25th ITS World Congress, Copenhagen, Denmark, 17-21 September 2018

Paper EU-TP1157

Verifying automated driving systems in simulation: framework and challenges

Zeyn Saigol[1*], Alan Peters[1]

1. Transport Systems Catapult, UK

[*] zeyn.saigol@ts.catapult.org.uk

Abstract

Autonomous vehicles are game-changers for the automotive industry. However, the automated driving system (ADS) that controls them will be extremely complex, and verifying its behaviour is a significant undertaking. The most promising approach currently is directed exhaustive testing – and given the size of the input space, such testing will have to be conducted mainly in simulation. We present a framework for a high-throughput, scenario-based, automated test system, and discuss the challenges involved in implementing such a system. As well as needing advances in the underlying computer science, particularly in sensor and environment modelling, there are many collaboration challenges, given the network of automotive and technology companies that are likely to play a part in interfacing an ADS to a simulated environment and evaluating its performance.

Keywords:

Autonomous driving systems, automation, simulation, verification.

1. Introduction

An Automated Driving System (ADS) must successfully achieve its driving task under every possible environment and traffic condition that falls within its operational design domain (ODD). These conditions include factors such as the type, location, and objectives of every nearby actor in the road scene, the road layout, traffic signals, weather, time-of-day, and many others. The complexity of both creating and verifying an ADS arises because not only are there an essentially infinite number of combinations of these factors, but the best action for the ADS to take may be different for each possible combination.

This complexity makes it almost impossible to write a complete specification for the behaviour of the ADS, which in turn means that traditional verification against a specification cannot easily be applied. Further, traditional automotive testing methods assume systems have a closed set of possible inputs, which does not map to the open environment that autonomous vehicles (AVs) must operate in. Given these, the obvious verification solution is extensive public-road testing; however, this turns out to be

impractical, with one calculation indicating that proving AVs have a fatality rate of <1.09 per 100 million miles (the human performance equivalent in the US) would need a fleet of 100 autonomous test vehicles driving 24 hours a day, 365 days a year for about 12.5 years [1]. Critically, any change to the ADS would require an assessment of the need to repeat some or all of these test miles.

The solution we are advocating here relies on extensive use of both scenario-based testing [2] and simulation [3,4]. Figure 1 illustrates how we see ADS verification fitting into the development V-cycle, showing that these activities are additional – in other words, it is still essential to verify that the mechanical components and non-ADS electronics all function correctly at all levels.

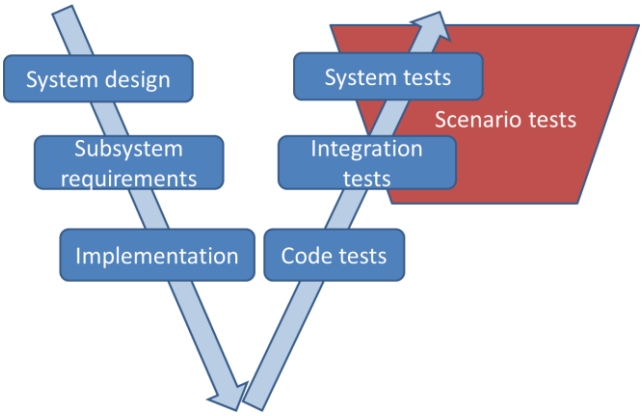

Figure 1 – V-model showing the role of scenario tests in the context of the systems engineering effort

Drawing on the Transport Systems Catapult's AV experience, which includes managing the first UK trial of full autonomy in a public place in 2016 [5,6], working closely with the BSI on strategic planning for AV standards [7], and exploring likely required AV testing scenarios [8], we conducted a simulation scoping study for the UK government at the end of 2017. This study examined the use of simulation to help verify AVs, and some of the outputs are presented here, in the form of a clear statement of the verification problems to be tackled and the shape of an integrated solution.

2. Current simulation tools, technologies, and processes

Vehicle manufacturers have been using simulations to develop and validate vehicle dynamics for many years, and such simulations represent a mature technology. Vehicle dynamics simulators model the physics of each component in a system, potentially including the engine, driveshaft, gearbox, suspension, tyres, chassis, and sub-elements of these. Similarly, the use of simulations for traffic modelling is well established. In this case, the aim is to answer questions such as identifying the best phase strategy for a set of traffic lights, or how to design a junction to maximise the throughput of traffic.

Despite these proven simulation tools, the increase in demand for ADAS systems and preparations for fully autonomous vehicles have been disruptive for the market. While traffic-level simulators can model road layouts, they do not require as much realism or detail as ADAS simulators do. And while vehicle dynamics simulators include some of the high-fidelity physics engines needed for ADAS and ADS testing, they only consider the vehicle itself and perhaps the road surface and gradient. They do not

model any of the road layout, traffic lights, street signs, other street furniture or surrounding infrastructure. Most critically, neither type of simulator models sensors such as cameras, radars or lidars (for example, Figure 2 shows a sensor field-of-view simulated in Gazebo[1], an open-source simulator developed for robotics research).

Ongoing developments in the field of automotive simulation tools include:

1. New entrants offering simulators specifically for ADAS and automated driving, with detailed sensor models as well as road/environment and vehicle dynamics simulation.
2. Traffic modelling tool vendors extending their products to encompass the behaviour of AVs.
3. Vehicle dynamics tools adding sensor simulation and road/environment modelling.
4. Significant efforts to connect the different types of simulators together, with commercial partnerships and/or technical interfaces, so that customers can get the benefits of all of them.
5. Some ADS developers creating a suitable simulator in-house, often starting from a games engine development framework. Such frameworks include all the elements of item (1).

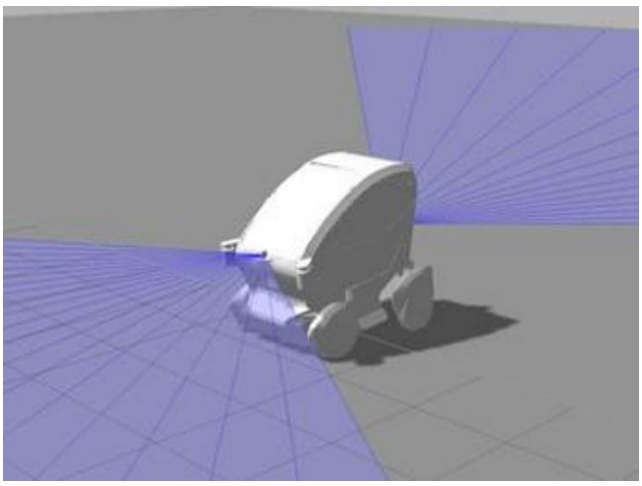

Figure 2: TSC's LUTZ pod as simulated in Gazebo

3. Links to existing functional safety process

The current mechanisms for maximizing road safety can be broadly divided into three topic areas:

- Vehicle design safety (ensuring the vehicle been designed and built correctly)
- Operational driving safety (is the vehicle driven correctly – a combination of training and enforcement)
- Environmental safety, (is the road infrastructure safe, how are risks outside vehicle addressed)

With the introduction of AVs, the ‘operational driving task’ merges with the ‘vehicle design task’ and becomes the responsibility of the system developer. Ensuring the performance of the operational driving

[1] http://gazebosim.org/

function, currently controlled via the mechanism of the driving test and law enforcement, becomes a technical challenge. Interestingly, to pass the current driving test is actually a relatively low pass criteria barrier (for the UK this is ~40 minutes in just one city or town in one set of environment conditions, plus a short theory and perception test). The new human driver is typically inexperienced and society's expectation is that the new driver will then considerably improve driving capability as they gain experience. An equivalent improvement of an ADS would only happen if the systems were updated after sale to customer (via over-the-air updates or at service interval) and the regulatory processes for these types of arrangements are not currently defined. Simulation has the potential to improve on this limited human driving test and to facilitate tests of ADS updates.

Current automotive functional safety practice (ISO 26262 and SOTIF PAS)

ISO 26262 [9] is the automotive industry functional safety standard with the remit of providing guidance to avoid potentially safety-critical situations caused by hardware and software failures. ISO 26262 has been successful in improving the safety of road vehicles and still has a key role to play in the safety of automated vehicles. However, it only helps with *some* aspects of the safety assurance problem.

A fundamental limitation of ISO 26262 is that it only covers *malfunctions* and not the so-called 'Safety of the Intended Functionality' (SOTIF). Functional insufficiencies of features are not covered (e.g. range or field-of-view of a lidar sensor for example) and manufacturers are left to satisfy themselves that the systems are safe in all cases even when there is no malfunction of any function. An attempt to address this issue for automated functions (*but critically only up to SAE Level 2*) has been made in ISO/AWI PAS 21448 (*currently in draft*) [10], which has the objective:

> *"This proposal is intended to be applied to systems for which a proper situational awareness by the item is critical for safety, and is derived from complex sensors and processing algorithms, which is often the case for ADAS systems".*

Although PAS 21448 does not have an explicit section on the use of simulations, it does imply that they can be used to understand the nature of failures (i.e. '*In some cases it is possible to emulate an unexpected behaviour of the sensor by means of fault injection at simulation level'.. …'outcomes of those simulations may be combined with results of safety analyses').* We suggest that simulation methods need to be developed and gradually incorporated into future automotive and ADS specific standards.

4. Vision for an integrated, automated simulation infrastructure

To enable the volume of simulation testing needed to ensure the intended functional safety of ADSs, a simulation framework has several requirements. Specifically, it must:

- Run without human intervention;
- Focus on the situations that ADSs find challenging;
- Apply some variation to the tests it runs, rather than simply repeating the same tests;
- Run a large number of tests per day.

Below we discuss some of the components which are needed to satisfy these requirements.

Scenarios

Challenging situations – often termed *scenarios* – are central to assuring the safety of AVs, as they are easily understood by humans, are easy to convert into automated test cases, and can capture the key characteristics of situations that are challenging for an ADS.

Scenarios should cover both edge cases and normal driving. Edge cases are situations that are expected to be extremely rare, but may be difficult for an ADS to handle correctly. We note that, if an event has a one-in-a-million chance of happening to a driver per year, given a fleet of 30 million vehicles we would expect 30 such events per year. Edge cases also include seemingly innocuous variations of normal driving that pose a challenge for an ADS, such as four vehicles arriving simultaneously at the four entrances onto a roundabout.

Test generator

The goal of testing is to find problems with the system-under-test, and a key metric is test coverage: of all the possible situations the system may encounter when deployed, how many have been tested? Designing tests to maximise coverage is often termed coverage-driven verification [11], and is one of the key responsibilities of the test generator. The other aspect of its role is to analyse the behaviour of the ADS from previously tested scenarios and then focus future tests on any potential areas of concern.

For each individual trial, the test generator will start by choosing which scenario to run. We anticipate there will be a limited number of scenarios, and each one will be used in many trials, with randomisation or "fuzzing" applied. The randomisation could simply mean changing the initial poses and velocities of the AV-under-test and the other actors, or it could also mean changing the colours and models of the other vehicles, weather conditions, time of day, degradation of road markings and street signs, and even the layout of the road itself — all depending on the capabilities of the simulator.

The goal of maximising coverage implies that we should test every possible decision the ADS could make in a particular scenario. For example, when joining a main road, the ADS should sometimes judge that it has space to join in front of an approaching vehicle, and sometimes that it should wait for the vehicle to pass. The test generator should manipulate conditions over successive trials to ensure both decisions are produced. A similar argument applies when analysis of previous results indicates the ADS came close to the boundaries of acceptable behaviour for a scenario: future tests should explore the nearby parameter space to ensure that those boundaries are never breached.

Full framework

The full framework espoused in this section is shown in Figure 3. The process flow is that first the test generator selects a scenario to test from a library of scenarios, and applies randomisation to it. Then the scenario is translated into an environment and actor initial conditions, and executed within the simulator. The ego vehicle is controlled by the ADS-under-test, and other actors are controlled by intelligent actor

models (as discussed later in this section). The input to the ADS is provided by sensor models, which simulate the raw sensor data expected given the AV's pose, the 3D environment, and poses of all other objects in the AV's field-of-view. The output from the ADS is passed through a vehicle dynamics model to generate an updated AV pose, and once the scenario is complete, the trajectory trace of the AV and all other actors is passed to a test oracle (also discussed below) for validation. Once enough scenarios have been executed, the ADS performance from each scenario can be examined to produce a summary.

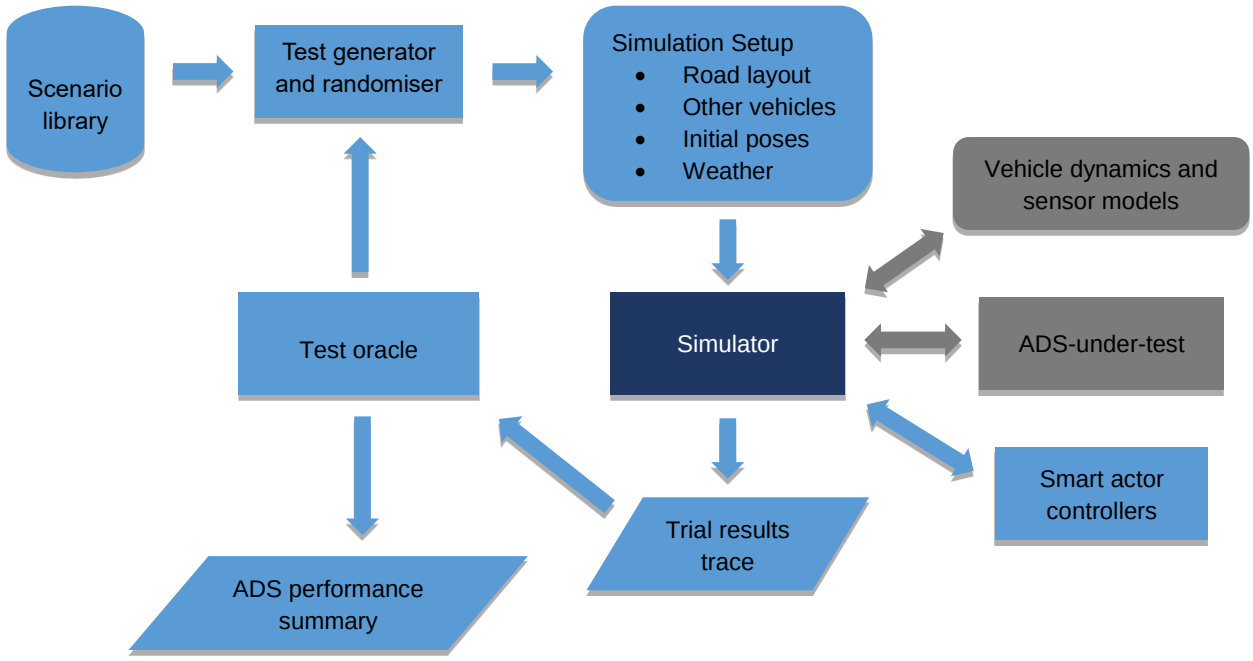

Figure 3 – Architecture for a complete integrated, automated, simulation ADS test framework

Smart actor controllers

Many interesting scenarios will involve the ADS interacting with other actors, including human-driven cars, pedestrians, bicycles, lorries, other AVs, and so on. Sometimes it will be sufficient for these to travel pre-defined paths, but more often, they will have to react to the decisions made by the ADS. Examples of this include:

- If the ADS stops unexpectedly, a simulated following vehicle should also stop;
- If the ADS drives slowly approaching a roundabout, an actor vehicle approaching from another direction should no longer give way to it;
- When an ADS is attempting to merge into fast-moving traffic, actor vehicles should slow down to make space for it.

Once scenario-based testing (which the architecture shown in Figure 3 is focused on) is established and efficient, the next step could be a town-scale simulated environment where multiple ADS-controlled AVs can interact over a longer timescale. While this will involve testing a larger proportion of "ordinary" driving, it creates the potential for stochastic interactions to result in unanticipated

challenging situations, and therefore increased test coverage.

Test oracle

If a human can watch what an AV does, it's relatively easy for them to say if it did the right thing or not. However, if a test environment has thousands or tens of thousands [12] of simulations running at any given time, human supervision quickly becomes unviable, and the requirement for automated test analysis emerges. The smart software doing this analysis is referred to as a *test oracle*, and must judge much more than whether or not the AV collided with anything. For example, did the AV unexpectedly straddle the centreline of the road, and cause an oncoming vehicle to swerve off the road? Did it fail to stop at a *Stop* sign? Did it wait for 30s at a *Give Way* sign, even though there was no other traffic in the vicinity? Did it crash into a fire hydrant, but in doing so avoided hitting a pedestrian crossing the road?

As well as watching for violations of the rules of the road (for a particular territory), the test oracle should judge if the ADS drove smoothly and considerately, and in keeping with the conventions of driving in that territory. While implementing a test oracle is probably simpler than creating an ADS in the first place, it is still a large and complex undertaking. Possibly, in the near term, algorithms will score the majority of the trials in a test run but will escalate cases it's not sure about to a human.

5. Key challenges

This section highlights some specific challenges likely to be encountered in building the automated simulation infrastructure described in Section 4. Given the number of organisations in the automotive sector, and the breadth of expertise required by the simulation framework, increased cooperation will be vital, in addition to advanced engineering solutions.

Firstly, testing in simulation is only worthwhile if you obtain the same results from the system-under-test as you would in an equivalent real-world situation. Given the sophisticated sensors deployed on AVs, this means the 3D model of the world used in the simulation must be a very good match to reality: it should either be a digital twin of a real-world location, or if an invented world, it should represent objects and infrastructure with as much detail as can be discerned by the AV's sensors. For the case of digital twins, capturing the data, cleansing it, and then performing sensor fusion and processing to synthesise a coherent representation is non-trivial. Even given this, a further factor is access and sharing of the data: progress in the field can happen more rapidly if different organisations do not have to duplicate effort in creating digital models of the road network.

Once you have a sufficiently high-fidelity environment model, you have to simulate the way each specific sensor on an AV will see that environment and convert it to electronic data. This requires sensor models which will encode the physics of how cameras, lidars, radars and other sensors work, and again these sensor models should be high fidelity if the behaviour of the ADS in simulation is to match its real-world behaviour. Using state-of-the-art camera models, it is possible to render extremely realistic images; however, there is a trade-off between realism and computation time, making model fidelity an issue for real-time simulators. For other sensors such as lidar and radar, there has been

much less research on efficient, high-quality simulation models, and this must be an area of significant investment to allow effective end-to-end validation of AVs in the future.

We expect sensor models to be produced both by simulation tool vendors and the sensor manufacturers themselves (often Tier 1 or 2 automotive suppliers). In either case, the two groups should work together, to ensure fidelity to real sensors and interoperability with common simulation tools.

In many cases, sensors will include some level of sensor processing algorithms, potentially even within the same physical unit as the sensor. In these cases the only way to test purely in software would be for the sensor supplier to provide their algorithms as independent executable software, to be used within a simulation. This requires significant trust and IP protection between all parties involved, as the sensor suppliers are likely to consider these algorithms as core to their business.

Depending on the stage in the validation cycle, the characteristics of the system-under-test, and the preferences of the ADS developer involved, the simulation framework will often have to include some hardware-in-the-loop. This can range from the full vehicle-in-the-loop, through sensors in the loop, vehicle ECUs (e.g. for ABS) in-the-loop, and finally just the ADS ECU in-the-loop. Facilitating easy substitution of any part of the simulation framework with real hardware relies on a clear, well-designed architecture and, again, close collaboration between all the organisations involved.

In summary, interfacing the tools and software from all stakeholders together into a single simulation will be a significant challenge. This will need:

- Strong protections for the IP of all parties involved, both on a legal and a technical level (for example, by encrypting and signing the executable code which is shared).
- Either significant development effort to code adapters between different components, or (ideally) standardisation of the APIs connecting the various boxes shown in Figure 3, to enable components from different organisations to be seamlessly plugged in together.

6. Relevant AV validation projects and initiatives

PEGASUS[2] is a highly relevant project, started in 2016 and involving 17 major German automotive organisations [2]. The project focuses on validation of a highway pilot and has 4 strands: SP1 to develop quality requirements and methods for finding critical scenarios; SP2 looking at how AV validation should be incorporated into the vehicle development process; SP3 to create tools to perform testing, including simulation, testing grounds and public roads; and SP4 on disseminating the project outputs.

Enable-S3[3] is a large Horizon 2020 project looking at verification of complex autonomous systems across several domains, including off-road ground-based vehicles (e.g. agricultural tractors), aviation and rail as well as AVs. The project will define methods for test data management, management of tests

[2] https://www.pegasusprojekt.de/en/
[3] https://www.enable-s3.eu/

themselves, and options for the test framework.

Other significant projects include Singapore's CETRAN initiative[4] led by NTU, the MOOVE and SVA projects[5] supported by the French government, and the Streetwise project[6] led by TNO.

Several standardisation initiatives are also making progress: OSI[7], the Open Simulation Interface, provides standards for exchanging sensor data between a simulation environment and an ADS, OpenScenario[8] defines an open XML standard for scenarios, and OpenDRIVE[9] (which is used by OpenScenario) defines road layouts and attributes in a portable manner.

A major project that the Transport Systems Catapult is leading is MUSICC (Multi-User Scenario Catalogue for CAVs), which is exploring the use of scenarios for the regulatory assurance of AVs. The project is supported by a broad coalition of UK stakeholders, under the strategic guidance of CCAV[10] and leveraging close liaison with Meridian Mobility[11], and is sponsored by the UK's Department for Transport (DfT). The DfT chairs the UNECE's WP.29 AutoVeh Task Force[12], which is developing methods for certification of autonomous and highly-automated vehicles, and the intention is for MUSICC to feed in to that process.

Guided by the needs and views of the stakeholders, the intended goals of the project are to adopt or define a format for storing scenarios, and create an open, live database which can store scenarios in that format. Creating scenarios (other than a minimal set for system testing) is out-of-scope for the project: we hope that other stakeholders will take up that baton going forward. However, the ethos of the project is to engage with all interested parties to help define the proof-of-concept format we will use for storing and sharing scenarios. The aim is to have a functional database by Spring 2019; this is an ambitious target, but one that should benefit the whole AV verification community by providing a prototype system to drive scenario-driven verification, and thus gain clarity about where future progress is most needed.

7. Conclusions

The capability of simulation to provide assurance of ADS is in its infancy and caution should be taken while the simulation capability is improving in fidelity. Development in computing power and artificial intelligence are both making this capability a more realistic option for the first time. However, the end goal of using simulation is important and will allow more confidence in ADS systems.

There is clearly a fast-evolving market for simulation tools in the AV domain. Due to the large number

[4] http://erian.ntu.edu.sg/Programmes/IRP/FMSs/Pages/Centre-of-Excellence-for-Testing-Research-of-AVs-NTU-CETRAN.aspx
[5] https://www.irt-systemx.fr/en/project/sva/
[6] https://www.tno.nl/en/about-tno/news/2017/7/tno-introduces-streetwise-scenario-based-methodology-for-validation-of-automated-driving/
[7] https://github.com/OpenSimulationInterface/open-simulation-interface
[8] http://www.openscenario.org/
[9] http://www.opendrive.org/
[10] https://www.gov.uk/government/organisations/centre-for-connected-and-autonomous-vehicles
[11] https://meridianmobility.tech/
[12] https://wiki.unece.org/pages/viewpage.action?pageId=60361611

of potential competing players adapting products from related areas, the market can expect to see consolidation and continuing alliances between tool vendors to perform parts of the verification task.

In the next 5 to 10 years, it may be possible that a common testing architecture will emerge and, if so, this is likely to have an alliance of industry-leading simulation tool providers as its basis.

The creation of the end-to-end automated simulation framework described in Section 4 is technically challenging, and additionally the framework is likely to integrate components from many different organisations (such as simulation tool developers, automobile OEMs, high-tech start-ups, Tier 1 and 2 suppliers, and digital mapping organisations). To work efficiently and take full advantage of rapidly advancing technology, the key players will have to communicate frequently and develop new methods of trusted collaborative working.

References

1. Kalra, N., S. M. Paddock (2016). *Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability*? RAND Corporation Technical Report.
2. Winner, H (2017). Safety Assurance for Highly Automated Driving – The PEGASUS Approach. In *TRB Annual Meeting 2017*, Washington DC, USA. January 2017. https://www.pegasusprojekt.de/files/tmpl/pdf/TRB%20Annual%20Meeting%202017%20Folien.pdf
3. Jayaraman, A., A. Micks, E. Gross (2017). *Creating 3D Virtual Driving Environments for Simulation-Aided Development of Autonomous Driving and Active Safety*. SAE Technical Paper. doi:10.4271/2017-01-0107
4. Schöner, H-P. (2017). The Role of Simulation in Development and Testing of Autonomous Vehicles. In *Driving Simulation Conference – DSC*, Stuttgart 2017-09-07
5. LUTZ Pathfinder and CavLab. Project website: https://ts.catapult.org.uk/innovation-centre/cav/cav-projects-at-the-tsc/self-driving-pods/ https://ts.catapult.org.uk/cavlab/
6. Peters, A (2015). Safety of the LUTZ Pathfinder Automated Vehicle. In Proceedings *22nd ITS World Congress*, Bordeaux, France. ERTICO (ITS Europe)
7. BSI & Transport Systems Catapult (2017). *Connected and autonomous vehicles: A UK standards strategy*. https://ts.catapult.org.uk/innovation-centre/cav/cav-projects-at-the-tsc/uk-standards-strategy-cavs/
8. Transport Systems Catapult (2017). *Taxonomy of Scenarios for Automated Driving*. TSC Technical Paper. https://ts.catapult.org.uk/intelligent-mobility/im-resources/research-papers/
9. ISO (2011). *ISO 26262 - Road Vehicles - Functional Safety*. Parts 1 to 10.
10. *ISO/WD PAS 21448 – Road Vehicles – Safety of the Intended Functionality*. Under development: https://www.iso.org/standard/70939.html
11. Araiza-Illan D., D. Western, A. Pipe, K. Eder (2015). Coverage-Driven Verification — An Approach to Verify Code for Robots that Directly Interact with Humans. In: N. Piterman (eds), *Hardware and Software: Verification and Testing.* LNCS, vol 9434. Springer, Cham.
12. Madrigal, A. C. (2017). Inside Waymo's Secret World for Training Self-Driving Cars. *The Atlantic*. https://www.theatlantic.com/technology/archive/2017/08/inside-waymos-secret-testing-and-simulation-facilities/537648/