On Verify and Validate a Next Generation Automotive Communication Network[a]

Sebastian Brunthaler, Thomas Waas and Markus Kucera
OTH Regensburg, Regensburg, Germany

Keywords: Automotive, Ethernet, In-car, Vehicle Communications, Vehicle Networks, Car-to-Car, Autonomous Driving.

Abstract: As a result of the enormous growth in data traffic for autonomous driving, the conventional in-vehicle network is no longer sufficient and requires new types of network concepts in a vehicle. This part of the automobile is known as the next generation communication network. Since the new car-systems can be extended by various services at any time, the network must adapt dynamically to new requirements wherever possible. For example, data flow must be configured dynamically between new services. Also data rates will be much higher in the future than today. This is one of the main reasons why we need to search for new technologies for data transfer in vehicles. This is based on an in-vehicle ethernet network. The process of configuring networks automatically has been discussed several times in recent years. One of the next steps is verifying and validating the automatic configuration process during the development of the new communication network. This research paper identifies several ways to ensure the automatically generated network configuration leads to a secure system. To achieve that, other parts of the company's enterprise IT architecture and network technologies, the conventional vehicle network and other options for verification and validation are analysed.

1 INTRODUCTION

With the introduction of autonomous driving in road traffic, vehicles must be able to process significantly more data than before. For example a few years ago, a car used to have one or two sensors, like cameras, which used to transmit data through the system. In the future, several cameras are expected to process high-resolution data by self-learning ADAS (Advanced Driver-Assistance Systems) and autonomous driving systems. Also, the resolution of images is constantly increasing as the quality of images increases. All these data has to be transmitted efficiently via the new network and forwarded to their destination in real time, if required. This is necessary because a vehicle must be able to classify objects and people within milliseconds. As an example, this is a way to avoid accidents caused by incorrectly detected elements. Therefore, this will be even more difficult in the future since, according to a prediction, more than 70% of the world's population is going to live in urban areas by 2050. Hence, the conurbations create complex driving situations for autonomous driving due to an increasing number of traffic. It presents a new dimension of influences and their complexity the automobile will have to cope with in the future. The current vehicle network is not designed for the enormously increasing requirements. Rigid bus systems such as CAN[1], LIN[2] and FlexRay[3] are no longer able to transmit the increased data volumes reliably and in the time required. Additionally, it is important to switch to a new vehicle network that has the functionality to respond to dynamic changes in the communications environment. For this, the Ethernet technology would be chosen. However, this is not going to replace the bus systems completely. The main application area will be handling the communication between the central part of the network and the connection of sensors, for instance, with high data throughput. For illustration, figure 1 shows the evolution from a currently rigid network to a server architecture. In traditional architecture, each computing unit is already configured statically in the beginning and usually remains consistent. In a domain architecture on the other hand, these computing units are usually consistent connected to each other with so-called servers. Thus, they can be configured freely and extended in their functionality. The next step is

[a]OTH Regensburg, Continental Automotive GmbH

[1]https://ieeexplore.ieee.org/document/4678912
[2]https://ieeexplore.ieee.org/document/1286317
[3]https://ieeexplore.ieee.org/document/4677484

Brunthaler, S., Waas, T. and Kucera, M.
On Verify and Validate a Next Generation Automotive Communication Networka.
DOI: 10.5220/0008191601210127
In *Proceedings of the 9th International Conference on Pervasive and Embedded Computing and Communication Systems (PECCS 2019)*, pages 121-127
ISBN: 978-989-758-385-8

a zone architecture, containing a central network of servers. Sensors and actuators are connected to the network via so-called zone ECUs. These in turn are connected to the central computing units like servers, which ultimately process the data. The final state of in vehicle network is nearly the same as the zone architecture. To the central powerful server units, hardware could be added and associated services runs on one of the server units. Unlike the zone architecture, the zone controllers are no longer required. The last two expansion stages of the E/E architecture (Electrical/Electronic Architecture) enable a modular expansion of a vehicle with various services and hardware through Plug&Play. This is a key aspect in the development of a next generation communication network. Vehicles are expected to be more personalized in the future resulting in a broad spectrum of configuration options. Therefore, variety of combinations requires the automatic configuration of a network to meet modularity requirements.(Carlson et al., 2019)

Due to the modularity of the system, current projects such as the A3F project which is explained in the section 2, aim at a service-oriented network architecture. This means, applications are supposed to be installed on a vehicle without having to physically visit a service point or the manufacturer of the vehicle. In the future, this will take place through a so-called App Store on the vehicle. Similar to smartphones, this includes a pool of applications to extend the functionality of a car. The ability of vehicle hardware extension through Plug&Play also implicates the need of a network reconfiguration from a single unit accessible to all components within the system. In order to fully integrate and support the hardware functionality into the new network from a software perspective, data streams, port rules and additional services must be configured and provided for the desired functionality. The A3F project follows the concept of software-defined networking, similar to what is known from software-defined networks[4] (SDN) or Software Oriented Architecture[5] (SOA). This are concepts about building a service-oriented architecture managed from a central location. This principle can be applied to the desired software extension of the system. However, this approach also involves risks. The number of different configurations possible in the system increases due to software extensions and different applications. For example, 10 services were available to extend a system with 5 server units, there were already 1001 possible combinations.

$$\frac{(n+m-1)!}{((n-1)!*m!)} = \frac{(5+10-1)!}{((5-1)!*10!)} = 1001$$

[4]https://ieeexplore.ieee.org/document/6994333

[5]https://ieeexplore.ieee.org/document/4026892

Since far more than these are planned for the future, system configuration can no longer be done for each service individually evaluated and checked for possible errors in the end. Using SDN's approach, configuration can be done from at least one central location. As of now, there are several approaches to ensure a verification and validation process, but no practicable solution for such complexity.

Another problem is represented by technologies like CAN, LIN and FlexRay mentioned earlier in this paper. In the future, these technologies will reach their limits in an attempt to meet the requirements as described for a new vehicle network. The data to be sent can raise up to several gigabit per second and must be sent in real time from a source Electronic Control Unit (ECU) to the destination ECU in a system. For communication within the car, therefore it is necessary to use a modern medium. Since 2010, several major automotive manufacturers have been working on integrating Ethernet into modern automotive architectures. However, Ethernet should not completely replace all bus systems but rather complement them in order to transfer large amounts of data between the devices as efficiently as possible. But not only the automobile manufacturers have researched for the integration of Ethernet, also several universities and key players in the global automotive industry, such as Continental, Bosch and Vector, are convinced of the necessity of Ethernet in the automobile (Steinbach et al., 2011)(Bello, 2011)(Eisele, 2018).

2 RELATED WORK

Over the last two years, the A3F research group mentioned in section I has developed a network management system which independently of components configures a vehicle network. The project is a cooperation between Continental Automotive GmbH and the University OTH Regensburg. The research group decided to define three expansion stages for in-vehicle networks in terms of autonomous driving. The last two states are called *Next Generation Networks*, possible network concepts are displayed in figure 1 on timeline starting in the year 2025.

- The first stage is to configure a network at the end of the manufacturing process line. This represents the today standard in automotive manufacturing.
- Secondly, the vehicle shall be reconfigured during functionalities have been added. In this extension the automatic process is used only if the vehicle is in a safe state. Looking forward, there is still a need for a precise definition of which state represents a safe state. An example for a save state

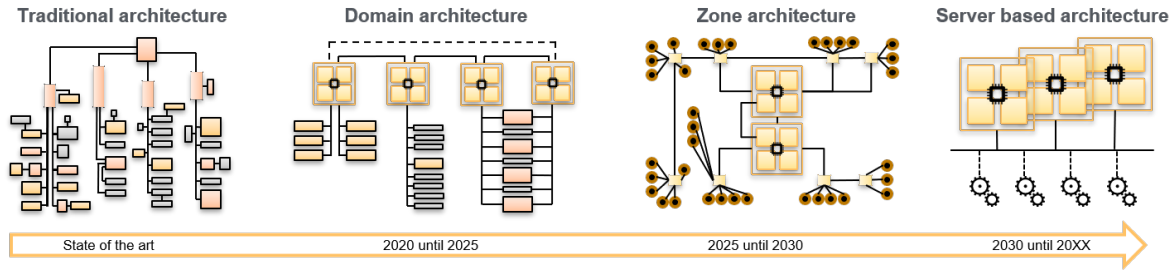

Figure 1: Future E/E Architecture (Carlson et al., 2019).

could be the idle state of a automobile, like parking position. Though, this topic is not going to be evaluated further in this paper.

- The final expansion stage represents the future of vehicle networks. Configuration can be carried out at any time. Thus, the functionality of a vehicle can be extended at any time. A conceivable scenario would be a new configuration already created and loaded in background in parallel to the current configuration. So, the new configuration is adopted into the network and its components but only approved after successful parallel operation. However, this requires more memory space and computing performance in the components.

Currently the project focuses on stage 2 where the vehicle has to reconfigure itself after changes in the system feature functionalities by adding a service or hardware. Figure 2 shows the concept of a management system which configured the network from a central unit in the system. The central icon on the illustration represents the network manager which is able to use external input such as network topology, message and service parameter from a manifest into a new configuration for the corresponding vehicle network. The automatically generated configuration should provide usage independent from the various communication network topologies, so that manual intervention is no longer necessary. The development of the central network management for the system was one of the A3F project outcomes. A considerable part of the network manager implementation is documented in two theses at the University Of Applied Sciences in Regensburg and can be consulted for more detailed information (Brunthaler, 2018)(Urban, 2018).

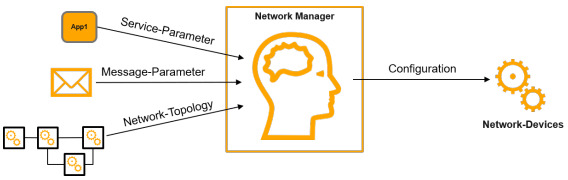

Figure 2: Central Networkmanager.

As mentioned earlier, work has been carried out in recent years on the automated configuration of a vehicle network using new vehicle components like servers. By means of a demonstrator structure a simulation can be realized. This structure has already been implemented in the last few months. In the laboratory setup, a huge part of the network can be extended and reconfigured with a variety of services without the need of external intervention. To underline this, some examples are the configuration of data streams and port rules at the respective switches or terminals, such as an ECU or sensor, as well as the provision of new services in the system. The next step to achieve the target of this project is to develop a process to verify and validate the automatically generated network configuration. To evaluate the process of automatically creating the configuration of a vehicle network, different approaches can be used for. These are examined for their feasibility and effort. The following table 1 shows the investigated topics.

Table 1: Concepts for verification and validation.

Technology	*Section*	*Source*
Independent Verification and Validation	2.1	(Open Alliance, 2017)(techopedia, 2019)
Code Validation concepts	2.2	IEEE(IEEE, 2012)(IEEEa, 2017)
System life cycle Processes	2.2	ISO(ISO, 2015)
Classification in test category	2.3	University of Prague(Sobotka, 2017)
Automatic and Simulation In-the-loop Test	2.4	(Bock et al., 2008)
Mathematical proof of algorithm	2.5	(Ziegenbalg et al., 2016)

In the following sections, the work on the topics listed in table 1 are explained in further detail. Only the points for which broad searches were necessary are listed.

2.1 Independent Verification and Validation

Independent Verification And Validation (IV&V) fully incorporates the concept of Verification And Validation (V&V). The difference to classic V&V is that IV&V is performed by a third party company not involved in development. In this way, software and hardware can be tested by third parties. The main advantage of independent testing is the accurate verification of the opportunity to accurately verify the spec-

ifications and functionality of the product to be tested the accurate verification of the opportunity to accurately verify the.(techopedia, 2019)(IEEEb, 2017)

In the A3F project Independent Software Verification And Validation (ISV&V) which is part of IV&V plays a major role. This is usually done according to previously given methods. In this case, the IV&V concept could look like this:

- ISV&V Concept Planning
  - System Criticality Analysis
  - Identification of Critical Components
  - Selection of suitable methods and audit tools
- Specification of Requirements
  - Verification for: Completeness, Correctness, Testability
- Verification of the Design
  - Verification of design and conformity between software requirements and interfaces
  - Feasibility check, as well as internal and external consistency
- Verification of the Code
  - Analysis of code metrics
  - Compliance with code guidelines and conformity testing
  - Completeness, Correctness, Consistency
- Validation
  - Identification of faulty or error-prone components/functions
  - Compliance with software and system requirements
  - Black- and white box testing

Within the A3F project and the automotive industry, one of the best known institutions for independent testing is the Open Alliance with their definition of the TC1 to TC14 Technical Comittees. The Open Alliance is a group of several companies interested in the standardization and testing Ethernet technologie in modern vehicle networks. Therefore, new test specifications are constantly being developed to ensure uniform protection of the new technology. These tests are also defined by test models such as IV&V. One of the specifications is the TC8. This document contains several tests for electronic vehicle controls which must be passed prior to installing the components on the vehicle. The tests in TC8 are located on the ISO/OSI layers one, two and three, known as the data link and network layer. The tests are usually done by independent test institutes according to the specifications of the Open Alliance.(Open Alliance, 2017)

2.2 System Life Cycle Processes and Code Validation

The concept of life cycle process is known from other IT areas and software development. In general, this demonstrates the applications V-cycle principle which is shown in figure 3. This principle is based on the constant monitoring of individual development stages. In contrast to TC8, this is not only a final check but a permanent one. This type of verification and validation should be saved for the development process of the next generation vehicle network, as it allows an early detection of errors and malfunctions in the system or software. Also, processes for code validation should be integrated at this point. That way, uniform standards can be guaranteed during development. The recommended way of implementing this is using of IEEE standardization (IEEE, 2012)(IEEEa, 2017), code guides and software patterns. The main advantages of this are that errors can be found more easily, as well as clearer and comprehensive code.

2.3 Define Classification in Test Categories

To ensure that the process of automated configuration creation delivers accurate results, not only the initial generated configuration, but also the algorithms have to be tested on its own. To achieve this, an inventory of the test metrics for this type of system must be made. One possibility is the *methods contained in the publication to verify and validate distributed automotive systems*(Sobotka, 2017) released test concept to define a classification for the individual elements. In Fig. 4 a model of the classification is represented graphically. Thus, the implementation and execution of tests can be guaranteed for all components which were generated during configuration.

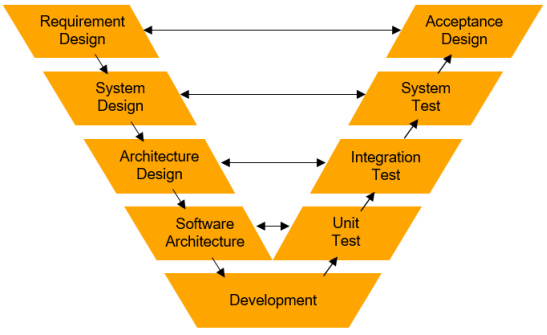

Figure 3: V-Cycle of development.

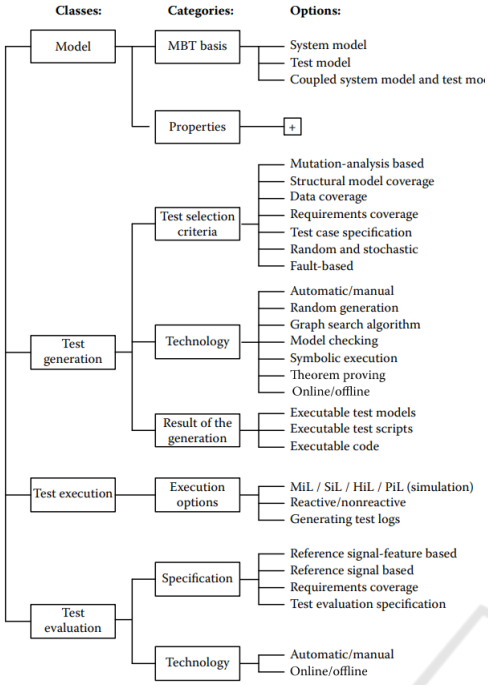

Figure 4: Classification for testing (Sobotka, 2017).

2.4 Automatic and Simulation In-the-loop Test

In order to cope with the growing complexity, the process of verification and validation must be automated to the extent possible. The concept Simulation In-the-loop test is limited to the application layer in the ISO/OSI[6] model, primarily. Firstly, simulations and tests are performed for the individual services to ensure they operate smoothly. This takes place during development by testing the source code before it gets uploaded in the so-called appstore. In the next step, automated tests are designed in the runtime environment of an application. Here, the complexity mentioned above plays a big role because not only the system itself grows but also the simulations from the in-the-loop tests quickly become very complex. Therefore, they cannot be tested directly in the vehicle in the required time due to the high number of possible combinations of the various services already discussed. Hence, tests must be carried out automatically in advance and their results must be available in order to compare them with the configuration created. For this purpose, test tools such as Jenkins[7], Ranorex[8], Gating[9] and Tsung[10] can be used. These are well suited for the development process of various applications. The application level testing approach appears promising as it is already used in related areas such as Internet of Things (IoT)(Murad et al., 2018) and Cloud Computing. In Cloud Computing, validation is performed at the level of cloud services, so every service is tested before it is deployed on a system (Sehgal and Bhatth, 2018).

One way to check the configuration is to transfer an automatically generated configuration of the vehicle network to a powerful backend which is able to run the tests or compare with existing results.This approach could be realized using Over-The-Air Technology (OTA), for instance. In case of a positive evaluation, the configuration can be adopted in the network. An example communication between vehicle and backend is shown in figure 5. This procedure is time-consuming and requires a powerful network and backend, which is associated with high costs. In recent times network expansions used to be a big topic. The 5G technology mentioned here could provide the bandwidth necessary for the network. This would enable the approach of transmission to a powerful backend, in general (Andrews et al., 2014).

2.5 Mathematical Proof of Algorithm

Under the assumption that a computing unit such as an ECU is implemented correctly and provides correct results, it is still very difficult to verify the accuracy of the algorithm for automated configuration generation. Ideally, a theoretical proof for the correctness of this algorithm should be found. For this purpose, each sub-function of the automatic configuration algorithm is checked separately. This procedure is called a partial verification of correctness. The next step is to check the entire algorithm for absolute correctness (Ziegenbalg et al., 2016). Since the full complexity of the network management functionality is not yet known, a mathematical proof approach will be difficult. Also disturbances by other network participants are not yet excluded by a formal proof and must be checked beforehand. Therefore, it is clear that the mathematical proof is one of the most complex operations for such an algorithm. One option is the implementation of mathematical and software checks to test the algorithm itself using the result as a complete induction verification of the functionality. A second possibility is a comprehensive brute force test of the network. This way, possible anomalies can be found after a system reconfiguration. This test examines a large number of cases in which errors can be

[6]https://ieeexplore.ieee.org/document/1094702
[7]https://jenkins.io/
[8]https://www.ranorex.com
[9]https://gatling.io/
[10]http://tsung.erlang-projects.org/

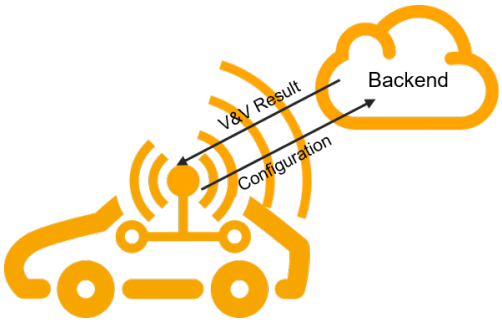

Figure 5: Backend communication.

detected. Hence, the chance of detecting all errors is high but there is no guarantee.

3 CHALLENGES OF TESTING DYNAMIC NETWORKS

The greatest difficulty in testing the next generation network lies in the variety of possible combinations of the individual software and hardware components that can be found in the system. Due to the modular extension of a network, there are practically no limits to the complexity of a system. This is a fundamental challenge for the approach of automated network configuration at a central point in the network. However, the real challenge for the future is the validation and verification of the created configuration.

3.1 Closely Coupling among Software, Hardware and Network

Due to the different dependencies between the individual components, these must be solved first in order to run specific tests for application and device level in the network. Further, before testing the functionality of the overall system, the individual levels have to be evaluated. Here, too, automation needs to be applied to testing.

3.2 Communication Paths

As the network grows, so does the number of possible communication channels in the system through which applications can exchange data. New account points such as switches are emerging which need to be configured to transfer data quickly and correctly. In addition, mechanisms such as redundancy can create complex problems for system verification. Checking all these paths and their devices is a difficult task.

3.3 Automated Verification and Validation

Automation is one of the most important issues in the verification and validation of a new network. Only by automating tests, it is possible to check the various configurations. For this purpose, tools must be found that not only test software automatically, but also fulfill test requirements such as the IV&V described in section 2.1. This will be most likely an extension of an existing test tool or a combination of several.

3.4 Security

Protection against unauthorized and unwanted modification or manipulation of an automatically created network configuration also plays an important role in verification and validation. If a configuration has been unauthorizedly modified from outside this should also be detected by software testing in the system. With this, the system should be protected internally and against external manipulation.

3.5 Safety

Not only security but also safety is an important aspect. In order to be able to determine exactly which danger emanates from the new system, an assessment must first be made according to the ASIL levels. These define the dangers for the environment of the vehicle. In the paper *Sicherheit von autonomem Fahren* the different driving situations are considered on the basis of ASIL and grouped into the different subclasses (Wirth and Reif, 2018). Based on the levels, various safeguards must be created which will also have an influence on the network, such as redundant and real-time communication. The term safety essentially contains two components, functional safety and safety of the intended functionality (SOTIF). Safety and security in combination are seen as a unit who's requirements needs to be met by the sytem at all times and must be fulfilled by the system.(Schnieder and Hosse, 2019)

4 PROMISING APPROACHES

All the approaches presented are important for verification and validation of the next generation vehicle network. Nevertheless, in the research area of Group A3F, a prioritisation has emerged which will be pursued further in the coming months. Primarily, this will be a combination of Independent Verification and Validation (IV&V) 2.1 and Software defined Network

and Architecture. With service-oriented architecture and testing, this forms a promising verification and validation approach.

5 WHERE TO NEXT

The next steps in this research project are the extension of the table to include concepts from other areas of corporate IT, as well as the detailed elaboration of the concepts described in promising approaches 4 concepts. After the complete elaboration of the most promising approaches, feasibility studies have to be deployed. These have yet to be defined and implemented. It would be conceivable to do feasibility studies on the existing A3F project demonstrator in order to obtain well-founded test results. Currently, the biggest problem with the system is the complexity of the new vehicle network which will be difficult to estimate. One of the described process for testing and designing the verification and validation concept is not sufficient. Therefore, a combination of several procedures must be put together to solve this problem.

REFERENCES

Ing. Jan Sobotka, *Methods for Verification and Validation of Automotive Distributed Systems*, Czech Technical University in Prague, August 2017.

Till Steinbach; Franz Korf; Thomas C. Schmidt, *Real-time Ethernet for Automotive Applications: A Solution for Future In-Car Networks*, Hamburg University of Applied Sciences, Germany, IEEE 2011.

Lucia Lo Bello, *The case for Ethernet in Automotive Communications*, University of Catania, Italy, 2011.

IEEE, *Inc. Standard for System and Software Verification and Validation*, IEEE 1012-2012, edition 2012.

IEEE, *Inc. Standard for System, Software and Hardware Verification and Validation*, IEEE 1012-2016, edition 2017.

IEEE, *Applying Standard Independent Verification and Validation (IV&V) Techniques within an Agile Framework: Is there a Compatibility Issue?*, 2017 Annual IEEE International Systems Conference (SysCon)

Open Alliance, *OPEN Alliance Automotive Ethernet ECU Test Specification*, Open Alliance, v2.0 edition 2017.

ISO, *Systems and software engineering – System life cycle processes*, ISO/IEC/IEEE 15288, edition 2015.

Sebastian Brunthaler, *Erweiterung eines Netzwerkmanager zur automatischen Konfiguration der sich im Fahrzeug befindenden Netzwerkgeräte anhand verschiedener Applikationsparameter*, OTH Regensburg, 2018.

Daniel Urban, *Untersuchung von Methoden zur Einbindung von virtuellen Umgebungen in ein zukünftiges Ethernet-basiertes Fahrzeugnetzwerk*, OTH Regensburg, 2018.

Jochen Ziegenbalg; Oliver Ziegenbalg; Bernd Ziegenbalg, *Korrektheit von Algorithmen Korrektheit von Computerergebnissen*, Algorithmen von Hammurapi bis Gödel, 2016.

Johannes Eisele, *Zwei Drittel der Weltbevölkerung werden 2050 in Städten leben*, Zeit Online 2018.

Thomas Bock; Markus Maurer; Franciscus Meel; Thomas Müller, *Vehicle in the Loop. In: ATZ Automobiltech Z 110 (1)*, S. 10–16. DOI: 10.1007/BF03221943.

G. Murad; A. Badarneh; A. Quscf; F. Almasalha, *Software Testing Techniques in IoT*, S. 17-24. 2018 8th International Conference on Computer Science and Information Technology (CSIT), Amman 2018

Dr. Lars Schnieder; René S. Hosse, *Leitfaden Safety of the Intended Functionality*, Springer Vieweg, 2019

Techopedia, https://www.techopedia.com/definition/24836/independent-verification-and-validation–iv&v

Thomas Erl, *Service-Oriented Architecture. Concepts, Technology, and Design*, Prentice Hall PTR, Upper Saddle River 2004

Eike Björn Schweißguth, *Entwicklung und Evaluierung eines SDN-gestützten echtzeitfähigen Gerätenetzwerkes*, Springer Vieweg, 2016

Naresh Sehgal; Pramod Chandra P. Bhatth, *Cloud Computing, Concepts and Practices*, Springer International Publishing, 2018

Patrick Wirth; Monika Ulrike Reif, *Sicherheit von autonomem Fahren*, Winterthur : ZHAW Zürcher Hochschule für Angewandte Wissenschaften, 2018

Steve Carlson; Christopher Mash; Christoph Wechsler; Helge Zinner; Olaf Grau; Natalie Wienckowski, *Automotive Ethernet: Beyond 10 Gb/s Electrical PHYs call for interest*, IEEE 802 LMSC March 2019 Plenary meeting in Vancouver, BC, Canada, 2019

Jeffrey G. Andrews; Stefano Buzzi; Wan Choi; Stephen V. Hanly; Angel Lozano; Anthony C. K. Soong; Jianzhong C. Zhang, *What Will 5G Be?*, IEEE Journal on Selected Areas in Communications (IEEE Journal on Selected Areas in Communications), 2014