Testing Autonomous Systems

Tilo Linz

Abstract The development of autonomous vehicles is currently being promoted massively, not least in the German automotive industry, under very high investments. The railway industry, shipbuilding, aircraft industry, and robot construction are also working on further developing their products (trains, ships, drones, robots, etc.) into self-driving or autonomous systems.

This chapter therefore discusses the question in which aspects the testing of future autonomous systems will differ from the testing of software-based systems of today's character and gives some suggestions for the corresponding further development of the test procedure.

Keywords Software testing · Software quality · Autonomous vehicles · Autonomous systems

1 Motivation

The development of autonomous vehicles is currently being promoted massively, not least in the German automotive industry, under very high investments. The railway industry, shipbuilding, aircraft industry, and robot construction are also working on further developing their products (trains, ships, drones, robots, etc.) into self-driving or autonomous systems.

The world's leading research and advisory company Gartner provides the following assessment in its report *Top 10 Strategic Technology Trends for 2019: Autonomous Things* [1]:

- By 2023, over 30% of operational warehouse workers will be supplemented by collaborative robots.

T. Linz
imbus AG, Möhrendorf, Germany

S. Goericke (ed.), *The Future of Software Quality Assurance*,
https://doi.org/10.1007/978-3-030-29509-7_5

- By 2025, more than 12% of newly produced vehicles will have autonomous driving hardware capability of Level 3 or higher of the SAE International Standard J3016.[1]
- By 2022, 40 of the world's 50 largest economies will permit routinely operated autonomous drone flights, up from none in 2018.

It can be assumed that within the next 10 years mobile systems will conquer the public space and be autonomously (or at least partially autonomously) "on the way" there.

The degree of autonomy of these systems depends on whether and how quickly manufacturers succeed in equipping their respective products with the sensors and artificial intelligence required for autonomous behavior.

The major challenge here is to ensure that these systems are sufficiently safe and that they are designed in such a way that they can be approved for use in public spaces (road traffic, airspace, waterways). The admissibility of the emerging systems and their fundamental social acceptance depend on whether the potential hazards to humans, animals, and property posed by such systems can be minimized and limited to an acceptable level.

Consensus must be reached on suitable approval criteria and existing approval procedures must be supplemented or new ones developed and adopted. Regardless of what the approval procedures will look like in detail, manufacturers will have to prove that their own products meet the approval criteria.

The systematic and risk-adequate testing of such products will play an important role in this context. Both the Expert Group on Artificial Intelligence of the European Commission and the Ethics Commission "Automated and Networked Driving" set up by the German Federal Minister of Transport and Digital Infrastructure explicitly formulate corresponding requirements for testing in their guidelines [3, 4].

This chapter therefore discusses the question in which aspects the testing of future autonomous systems will differ from the testing of software-based systems of today's character and gives some suggestions for the corresponding further development of the test procedure.

2 Autonomous Systems

We understand the term "Autonomous System" in this chapter as a generic term for the most diverse forms of vehicles, means of transport, robots, or devices that are capable of moving in space in a self-controlling manner – without direct human intervention.

An older term for such systems is "Unmanned System (UMS)" [5]. The term emphasizes the contrast with conventional systems that require a driver or pilot on board and also includes nonautonomous, remote-controlled systems.

The modern term is "Autonomous Things (AuT)" [6]. This term is based on the term "Internet of Things (IoT)" and thus conveys the aspects that Autonomous

[1] See [2].

Systems can be networked with each other and with IT systems on the internet, but also the development towards (physically) ever smaller autonomous things.

Examples of[2] such systems are:

- Motor vehicles (cars, lorries) which partially or (in the future) completely take over the function of the driver[3]
- Driverless transport vehicles that are used, for example, for logistics tasks and/or in production facilities[4]
- Ocean-going vessels, boats, inland waterway vessels, and other watercrafts[5] which are used, for example, for the transport of goods
- Driverless underwater vehicles or underwater robots which, for example, carry out[6] inspection or repair tasks under water independently
- Driverless trains, suburban trains, underground trains, or train systems for passenger or freight transport[7]
- Unmanned or pilotless aircrafts, helicopters, or drones[8]
- Mobile robots, walking robots, humanoid robots that are used for assembly, transport, rescue, or assistance tasks[9]
- Mobile service or household robots, for example, automatic lawn mowers or vacuum cleaners, which carry out service work in the household[10] and communicate with the "Smart Home" if necessary

Although all these systems are very different, they share some common characteristics:

- These are cyber-physical systems, that is, they consist of a combination of "informatic, software-technical components with mechanical and electronic parts."[11]
- They are mobile within their operational environment, that is, they can control their movements themselves and navigate independently (target-oriented or task-oriented).

[2]The listed examples name civil areas of application. However, the development of autonomous systems and corresponding technologies has been and continues to be strongly motivated and financed also because of their potential applications in the military sector.

[3] https://en.wikipedia.org/wiki/Autonomous_car, https://de.wikipedia.org/wiki/Autonomes_Fahren

[4] https://en.wikipedia.org/wiki/Automated_guided_vehicle, https://de.wikipedia.org/wiki/Fahrerloses_Transportfahrzeug

[5] https://en.wikipedia.org/wiki/Autonomous_cargo_ship, https://en.wikipedia.org/wiki/Unmanned_surface_vehicle

[6]https://en.wikipedia.org/wiki/Autonomous_underwater_vehicle

[7]https://en.wikipedia.org/wiki/Automatic_train_operation, https://en.wikipedia.org/wiki/List_of_automated_train_systems

[8]https://en.wikipedia.org/wiki/Unmanned_aerial_vehicle

[9]https://en.wikipedia.org/wiki/Robot#General-purpose_autonomous_robots, https://en.wikipedia.org/wiki/Autonomous_robot, https://en.wikipedia.org/wiki/Legged_robot, https://en.wikipedia.org/wiki/Humanoid_robot

[10]https://en.wikipedia.org/wiki/Service_robot

[11]https://en.wikipedia.org/wiki/Cyber-physical_system

- They can perform a specific task (e.g., mowing the lawn) or head for a specific destination (e.g., "drive to Hamburg") without having to specify the details of the task or the exact route in advance.

2.1 Autonomy and Autonomy Levels

"Autonomy" (of an UMS) is defined in [5] as: "A UMS's own ability of integrated sensing, perceiving, analyzing, communicating, planning, decision-making, and acting/executing, to achieve its goals as assigned by its human operator(s) through designed Human-Robot Interface (HRI) or by another system that the UMS communicates with."

The degree to which an autonomous system fulfils these properties (*sensing, perceiving, analyzing*, etc.) can be very different. In order to be able to classify systems according to their degree of autonomy, various classification systems were defined.

A well-known scale of this kind is the classification of autonomy levels for autonomous driving according to SAE Standard J3016 (see [2]). The following table is a simplified representation of these levels based on [7]:

SAE level	Name	Description	Control	Environment observation	Fallback
0	No automation	The driver drives independently, even if supporting systems are available.	Driver	Driver	–
1	Driver assistance	Driver assistance systems assist in vehicle operation during longitudinal or lateral steering.	Driver and system	Driver	Driver
2	Partial automation	One or more driver assistance systems assist in vehicle operation during longitudinal and simultaneous lateral control	System	Driver	Driver
3	Conditional automation	Autonomous driving with the expectation that the driver must react to a request for intervention.	System	System	Driver
4	High automation	Automated guidance of the vehicle without the expectation that the driver will react to a request for intervention. Without any human reaction, the vehicle continues to steer autonomously.	System	System	System
5	Full automation	Completely autonomous driving, in which the dynamic driving task is performed under any road surface and environmental condition, which is also controlled by a human driver.	System	System	System

The SAE levels are structured according to the division of tasks between driver and vehicle.[12] For robots and other basically driverless, autonomous systems, a more general definition is needed. [5] defines a generic framework for "Autonomy Levels for Unmanned Systems (ALFUS)" that is applicable to all types of UMS or autonomous systems with three assessment dimensions:

1. Mission Complexity (MC)
2. Environmental Complexity (EC)
3. Human Independence (HI)

The framework describes how a metric-based classification can be performed within each of these dimensions and how an overall system rating ("Contextual Autonomous Capability") can be determined from this.

2.2 Capabilities of Fully Autonomous Systems

A fully autonomous system should be able to accomplish a predetermined mission goal without human intervention. For a service robot, one such goal could be "get me a bottle of water from the kitchen." A fully autonomous car should be able to drive its passengers "to Hamburg."

The system must be able to navigate autonomously in its respective environment. And it must be able to detect previously unknown or ad hoc obstacles and then avoid them (e.g., by an autonomous vehicle recognizing a blocked road and then bypassing it), or remove them (e.g., by a service robot opening the closed door that blocks the way to the kitchen).

In more general terms, this means that a fully autonomous system must be able to recognize and interpret situations or events within a certain spatial and temporal radius. In the context of the identified situation, it must be able to evaluate possible options for action and select the appropriate or best option with regard to the mission objective and then implement it as measures.

3 Safety of Autonomous Systems

It is obvious that a self-driving car or autonomous robot poses a danger to people, animals, objects, and infrastructure in its vicinity. Depending on the mass and movement speed of the system (or of system parts, e.g., a robotic gripping arm), the danger can be considerable or fatal. Possible hazard categories are:

[12] [2] itself avoids the term "autonomous" because "... in jurisprudence, autonomy refers to the capacity for self-governance. In this sense, also, 'autonomous' is a misnomer as applied to automated driving technology, because even the most advanced ADSs are not 'self-governing' .... For these reasons, this document does not use the popular term 'autonomous' to describe driving automation."

- Infringement of uninvolved third parties by the autonomously moving system
- The violation of direct users, operators, or passengers of the autonomous system
- Injury to animals or damage to objects or infrastructure in the track or operating radius of the system by the system
- Damage to other objects caused by objects that the system handles or has handled
- Damage to the system itself, for example, due to a maneuvering error

Since human intervention may take place too late in a dangerous situation or (for systems with a high autonomy level) is not planned at all, the autonomous system itself must be sufficiently safe. In the overall life cycle of an autonomous system (from development to deployment to decommissioning), the topic of "safety" therefore has an extraordinarily high priority.

The associated safety levels (SIL levels) are defined in the series of standards [8]. The term "safety" is defined there as:

- Freedom from unacceptable risk of physical injury or of damage to the health of people, either directly, or indirectly as a result of damage to property or to the environment. [9].

To ensure sufficient safety, a system must have "functional safety":

- Functional safety is the part of the overall safety that depends on a system or equipment operating correctly in response to its inputs. Functional safety is the detection of a potentially dangerous condition resulting in the activation of a protective or corrective device or mechanism to prevent hazardous events arising or providing mitigation to reduce the consequence of the hazardous event ...
- ... The aim of Functional safety is to bring risk down to a tolerable level and to reduce its negative impact. [9].

3.1 Safety in Normal Operation

The dangers described above primarily result from the movement of the system or system components (e.g., a gripping arm). The level of danger or the associated risk of damage depends on the speed and mass of the system and the complexity and variability of its environment (Environmental Complexity). The following examples illustrate this:

- With a semi-autonomous, automatic lawn mower, the area to be mown is bordered, for example, by a signal wire. The movement space garden is a controlled environment. The robot's movement speed and movement energy are low. Contact-based collision detection is sufficient for obstacle detection. The risk posed by the rotating cutting knife is protected to an acceptable level (for operation within the controlled environment) by the housing and by sensors which detect lifting of the robot or blocking of the knife.
- For a fully autonomous car, the range of motion is open. Motion speed and kinetic energy can be very high. The car moves simultaneously to many other road users in a confined space. Obstacles of any kind can "appear" in the route at any time. Evasion is a necessary part of "normal operation." For safe driving in compliance

with traffic regulations, extremely reliable, fast, predictive obstacle detection is required.

When a robot interacts with objects, damage can also be caused indirectly (in addition to the danger of damaging the object or robot). The following examples from [10, p.77] illustrate this:

- A service robot is instructed to bring the dishes to the kitchen sink. In order to deposit the dishes near to the sink, it recognizes the modern ceramic stove top as preferable surface and deposits the dishes there ... If now a cooking plate is still hot, and there is, for instance, a plastic salad bowl, or a cutting board amongst the dishes, obviously, some risks arise. The situation in which a plastic or wooden object is located very close or on top of the cooking plate can be considered as not safe anymore, since the risk of toxic vapor or fire by inflamed plastic or wood is potentially present.

  The worst case accident can be a residential fire causing human injury or death. The risk is not present in a situation in which these objects are located apart the cooking plate (with a certain safety margin), independent from the state of the cooking plate.
- A service robot is instructed to "watering the plants." In this connection, it is assumed that a power plug fell into a plant pot ... If the robot is watering the plant, the risk of electrical shock arises, both, for human and robot. The risk factors can be considered to be the following: The object recognition again recognizes the power plug while having the watering can grasped (or any plant watering device) and additionally, it can be detected that there is water in the watering can (or similar device). In consequence, a rule should be integrated that instructs the robot not to approaching too close with the watering can to a power plug, or the like, in order to avoid that it is struck by a water jet.

In order to be functionally safe, a highly or fully autonomous system must therefore have appropriate capabilities and strategies to identify situations as potentially dangerous and then respond appropriately to the situation in order to avoid imminent danger or minimize[13] damage as far as possible. The examples *cooking plate* and *watering the plants* make it clear that pure obstacle detection alone is not always sufficient. In complex operational environments with complex possible missions of the autonomous system, some dangers can only be recognized if a certain "understanding" of cause-effect relationships is given.

Such capabilities and strategies must be part of the "intelligence" of highly autonomous systems. The intended system functionality and the necessary safety functions cannot be implemented separately, but are two sides of the same coin.

3.2 Safety in Failure Mode

If parts of the autonomous system fail, become damaged, or do not function as intended (because of hardware faults, such as contamination or defect of a sensor),

[13]The media in this context mainly discuss variants of the so-called "trolley problem", that is, the question of whether and how an intelligent vehicle should weigh the injury or death of one person or group of persons at the expense of another person or group of persons in order to minimize the consequences of an unavoidable accident (see [11]).

the danger that the system causes damage is naturally even greater than in normal operation.

If a (rare) environmental situation occurs that is "not intended" by the software or that causes a software defect that has hitherto remained undetected in the system to take effect, this can transform an inherently harmless situation into a dangerous one and/or render existing safety functions ineffective.

With conventional, nonautonomous safety-critical systems, sufficiently safe behavior can usually be achieved by a "fail-safe" strategy. This means that the system is designed in such a way that in the event of a technical fault, the system is switched off or its operation is stopped, thereby greatly reducing or eliminating immediate danger (to the user or the environment).

This approach is not sufficient for autonomous systems! If a self-driving car would stop "in the middle of the road" in the event of a failure of an important sensor, the car would increase the danger it poses instead of reducing it. Autonomous systems should therefore have appropriate "fail-operational" capabilities (see [12]). A self-driving car should act as a human driver would: pilot to the side of the road, park there, and notify the breakdown service.

4 Testing Autonomous Systems

In which points does the testing of autonomous systems differ from the testing of software-based systems of today's character? To answer this, we consider the following subquestions:

- Which test topics need to be covered?
- What new testing methods are needed?
- Which requirements for the test process become more stringent?

4.1 Quality Characteristics and Test Topics

The objective of testing is to create confidence that a product meets the requirements of its stakeholders (customers, manufacturers, legislator, etc.). "Those stakeholders' needs (functionality, performance, security, maintainability, etc.) are precisely what is represented in the quality model, which categorizes the product quality into characteristics and sub-characteristics." [13]. This ISO 25010 [13] product quality model distinguishes between the following eight quality characteristics: Functional Suitability, Performance Efficiency, Compatibility, Usability, Reliability, Security, Maintainability, and Portability.

These quality characteristics can be used as a starting point when creating a test plan or test case catalog for testing an autonomous system. Within each of these quality characteristics, of course, it must be analyzed individually which specific

requirements the system to be tested should meet and what should therefore be checked in detail by test cases.

Utilizing this approach a test plan for the mobile robot "Mobipick" [14] of the German Research Center for Artificial Intelligence (DFKI) was created in 2018 as part of a cooperation project between imbus AG and DFKI. The test contents were recorded in the cloud-based test management system [15] and made available to the DFKI scientists and the project team. The following list references this case study to illustrate which topics and questions need to be considered when testing an autonomous system:

- *Functional Suitability*: It must be checked if the functional properties of the system are implemented "complete," "correct," and "appropriate." The functions of each individual component of the system are affected (at lower levels). At the highest level, the ability of the overall system to complete its mission shall be tested. The "Mobipick" test cases for example focuses on the functions "Navigation" and "Grabbing" and the resulting mission pattern: approach an object at a destination, grab it, pick it up and put it down at another location. Testing the functionality also must include testing the system's load limits! A restriction for gripping could be, for example, that the robot tips over in the case of heavy objects or is deflected from its direction of travel. Such boundary cases and the system behavior in such boundary cases must also be considered and examined.
- *Performance Efficiency*: The time behavior of the system and its components and the consumption of resources must be checked.
  - Possible questions regarding time behavior are: is the exercise of a function (e.g., obstacle detection) or mission (object approach, grab, and pick up) expected in a certain time period or with a certain (min/max) speed?
  - Possible tests regarding resource consumption (e.g., battery power) are: run longest application scenario on full battery to check range of battery; start mission on low battery to check out of energy behavior; start mission on low battery at different distances to charging station to check station location and estimate power consumption to station.
- *Compatibility*: This concerns the interoperability between components of the system itself (sensors, controls, actuators) as well as compatibility with external systems. Possible questions are: Can the control software, which was brought to the robot initially or after an update, take over sensor data, process it and control actuators correctly? Are the protocols for communication compatible between robot components or with external systems?
- *Usability*: What possibilities does the user have to operate the robot or to communicate with it? How are orders given to the robot? Are there any feedback messages in the event of operating errors and failure to understand the command? How does the robot communicate its status? Which channels and media are used for transmission: via touch panel on the robot, via app over WLAN, or via voice control? This also includes the handling of objects: can the robot hand over a gripped object to its user precisely enough?

- *Reliability*: Reliability is the ability of the system to maintain its once achieved quality level under certain conditions over a fixed period of time. Test topics can be: Can the robot repeat a behavior several times in a row without errors, or do joints misalign in continuous operation? Can the robot tolerate/compensate (hardware) errors to a certain degree?
- *Security*: To check how resistant the system is against unwanted access or criminal attack on data of the system or its users or on the entire system itself. Questions can be:
  - Does the operator need a password to switch on? How secure is this? With autonomous robots such as "Mobipick," the highest security risk arises from the control mode. The easier it is to manipulate the commands given to the system, the easier it is to (maliciously) take over or shut down the system. Is the robot operated via WLAN/radio? Is the data exchange with the system and within the system encrypted? Can third parties read along, possibly latch into the data traffic and manipulate or even take over the system? The unauthorized takeover of an autonomous system can have serious consequences, in extreme cases its use as a weapon. Therefore, security features are always safety-relevant features!
  - In order to be able to clarify liability issues in the event of an accident, legislators already require autonomous vehicles to record usage data during operation. In Germany these must be kept available for 6 months (see [16]). Similar requirements are expected for other autonomous systems. The GDPR-compliant data security of the system, but also associated (cloud based) accounting or management systems, is therefore another important issue.
- *Maintainability*: A good maintainability is given if software and hardware are modular and the respective components are reusable and easily changeable. Questions in this context are: how are dependencies between software and hardware managed? Does the software recognize which hardware it needs? How do the update mechanisms work? Is it defined which regression tests are to be performed after changes?
- *Portability*: At first glance, the software of robots can be transferred to other robot types to a very limited extent because it is strongly adapted to the specific conditions of the hardware platform and the respective firmware.
  - Individual software components (e.g., for navigation), on the other hand, are generic or based on libraries. It must be tested whether the libraries used in the concrete robot (e.g., "Mobipick") actually work faultlessly on this specific platform.
  - The autonomous system itself can also be "ported" or modified for use in other (than originally intended) environments. For example, by installing additional sensors and associated evaluation software.

The examples show how complex and time-consuming the testing of an autonomous system can be. An important finding is:

- "Functional safety" is not just a sub-item of "Functional Suitability"! Each of the eight quality characteristics from ISO 25010 [13] contains aspects which (especially if there are weaknesses) influence whether the system can be assessed as "functional safe." This is particularly true for the topic "Security."

4.2 Implications of Learning

The intelligence of highly autonomous systems will largely be based on learning algorithms (machine learning). Learning will not only be limited to the development phase of a system (learning system). From a certain Mission Complexity and Environmental Complexity on, it will be necessary for autonomous systems to learn from data they collect during normal operation (self-learning system) and thus continuously improve their behavior or adapt it for rare situations. This poses completely new questions to the development, testing, and approval of such systems:

> If robots are required to be able to learn, this reveals additional questions with regard to the problem to ensure safe robot behavior. Learning capabilities implicate that the learning system is changed by the learning process. Hence, the system behavior is not anymore determined by its initial (designed) structure, and not only structure deviations due to occurring faults are of interest anymore. Learning changes the systems structure; thus, its behavior can as well be determined by the newly learned aspects. The residual incompleteness of the safety-related knowledge consequence is that the system differs from its initially designed version. [10, p.131]

> The testing branch is facing new questions: how to test that a system is learning the right thing? How do test cases, which check that certain facts have been learned correctly, look like? How to test that a system correctly processes the learned knowledge by forgetting for example wrong or obsolete information or abstracting other information? How to test that (for example with robot cars) self-learning software follows specific ethic rules? How to formulate test strategies and test cases in such a way that they can handle the "fuzziness" of the behavior of AI systems? [17]

> With regard to the introduction of self-learning systems, the protection of users' physical integrity must be a top priority ... As long as there is no sufficient certainty that self-learning systems can correctly assess these situations or comply with safety requirements, decoupling of self-learning systems from safety-critical functions should be prescribed. The use of self-learning systems is therefore conceivable with the current state of the art only for functions that are not directly relevant to safety. [4]

4.3 New Test Method: Scenario-Based Testing

An autonomous system is characterized by the fact that it is capable of independently heading for and achieving a given mission goal. The subtasks that the system must solve for this can be formulated as test tasks and look as follows:

- Sensing: Can the system capture the signals and data relevant to its mission and occurring in its environment?

- Perceiving: Can it recognize patterns or situations based on signals and data?
- Analyzing: Can it identify options for action appropriate to the respective situation?
- Planning: Can it select the appropriate or best options for action?
- Acting: Can it implement the chosen action correctly and on time?

The systematic testing of this chain of tasks requires a catalogue of relevant situations that is as comprehensive as possible. These situations must be able to be varied in many parameters (analogous to different equivalence classes when testing classic IT systems): For example, the "Mobipick" service robot should be able to detect a closed door as an obstacle under different lighting conditions (daylight, bright sunlight, at night) and with different door materials (wooden door, glass door, metal door).

It must be possible to link the situations into scenarios (successive situations) in order to bring about specific situations in a targeted manner, in order to be able to examine alternative paths of action, but also in order to be able to examine the development over time for a specific situation and the timely, forward-looking action of the autonomous system.

Such testing of the behavior of a system in a sequence of situations is referred to as "Scenario-based Testing." [4] proposes "... to transfer relevant scenarios to a central scenario catalogue of a neutral authority in order to create corresponding generally valid specifications, including any acceptance tests.". The standardization of formats for the exchange of such scenarios is being worked on. ASAM OpenSCENARIO "... defines a file format for the description of the dynamic content of driving and traffic simulators .... The standard describes vehicle maneuvers in a storyboard, which is subdivided in stories, acts and sequences." [18].

Scenario-based testing requires that the same test procedure is repeated in a large number of variations of the test environment. When testing classic software or IT systems, however, the test environment is constant or limited to a few predefined variants. If the IT system successfully passes its tests in these environments, it can be considered suitable for use with low or acceptable risk.

If a robot or a self-driving car passes its tests in only one or a few test environments, the system may still be totally unsuitable for real operation, or even pose an extreme safety risk. When testing autonomous systems, the systematic variation of the test environment is therefore an essential and decisive part of the test strategy.

4.4 Requirements for the Test Process

The combination of "complex cyber-physical system" with "Mission Complexity" and "Environmental Complexity" leads to an astronomical number of potentially testable scenarios. Each of these scenarios, in turn, consists of situation sequences, with the possibility of variation in the respective system status, the environmental situation and the potential options for action of the system. Since safety requirements are not an isolated "subchapter of the test plan," but are present throughout all

scenarios, it is difficult and risky to reduce testing effort by prioritizing and omitting scenarios.

Testing only one such scenario in reality can require enormous effort (a secure test site is required, and changing the test setup and the subsequent repeated test drives in that site requires a lot of effort and time). A very large proportion of the necessary tests must and will therefore be carried out in the form of simulations.

Nevertheless, some of the scenarios will always have to take place additionally in reality. Because simulations can be error-prone and they usually will not be physically complete.

An important measure to gain time and safety is a consistent shift-left of tests to the lowest possible test levels and continuous testing during development at all test levels in parallel: at the level of each individual component, for each subsystem, and at the system level. Test-driven development and the formal verification of safety-critical components will play an increasingly important role. Continuous monitoring of the systems in operation ("shift-right") and, if necessary, quick reaction to problems in the field, will also be indispensable. In the *Ethics Guidelines for Trustworthy AI* of the European Commission corresponding demands are clearly formulated: "Testing and validation of the system should occur as early as possible, ensuring that the system behaves as intended throughout its entire life cycle and especially after deployment. It should include all components of an AI system, including data, pre-trained models, environments and the behaviour of the system as a whole." [3].

The test contents and test results of all test levels and the data from fleet operation must be continuously monitored, evaluated, and checked by test management in order to be able to identify gaps in the test coverage but also to reduce redundancies.

Significantly increased importance will be attached to testing by independent third parties. Here, too, [3] formulates proposals: "The testing processes should be designed and performed by an as diverse group of people as possible. Multiple metrics should be developed to cover the categories that are being tested for different perspectives. Adversarial testing by trusted and diverse 'red teams' deliberately attempting to 'break' the system to find vulnerabilities, and 'bug bounties' that incentivise outsiders to detect and responsibly report system errors and weaknesses, can be considered.".

5 Conclusion and Outlook

Procedures and best practices from the testing of classical software and IT systems, as well as from the field of conventional, safety-critical systems or vehicle components,[14] are also still valid for the testing of autonomous systems.

[14]ISO 26262:2018, "Road vehicles - Functional safety," is the ISO series of standards for safety-related electrical/electronic systems in motor vehicles.

A central question is how functional safety of autonomous systems can be guaranteed and tested. The intended system functionality and the necessary safety functions cannot be implemented separately, but are two sides of the same coin. Accordingly, it is not possible to separate the aspects of functionality and safety during testing.

Manufacturers of autonomous systems need procedures and tools by means of which they can test the functionality and safety of such products seamlessly, but nevertheless with economically justifiable effort, and prove them to the approval authorities.

One approach is Scenario-Based Testing. Scenarios can be used to model and describe usage situations and mission processes of an autonomous system. These scenarios can then be used as test instructions for testing in simulations or in reality.

In addition to the standardization of scenario formats or scenario languages, tools are needed to capture and manage scenarios. Integrations between such scenario editors, simulation tools, test benches, and test management tools need to be developed. Such tools or tool chains should also help to create scenario variants systematically and to evaluate scenarios and tests automatically, for example, with regard to safety relevance and achieved test coverage.

References[15]

1. Gartner “Top 10 Strategic Technology Trends for 2019: Autonomous Things”, Brian Burke, David Cearley, 13 March 2019
2. Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles, SAE - On-Road Automated Driving (ORAD) committee, https://saemobilus.sae.org/content/J3016_201806/
3. Independent High-Level Expert Group on Artificial Intelligence – set up by the European Commision, Ethics Guidelines for Trustworthy AI, 04/2019. https://ec.europa.eu/futurium/en/ai-alliance-consultation/guidelines
4. Automatisiertes und Vernetztes Fahren - Bericht-der-Ethik-Kommission, Bundesministerium für Verkehr und digitale Infrastruktur. https://www.bmvi.de/SharedDocs/DE/Publikationen/DG/bericht-der-ethik-kommission.html (2017)
5. Autonomy Levels for Unmanned Systems (ALFUS) Framework Volume I: Terminology, Version 2.0, Autonomy Levels for Unmanned Systems (ALFUS) Framework Volume II: Framework Models Version 1.0, https://www.nist.gov/el/intelligent-systems-division-73500/cognition-and-collaboration-systems/autonomy-levels-unmanned
6. https://en.wikipedia.org/wiki/Autonomous_things
7. https://de.wikipedia.org/wiki/SAE_J3016
8. IEC 61508:2010, Functional safety of electrical/electronic/programmable electronic safety-related systems - Parts 1 to 7, https://www.iec.ch/functionalsafety/
9. IEC 61508 Explained, https://www.iec.ch/functionalsafety/explained/
10. Safety of Autonomous Cognitive-oriented Robots, Philipp Ertle, Dissertation, Fakultät für Ingenieurwissenschaften, Abteilung Maschinenbau der Universität Duisburg-Essen (2013)

[15]The validity of the given URLs refers to July 2019.

11. Eimler, S., Geisler, S., Mischewski, P., Ethik im autonomen Fahrzeug: Zum menschlichen Verhalten in drohenden Unfallsituationen, Hochschule Ruhr West, veröffentlicht durch die Gesellschaft für Informatik e. V. 2018 in R. Dachselt, G. Weber (Hrsg.): Mensch und Computer 2018 – Workshopband, 02.–05. September 2018, Dresden
12. Temple, C., Vilela, A.: Fehlertolerante Systeme im Fahrzeug – von "fail-safe" zu "fail-operational". https://www.elektroniknet.de/elektronik-automotive/assistenzsysteme/fehlertolerante-systeme-im-fahrzeug-von-fail-safe-zu-fail-operational-110612.html (2014)
13. https://iso25000.com/index.php/en/iso-25000-standards/iso-25010
14. Deutsches Forschungszentrum für Künstliche Intelligenz GmbH (DFKI), Robotics Innovation Center, Robotersystem Mobipick. https://robotik.dfki-bremen.de/de/forschung/robotersysteme/mobipick.html
15. Cloudbasierten Testmanagementsystem der imbus AG. https://www.testbench.com
16. Deutscher Bundestag, Straßenverkehrsgesetz für automatisiertes Fahren, Drucksache 18/11776 vom 29.03.2017. https://www.bundestag.de/dokumente/textarchiv/2017/kw13-de-automatisiertes-fahren-499928, http://dip21.bundestag.de/dip21/btd/18/113/1811300.pdf
17. Flessner, B.: The Future of Testing, imbus Trend Study, 3rd edition. https://www.imbus.de/downloads/ (2017)
18. ASAM OpenSCENARIO. https://www.asam.net/standards/detail/opensce nario/

Further Reading

[ISO 10218:2011-07] Robots and robotic devices - Safety requirements for industrial robots, Part 1: Robots, Part 2: Robot systems and integration
[ISO 12100:2010] Safety of machinery - General principles for design - Risk assessment and risk reduction
[ISO 13482:2014] Robots and robotic devices - Safety requirements for personal care robots
[ISO 8373:2012-03] Robots and robotic devices – Vocabulary