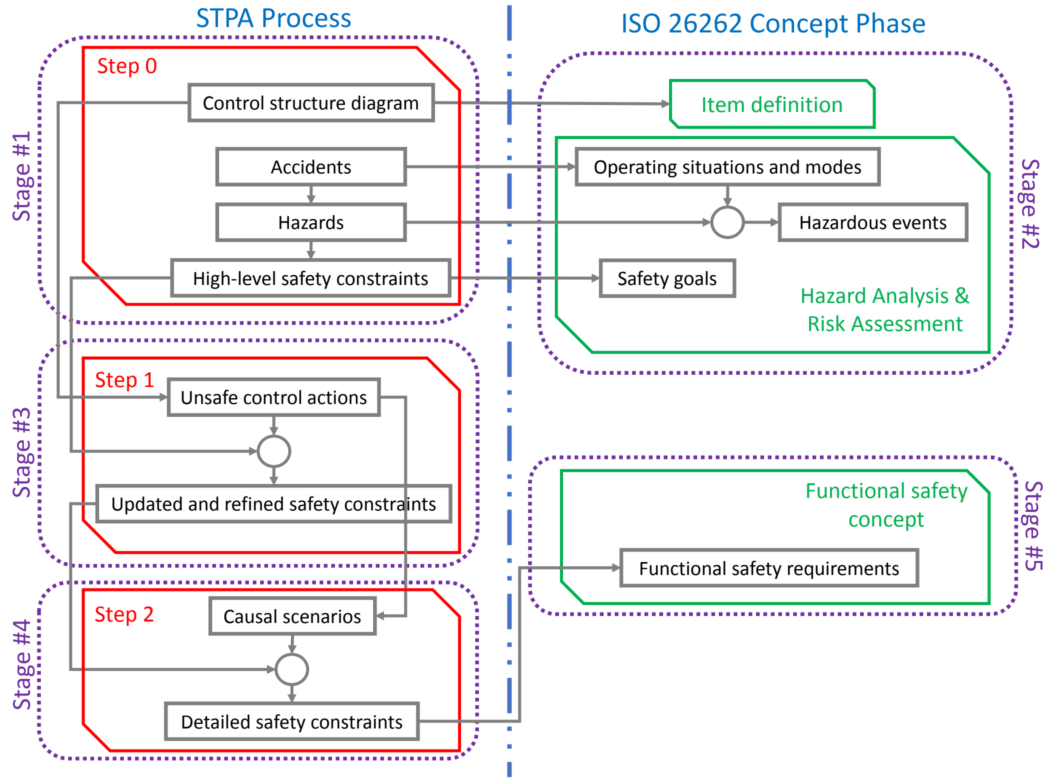
STPA Process
ISO 26262 Concept Phase
Stage #1
Step 0
Control structure diagram
Accidents
Hazards
High-level safety constraints
Item definition
Operating situations and modes
Hazardous events
Safety goals
Hazard Analysis & Risk Assessment
Stage #2
Stage #3
Step 1
Unsafe control actions
Updated and refined safety constraints
Functional safety concept
Functional safety requirements
Stage #5
Stage #4
Step 2
Causal scenarios
Detailed safety constraints

Figure 3. Integration of STPA into the concept phase of ISO 26262.

1) Perform Step 0 of STPA. Firstly, identify the accidents that could happen to the AV. Secondly, identify the high-level system hazards, which could lead to the identified accidents. Thirdly, define the high-level safety constraints for mitigating/avoiding the identified hazards. Finally, draw the system-level control structure - a diagram that represent the functional model of the system, which shows the major components of the system as well as their interfaces and boundaries.
2) Utilize the output of STPA Step 0 in ISO 26262 concept phase. Firstly, the information extracted from the control structure could contribute to a more precise *item definition*. Secondly, the list of accidents is used to derive the operating situations and modes. Thirdly, identified hazards and derived operating situations are combined to form the HEs. Fourthly, the high-level safety constraints are considered in formulating the safety goals for the HEs.
3) Perform Step 1 of STPA. Firstly, identify the control actions from the control structure diagram. Secondly, check if the control actions can be unsafe (i.e causing some previously identified hazards or additional ones), if they are not provided, or incorrectly provided, or untimely provided, or stopped/applied too soon/long. Then, translate the unsafe control actions into safety constraints by using the the guide words like ”shall” and ”must”, which could also be used to refined previously identified safety constraints.
4) Perform Step 2 of STPA. Firstly, identify the causal scenarios for the unsafe control actions, based on the control structure diagram. Secondly, derive new or more detailed safety constraint for the identified causal scenarios.
5) Utilize the output of STPA Step 2 in ISO 26262 concept phase. The finalized and detailed safety constraints are inputted to the *functional safety concept* of ISO 26262 to derive the functional safety requirements.

### *D. AV Security Analysis*

SAE J3061 is a vehicle cyber-security standard, which was developed using the ISO 26262 standard as a base. Thus, both standards consist of similar phases. Security process, defined by SAE J3061, includes concept, product development, and production & operation phases. Threat Analysis and Risk Assessment (TARA) is performed during the concept phase, where threats, security risks, and security goals are defined. In the product development phase, security requirements are defined based on the security goals, and the security countermeasures are developed.

Attack tree analysis [12][31] is often used for performing TARA. It helps to determine the potential paths that an attacker could take to lead to the top-level threat [12]. An attack tree is a graph, where the nodes represent attack events, and the edges - attack paths through system, which could be connected using AND and OR gates.

Behavior diagrams, such as Data-Flow Diagrams (DFD) [32] and Information-Flow Diagrams (IFD) [16] could be used for identifying the attacks to be included in attack trees analysis. DFDs include elements, such as processes, data flows, and data store, and are used to model data flows between software components. IFDs include units and information flows between them, and could be used to model information flows between software and hardware components, such as actuators, controllers, sensors, etc. In [16], we proposed a method for generating IFDs using the Six-Step model in order to identify possible attacks on CPSs.

## III. Integrated Autonomous Vehicle Safety and Security Analysis Approach

This section proposes an approach for integrated AV safety and security analysis, which used the Six-Step Model and STPA methods, and is compliant with the international standards SAE J3016, SAE J3061 and ISO 26262.

Figure 4 describes the proposed approach and shows the relationships between steps of the Six-Step Model and various artefacts from AV definition and design, safety analysis, and security analysis processes.

The steps of the AV Six-Step Model are performed in the following order:

- Steps (1) and (2). Autonomous driving functions and the systems (structure), which implement these functions, are defined during AV definition and design process. As a result, AV functional and structural hierarchies are defined and added to the Six-Step Model, along with their relationships. The functions and structure will be continuously updated based on the results of the safety and security analysis.
- Steps (3) and (4). These steps correspond to AV vulnerability (hazard and threat) analysis. On the safety side, HARA (as defined by ISO 26262) and STPA are performed in order to identify and evaluate hazardous events, and define AV functional safety requirements. At the end of the hazard analysis phase, failures, which are considered in security requirements, are extracted from the fault trees and added to the the Six-Step Model (Step (3)). On the security side, TARA (as defined by SAE J3061) is performed in order to evaluate security threats and derive AV functional security requirements. The AV structural hierarchy, defined in step (2), could be used to define attack surfaces and construct information-flow models (see [16]), which helps to identify possible attacks and construct attack trees, as described in Section II-D. The risks associated with each attack are then evaluated and security requirements are defined. Similarly to failures, the attacks, included in security requirements, are extracted from the attack trees and added to the Six-Step Model (Step (4)). The relationships between attacks, failures, functions, and structures, are also added to the Six-Step model.
- Steps (5) and (6). During these steps, safety and security countermeasures are selected and added to the model along with their relationships to remaining elements of the model. On the safety side, functional safety requirements are refined into technical requirements and corresponding countermeasures are designed for satisfying these requirements. Similarly, on the security side, functional security requirements are decomposed into technical requirements for security countermeasures. The countermeasures from both sides are added to the Six-Step Model to analyze their relationships to the remaining elements of the model. In particular, the matrices are useful to make sure that each countermeasure is really needed (addresses attacks/failures not completely covered by any other countermeasures, shown in matrices X-A, X-B, Z-A, and A-B), and that there are no contradictions among countermeasures (matrix Z-X).

The AV Six-Step Model, constructed during steps (1)-(6), is a backbone of AV vulnerability analysis. It supports three AV processes, namely, AV definition and design, AV safety analysis, and AV security analysis, as shown in Figure 4. Furthermore, it enables integration of safety and security artefacts, developed throughout the entire AV life-cycle (such as failures, attacks, safety and security countermeasures) into AV function and structure hierarchies to assure their consistency and completeness.

The AV Six-Step Model has to be maintained throughout the entire AV life-cycle. This is particularly important for security analysis, as new threats are continually identified and have to be analyzed.

## IV. Six-Step Model Example of an AV

This example describes a high automation AV. Its Six-Step Model is shown in Figure 5. Due to space limitations, only an excerpt of the Six-Step Model is included in Figure 5. Furthermore, only the high degree relationships between elements are shown.

The AV, described in this example, performs three main autonomous driving functions, i.e., perception, decision & control, and vehicle platform manipulation, as described in Section II-A. The perception function can be further decomposed into sensing, sensor fusion, localization, semantic understanding, and world model (see [17]). These functions are added at the top of to Six-Step Model and their inter-relationships are identified, as shown in Figure 5 step (1).

The main systems of AV, which implement driving automation functions, are: cognitive driving intelligence, vehicle platform, and communication system [17][18]. The cognitive driving intelligence includes on-board computer and external sensors for perception of environment, such as Light Detection and Ranging (LIDAR), cameras, and ultrasound sensors [33]. The vehicle platform includes controllers (ECUs) and actuators, which implement the desired motion. The communication system includes in-vehicle and V2X (vehicle to vehicle, infrastructure, and humans) communication networks. In this example, only in-vehicle communication is considered. All these structural elements are added to model in step (2).

In steps (3) and (4), we add failures and attacks to the Six-Step Model. In this example, we describe the LIDAR failures and attacks. LIDAR combined with camera are used for navigation in AVs. Together with other sensors, they provide necessary information for performing AV localization function (determining the location of vehicle with respect to its surroundings). LIDAR includes the following components: laser lens filter, receiver, power regulator, rotating mirror, etc. In this example, LIDAR is connected to on-board computer through Ethernet in order to send its readings.

Fault trees are commonly used for safety analysis, as described in Section II-C. An example of LIDAR fault tree is

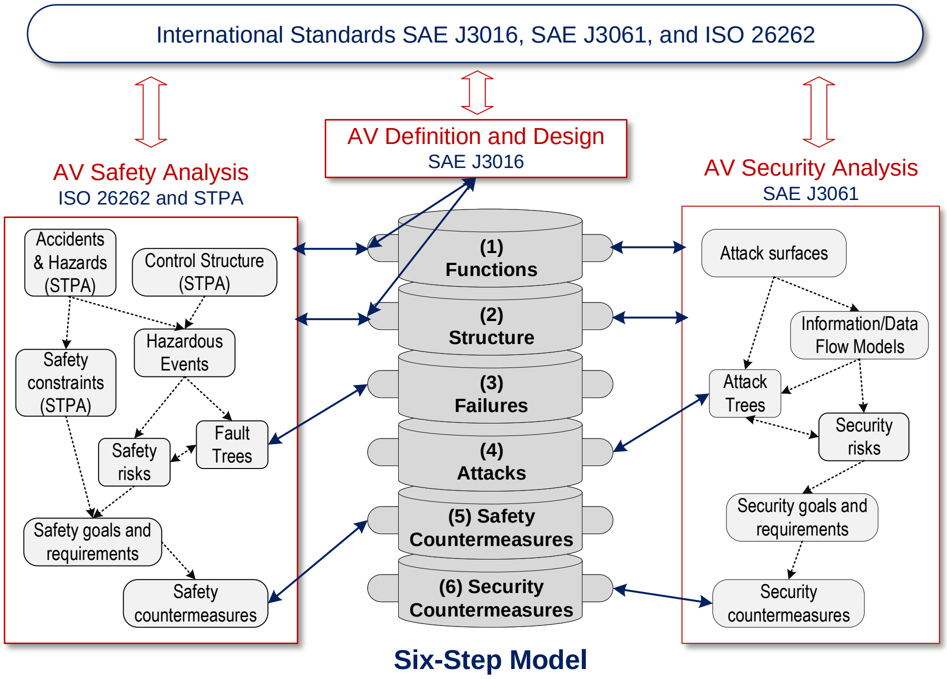

Figure 4. The Six-Step Model as a backbone for integrated AV safety and security analysis.

shown in Figure 6. The top-level undesired event is localization failure, which uses LIDAR readings in combination with other sensors for AV localization. Thus, localization failure could happen if either LIDAR or other sensors fail. LIDAR failure can be further decomposed into electrical, LIDAR component, or LIDAR communication failures, as shown in Figure 6. At the end of fault tree analysis, failures are added to the Six-Step Model. Due to space limitations, only a high-level LIDAR failure is shown in Figure 5.

STPA can be used to complement AV safety analysis and help identify failures, not captured in fault trees, as described in Section II-C. Figure 7 depicts a high-level control structure for a typical AV, which is a prerequisite in STPA for identifying the inadequate controls that could result in hazards or hazardous events. The arrows shown in Figure 7 signify the control relationships between the components. For example, the sensors (e.g., LIDAR) send their readings to the computer for computation purposes, and subsequently the computer commands the ECUs for manipulating the vehicle's motion accordingly. Each control can be evaluated in four different ways (i.e., not provided, incorrectly provided, untimely provided, and stopped/applied too soon/long). For instance, if the LIDAR readings are not provided, then the localization, which is performed by the computer and is dependent on LIDAR readings, is likely to fail. A corresponding safety constraint would be “The computer must always receive LIDAR readings”. Then, how each inadequate control could occur is to be identified. For example, the computer receives no data from LIDAR because it is disconnected from the Ethernet. Hence, the corresponding safety constraint would be “LIDAR must be connected to the computer at all times”. Certainly, more low-level control structure diagrams could be drawn to show more explicit interaction between the components as well as their corresponding sub-components; thus, one can derive more explicitly the unsafe control actions and their causal scenarios (failures) and hence the safety constraints. At the end of integrated safety analysis, the failures and safety countermeasures, as well as the updated structure and functions, are added to the Six-Step model.

Attack trees can be used for security analysis, as described in Section II-D. They show attack paths through the system. An example of a LIDAR attack tree is shown in Figure 8. As we can see from Figure 8, an attacker can execute either cyber or direct physical attack on the LIDAR. To execute cyber attack with the goal to alter LIDAR readings, an attacker can use Ethernet, since LIDAR is connected to Ethernet. Two common types of attacks, deception and Denial of Service (DoS) can be performed on sensor readings. Deception attack is used to modify sensor readings, while DoS attack - to prevent on-board computer from timely receiving the readings. Alternatively, an attacker can get access to the on-board computer and modify the LIDAR readings received by on-board computer, as shown in Figure 8. Attacks on LIDAR and security countermeasures are summarized in [34]. The information from the LIDAR attack tree (Figure 8) is added to the Six-Step Model in step (4) (see Figure 5).

As we can see from Figure 5, the main function affected by either the LIDAR failure or attack is the sensing function. Furthermore, there is a strong relationship between LIDAR attack and failure, LIDAR attack is strongly related to Ethernet (i.e., an attacker can attack LIDAR through Ethernet).

To mitigate sensor attacks and failures, it is necessary to provide sensor redundancy and perform sensor fusion. A combination of LIDAR, Radar, and Camera provides good coverage of AV tasks in most of the environmental conditions [33]. Radar is added to the model in step (5) as a safety

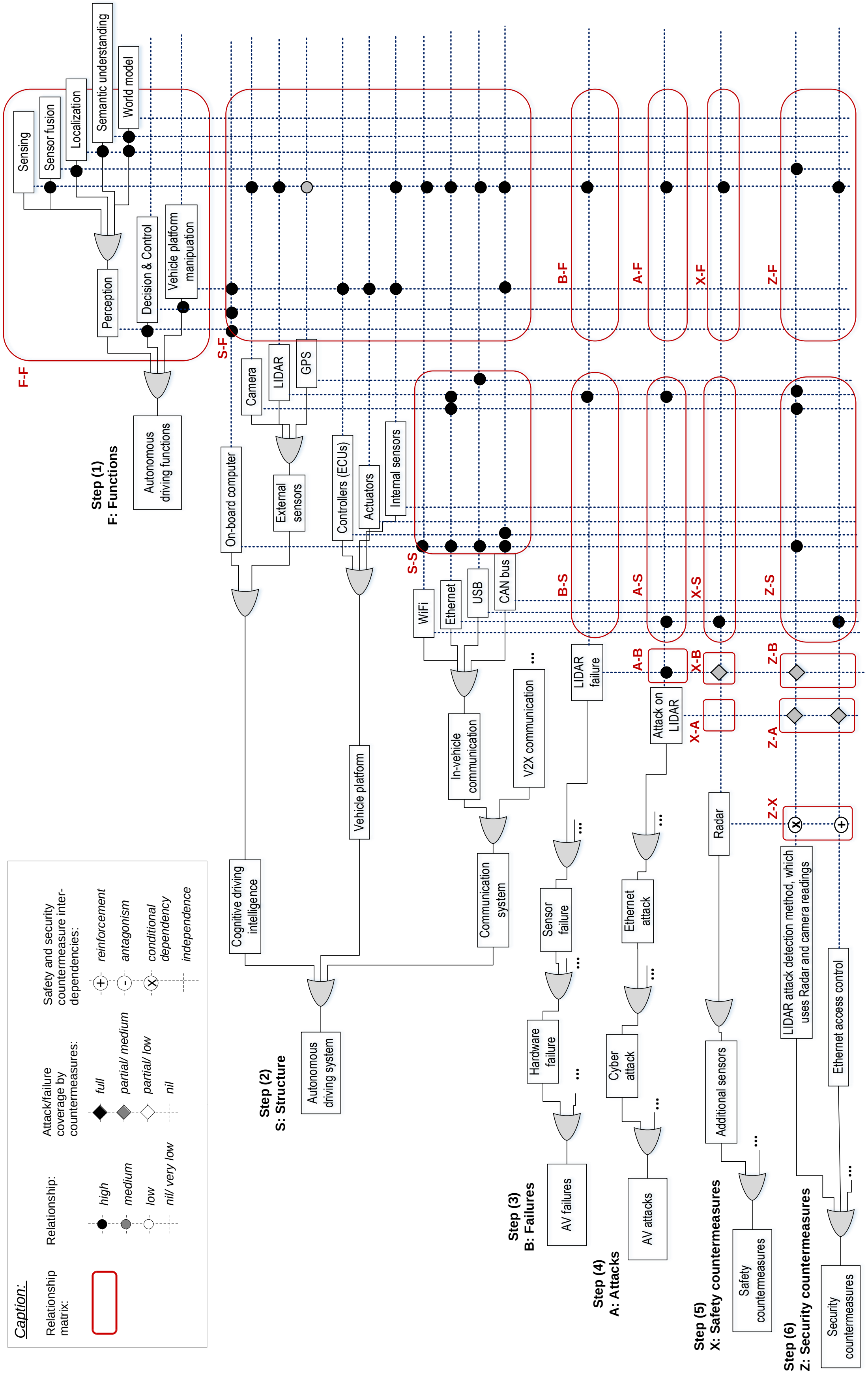

Figure 5. An example of AV Six-Step Model.

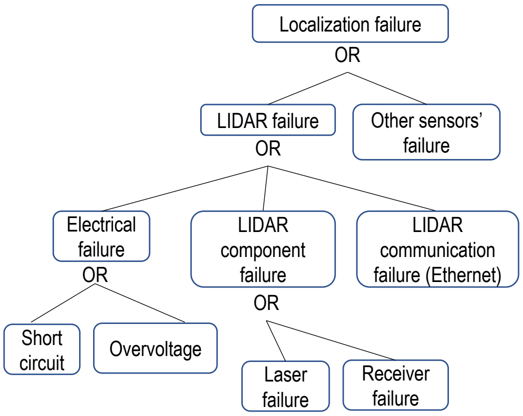

Figure 6. LIDAR failure tree example.

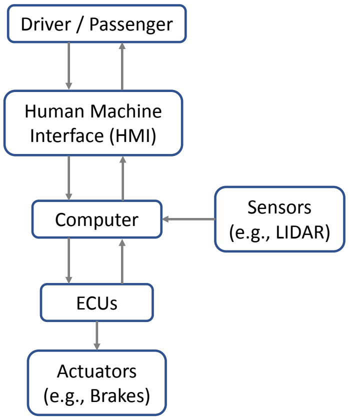

Figure 7. High-level control structure diagram of an AV.

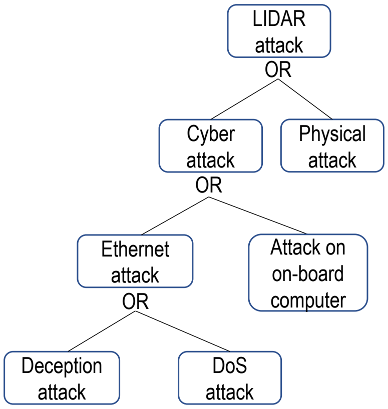

Figure 8. LIDAR attack tree example.

countermeasure. In case of LIDAR failure, Radar and Camera will still be able to perform sensing of the driving environment.

Security countermeasures could include redundancy: multiple LIDARs, or V2X communication to compare measurements of target vehicle with nearby vehicles [34]. However, due to high cost of LIDAR, multiple LIDARs are not considered in this AV. Furthermore, there is no V2X communication in this AV example. If the vehicle had V2X communication, LIDAR attacks could be detected by cross-comparing LIDAR readings of the nearby vehicles.

Various LIDAR attack detection and mitigation methods can be implemented inside on-board computer, e.g., LIDAR attacks can be detected by comparing LIDAR readings to Radar and Camera readings, while shorter or randomized LIDAR scanning interval could help in preventing the attacks [34]. In Figure 5, a security countermeasure, "LIDAR attack detection method, which uses Radar and Camera readings", is added. Additional countermeasure, "Ethernet access control", is used to prevent LIDAR attacks.

Matrices X-A, X-B, Z-A, Z-B, and Z-X are very useful for integrated safety and security analysis. X-B shows that Radar provides partial coverage of LIDAR failure, as Radar cannot fully replace LIDAR. Z-A and Z-B indicate that LIDAR attack detection method will be able to provide coverage not only for LIDAR attacks, but also failures, as it will detect corrupt LIDAR readings, which could happen in either case. Finally, matrix Z-X shows the inter-dependencies between safety and security countermeasures. As we can see from Figure 5, Radar (safety countermeasure) and the LIDAR attack detection method (security countermeasure) share a conditional dependency (denoted by x), i.e., in order to implement the attack detection method, we need a Radar; while Radar and Ethernet access control mechanism reinforce each other.

As the new structural component, Radar, has been added to the model in Step (5), it is necessary to return to the step (2) to include it to AV structural hierarchy and to establish its relationships to the remaining elements of the model.

## V. Conclusion and Future Work

In this paper, an approach for integrated AV safety and security analysis is proposed, which is compliant with the international standards SAE J3016, SAE J3061, and ISO 26262. STPA method is integrated into the concept phase of ISO 26262 for acquiring more accurate and detailed lists of functions, failures and safety countermeasures. The proposed method uses the Six-Step Model for achieving and maintaining integration and alignment among safety and security artefacts throughout the entire AV life-cycle. The Six-Step Model incorporates six hierarchies of AVs, namely, functions, structure, failures, attack, safety countermeasures, and security countermeasures. An example of an AV Six-Step Model is included to demonstrate the usefulness of the proposed approach.

Future work will include the refinement of the proposed approach to facilitate its application in industry and the use by other researchers. Furthermore, we are currently extending the proposed approach for application to transportation system (system-of-systems) level.

## References

[1] G. Sabaliauskaite and J. Cui, "Integrating Autonomous Vehicle Safety and Security," in Proceedings of the $2^{nd}$ International Conference on Cyber-Technologies and Cyber-Systems (CYBER) November 12–16, 2017, Barcelona, Spain. IARIA, Nov. 2017, pp. 75–81, ISBN: 978-1-61208-605-7.

[2] G. Sabaliauskaite and A. P. Mathur, Aligning Cyber-Physical System Safety and Security. Cham: Springer International Publishing, 2015, pp. 41–53. [Online]. Available: https://doi.org/10.1007/978-3-319-12544-2_4

[3] L. Piètre-Cambacédès and M. Bouissou, "Cross-fertilization between safety and security engineering," Reliability Engineering & System Safety, vol. 110, 2013, pp. 110 – 126.

[4] SAE J3016: Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems. SAE International, Sep. 2016.

[5] ISO26262-2:2011, Road Vehicles – Functional Safety – Part2: Management of Functional Safety. International Organization of Standardization, ISO, 2011.

[6] F. Warg et al., Defining Autonomous Functions Using Iterative Hazard Analysis and Requirements Refinement. Cham: Springer International Publishing, 2016, pp. 286–297. [Online]. Available: https://doi.org/10.1007/978-3-319-45480-1_23

[7] A. Abdulkhaleq et al., "A systematic approach based on stpa for developing a dependable architecture for fully automated driving vehicles," Procedia Engineering, vol. 179, no. Supplement C, 2017, pp. 41 – 51.

[8] Q. V. E. Hommes, "Assessment of safety standards for automotive electronic control systems," 2016, (Report No. DOT HS 812 285).

[9] A. Abdulkhaleq, S. Wagner, D. Lammering, H. Boehmert, and P. Blueher, "Using STPA in compliance with ISO 26262 for developing a safe architecture for fully automated vehicles," CoRR, vol. abs/1703.03657, 2017.

[10] A. Mallya, V. Pantelic, M. Adedjouma, M. Lawford, and A. Wassyng, Using STPA in an ISO 26262 Compliant Process. Cham: Springer International Publishing, 2016, pp. 117–129.

[11] N. G. Leveson, Engineering a Safer World: Systems Thinking Applied to Safety.

[12] SAE J3061: Cybersecurity Guidebook for Cyber-Physical Vehicle Systems. SAE International, Jan. 2016.

[13] ISO/SAE AWI 21434, Road Vehicles – Cybersecurity engineering. International Organization of Standardization, ISO, Under development.

[14] Automated Driving Systems 2.0. A Vision for Safety. National Highway Traffic Safety Administration, NHTSA, U.S. Department of Transportation, Sep. 2017.

[15] G. Sabaliauskaite, S. Adepu, and A. Mathur, "A six-step model for safety and security analysis of cyber-physical systems," in the 11th International Conference on Critical Information Infrastructures Security (CRITIS), Oct. 2016.

[16] G. Sabaliauskaite and S. Adepu, "Integrating six-step model with information flow diagrams for comprehensive analysis of cyber-physical system safety and security," in 2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE), Jan. 2017, pp. 41–48.

[17] S. Behere and M. Törngren, "A functional reference architecture for autonomous driving," Inf. Softw. Technol., vol. 73, no. C, May 2016, pp. 136–150. [Online]. Available: http://dx.doi.org/10.1016/j.infsof.2015.12.008

[18] S. Kato, E. Takeuchi, Y. Ishiguro, Y. Ninomiya, K. Takeda, and T. Hamada, "An open approach to autonomous vehicles," IEEE Micro, vol. 35, no. 6, Nov. 2015, pp. 60–68.

[19] P. Bhavsar, P. Das, M. Paugh, K. Dey, and M. Chowdhury, "Risk analysis of autonomous vehicles in mixed traffic streams," Transportation Research Record: Journal of the Transportation Research Board, vol. 2625, 2017, pp. 51–61.

[20] L. Piètre-Cambacédès and M. Bouissou, "Modeling safety and security interdependencies with bdmp (boolean logic driven markov processes)," in 2010 IEEE International Conference on Systems, Man and Cybernetics, Oct. 2010, pp. 2852–2861.

[21] I. Ibarra, S. Hartley, S. Crozier, and D. Ward, "Iso 26262 concept phase safety argument for a complex item," 2012.

[22] C. Bergenhem, R. Johansson, A. Söderberg, J. Nilsson, J. Tryggvesson, M. Törngren, and S. Ursing, "How to reach complete safety requirement refinement for autonomous vehicles," in CARS 2015 - Critical Automotive applications: Robustness & Safety, M. Roy, Ed., Paris, France, Sep. 2015.

[23] I. Habli, I. Ibarra, R. Rivett, and T. Kelly, "Model-based assurance for justifying automotive functional safety," 2010.

[24] "Standardization efforts on autonomous driving safety barely under way," The Hansen Report on Automotive Electronics, 2017.

[25] R. Johansson, "Efficient Identification of Safety Goals in the Automotive E/E Domain," in 8th European Congress on Embedded Real Time Software and Systems (ERTS 2016), Toulouse, France, Jan. 2016.

[26] J. Chen, Y. Lu, S. Zhang, and P. Tang, "Stpa-based hazard analysis of complex uav system in take-off," in The 3rd International Conference on Transportation Information and Safety, Wuhan, P. R. China, 2015.

[27] T. Ishimatsu, N. G. Leveson, J. Thomas, M. Katahira, Y. Miyamoto, and H. Nakao, "Modeling and hazard analysis using stpa," in Proceedings of the 4th IAASS Conference, Making Safety Matter, Huntsville, Alabama, USA, May 2010.

[28] P. Asare, J. Lach, and J. A. Stankovic, "Fstpa-i: A formal approach to hazard identification via system theoretic process analysis," in 2013 ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS), Philadelphia, PA, USA, 2013.

[29] A. Abdulkhaleq and S. Wagner, "Experiences with applying stpa to software-intensive systems in the automotive domain," 2013.

[30] A. Abdulkhaleq, S. Wagner, and N. Leveson, "A comprehensive safety engineering approach for software-intensive systems based on stpa," Procedia Engineering, vol. 128, Oct. 2015, pp. 2 – 11.

[31] B. Schneier, Attack Trees. Wiley Publishing, Inc., Indianapolis, Indiana, 2015, in Book, Secrets and Lies.

[32] Z. Ma and C. Schmittner, "Threat modeling for automotive security analysis," Advanced Science and Technology Letters, vol. 139, 2016, pp. 333–339.

[33] "Beyond the Headlights: ADAS and Autonomous Sensing," 2016, URL: http://woodsidecap.com/wp-content/uploads/2016/12/20160927-Auto-Vision-Systems-Report_FINAL.pdf [accessed: 2018-05-08].

[34] J. Petit, B. Stottelaar, M. Feiri, and F. Kargl, "Remote Attacks on Automated Vehicles Sensors: Experiments on Camera and LiDAR," in Black Hat Europe, Nov. 2015.