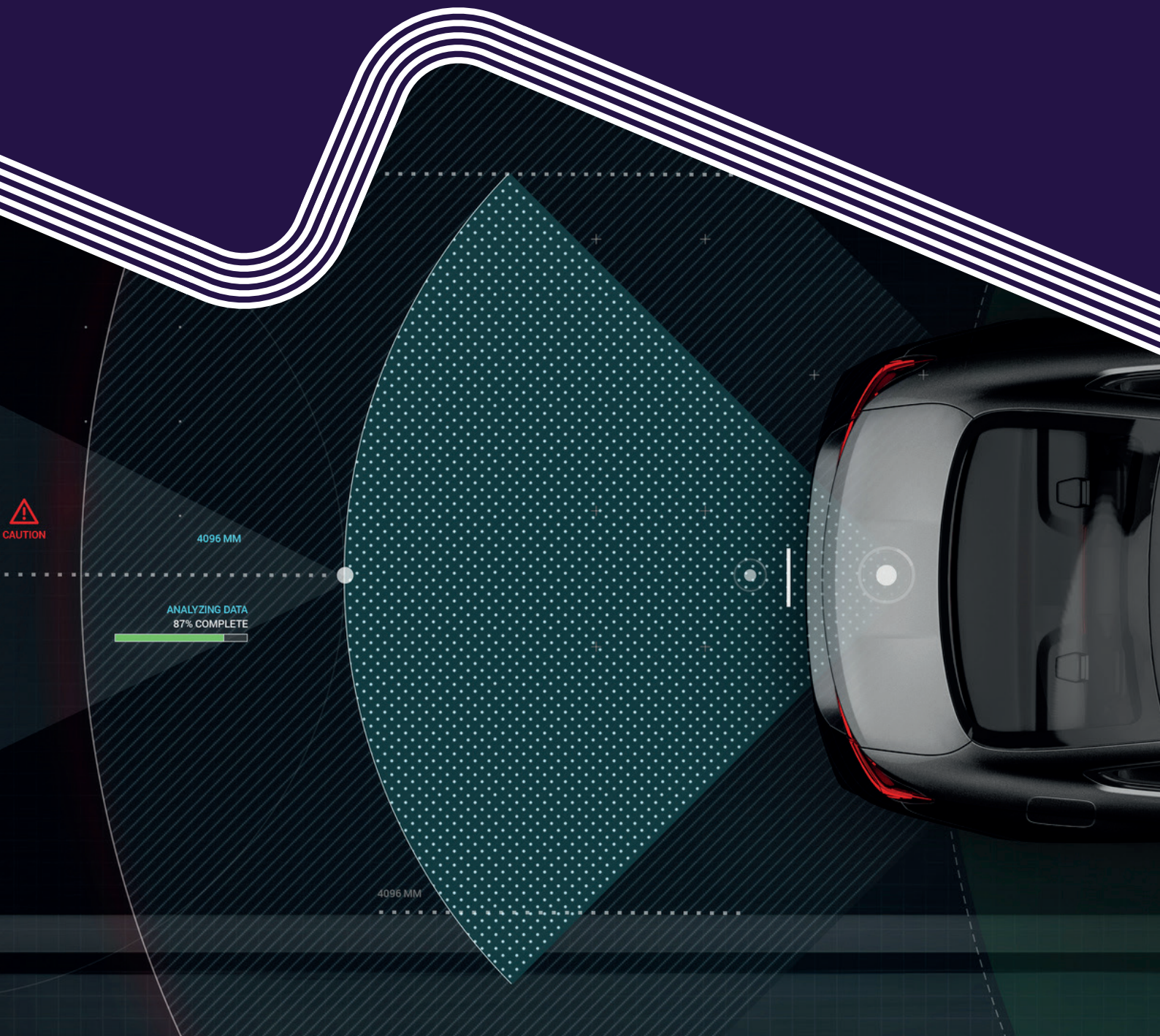# REGULATING AND ACCELERATING DEVELOPMENT OF HIGHLY AUTOMATED AND AUTONOMOUS VEHICLES THROUGH SIMULATION AND MODELLING

TECHNICAL REPORT – March 2018

cATAPULT
Transport Systems

CAUTION

4096 MM

ANALYZING DATA
87% COMPLETE

4096 MM

Contents

DISCLAIMER

This report has been produced by the Transport Systems Catapult under a Memorandum of Understanding with the Department for Transport. Any views expressed in this report are not necessarily those of the Department for Transport.

RECORD OF CHANGES:

RELEASED TO	VERSION	REASON FOR CHANGE	DATE
CCAV	Version 1	Public Version	March 2018

AUTHORS:

Dr Zeyn Saigol – Senior Technologist, Transport Systems Catapult.

Dr Alan Peters – Principal Technologist, Transport Systems Catapult.

Matthew Barton – Head of Project Delivery, Transport Systems Catapult.

Mark Taylor – Senior Technologist, Transport Systems Catapult.

Please address any questions / corrections to:

Alan.Peters@ts.catapult.org.uk

1 EXECUTIVE SUMMARY

Autonomous vehicles (AVs) will be required to perform the driving task in the presence of a huge variety of road layouts, other road user movements and environmental conditions. This means the software and sensors that constitute the Automated Driving System (ADS) and enable a vehicle to operate in a highly automated or autonomous manner (e.g. SAE Level 3[1] and above), must be highly sophisticated, able to acquire appropriate contextual information and make safe and predictable decisions. Demonstration of confidence in such a system is a major step towards the introduction of AVs to our roads.

In considering the replacement of the human driver by a system of sensors and automated decision making, it is not realistic to test every combination of sensor input and driving situation; however, confidence is required that the ADS can operate effectively at a whole system level across the full range of situations it is likely to encounter. It is not viable to achieve this from physical testing of a limited number of use cases that would be encountered on a defined set of test routes. At this moment in time, the number of miles an AV would have to accumulate to be tested in all situations is incalculable, but likely be measured in billions. Simulation, modelling and testing has the potential to fill this gap and to enable rigorous, controlled and timely evaluation of ADS systems.

The study seeks to explore capabilities from the perspective of:

• **Simulation and modelling for design and development of ADS:** clarify the requirements for a simulation capability that will accelerate the development of ADSs.

• **Simulation for regulation and approval:** A key challenge for regulators is to understand how to determine whether AVs, at the whole system level, are safe so that they can be approved for sale in the international market.

Simulation and Test Framework Proposals

The project has proposed two phases of interventions that could be enacted to provide a framework for the test and simulation of AVs:

• Phase 1 addresses the short term need to provide regulators with a means to assess the performance of an ADS, it relies on:
  – sensor processing being evaluated in the real world at a controlled test facility or proving ground
  – a simulated environment (e.g. a well-characterised digital twin of a test facility) to evaluate the ADS
  – a physical test at a proving ground of the driving performance of an AV

• Phase 2 addresses the longer term needs of regulators and industry, building upon Phase 1. It integrates sensor testing into a simulated environment that will ultimately enable the timely test and evaluation of OTA updates to AVs

It is anticipated that in time the dependence on physical testing will decrease, whilst the extent and confidence in simulated testing grows. It is expected that an element of physical testing will always remain as part of the evaluation of AVs.

[1] https://www.sae.org/misc/pdfs/automated_driving.pdf

2 INTRODUCTION

2.1 CONTEXT

The automotive industry, and by implication the transport industry, stands on the cusp of a revolution. After over a century of evolution in which the basic premise of a vehicle (chassis, drivetrain, driver role) has changed little, advances in technology that were once the subject of science fiction and futurologists are now almost with us, solutions that will allow the driver role to be removed from the vehicle, replaced by an array of sensors, controls and decision-making software. Highly automated and autonomous vehicles are nearly here.

Many benefits are anticipated to arise from the introduction of highly automated driving, the removal of human error (a contributory cause in ca. 80% of UK road accidents) is a key benefit, together with reduced congestion, increased accessibility to transport, reduced impact on the environment. For the companies that take the lead in bringing highly automated and autonomous vehicles (AVs) to market first and successfully then there is a potential economic opportunity, likewise a range of disruptive business models may be enabled by AVs.

To realise these and other benefits, then the vehicle fleet will need to evolve such that AVs are the predominant type of vehicle on the road. For this change to occur then regulators need to be confident in the reliability, safety and performance of the systems that replace the human driver and the integration of these systems with the rest of the vehicle. Without supporting evidence then this game changing new technology will not be allowed on the roads of the UK, Europe and the rest of the world. Likewise, the public needs to be assured of the safety of these vehicles and convinced of the benefits they will bring.

2.1.1 Project Objectives

AVs must be able to drive under the same variety and unpredictability of circumstances that human drivers currently manage. This means the software and sensors that constitute the Automated Driving System (ADS) and enable a vehicle to operate in a highly automated or autonomous manner (e.g. SAE Level 3[2] and above), must be highly sophisticated, able to acquire appropriate contextual information and make safe and predictable decisions. Demonstration of confidence in such a system is a major step towards the introduction of AVs to our roads.

In considering the replacement of the human driver by a system of sensors and automated decision making, it is not realistic to test every combination of sensor input and driving situation; however, confidence is required that the ADS can operate effectively at a whole system level across the full range of situations it is likely to encounter. It is not viable to achieve this from physical testing of a limited number of use cases that would be encountered on a defined set of test routes. At this moment in time, the number of miles an AV would have to accumulate to be tested in all situations is incalculable, but likely be measured in billions.

Simulation, modelling and testing has the potential to fill this gap and to enable rigorous, controlled and timely evaluation of ADS systems.

The study seeks to explore capabilities from the perspective of:

- Simulation and modelling for design and development of ADS: clarify the requirements for a simulation capability that will accelerate the development of ADSs.
- Simulation for regulation and approval: A key challenge for regulators is to understand how to determine whether AVs, at the whole system level, are safe so that they can be approved for sale in the international market.

2.1.2 An Ecosystem for Simulation and Test

The automotive industry has developed and refined its design and verification process to deliver vehicles that are demonstrably safer, more efficient, and appealing than those available to the generation that went before us. In a scenario where the human driver is replaced, all of the existing test and verification requirements remain, but the testing industry needs to supplement existing approaches with a means of designing, building, testing and verifying that the ADS performs at least as well a human driver in all scenarios. Arguably the solutions developed need to be incrementally better than the human driver to convince the public to relinquish the driving task.

---

[2] https://www.sae.org/misc/pdfs/automated_driving.pdf

Developing a conventional new vehicle is a costly and time-consuming process, typically taking around four years and costing in the billions, for example Dyson's recently announced electric vehicle is a £2.5bn project that started in 2015 and will bring a product to market in 2020[3]. Within the UK an ecosystem exists to support the development of conventional vehicles: design agencies will inform the user experience of the vehicle generally designed in house by an OEM; a number of Tier 1 and Tier 2 suppliers (both within the UK and the wider continent) will support the productionisation of the design and the provision of a supply chain to enable its manufacture. Companies such as Horiba-MIRA and Millbrook will provide services to the OEM and regulator to test and verify the design and its worthiness for Type Approval.

The need for this conventional vehicle ecosystem remains as the industry transitions into the era of AVs, however the ecosystem needs to be extended and supplemented to address the need to verify the design of the ADS and demonstrate its worthiness for Type Approval and hence introduction to the market.

Regulators such as NHTSA and UNECE are currently working with the automotive industry to develop regulations that will support the introduction of AVs to the market, the developments to UNECE Regulation 79 for Automated Steering being a prime example. In this example regulators from around Europe are collaborating with OEMs to develop a set of standards and requirements that will allow the introduction to the market of a number of automated driving use cases.

If an ecosystem is to be created that supports the development and introduction to the mass market of an extensive suite of highly automated and autonomous driving use cases, in a new generation of vehicles, then the existing collaboration between OEMs and regulators should be extended significantly. Additional stakeholders will need to be involved to define and demonstrate the standards that this new generation of vehicles must meet to obtain Type Approval. As a bare minimum, industry and regulators need to agree what constitutes a safe and reliable system and how to evaluate the solutions developed by industry.

2.1.3 Key Challenges

The study considers the challenges associated with the simulation and modelling of ADS implementations from the perspective of the technical issues to be resolved and how these align with the challenges faced by regulators.

From a technical perspective, a highly automated or autonomous vehicle needs to be able to:

1. Know where it is at any given point in time;
2. Know what is going on around it, and
3. Determine how to safely and efficiently navigate its environment to reach its destination.

To successfully deliver the above the vehicle must be able to respond to many factors, such as different 'actor types' (vehicles, pedestrians, animals etc.), different road conditions and layouts, a range of permanent and temporary traffic signals, different weather conditions etc.

Depending on the inputs received from ADS sensors, then different decisions may be required for each possible sensor combination. Following a traditional verification and validation test cycle approach (see Appendix **8.3**) to test the performance across the full range of situations that may be encountered would require an almost infinite number of test scripts to be written and executed in physical world situations. Whilst it is possible (albeit unlikely) that such a test suite could be executed and successfully passed, thereby demonstrating that the software is bug-free, it is still possible that the AV could make poor decisions resulting from a combination of poor system specification and unplanned input conditions, effectively specification bugs. In addition, it is not appropriate to solely test individual system modules in isolation, since assessing the performance of a fully integrated ADS implemented in a vehicle is essential to determining if the system as a whole meets the required performance levels.

[3] https://www.theguardian.com/technology/2017/sep/26/james-dyson-electric-car-2020

From a regulatory perspective, many challenges need to be addressed in an environment where emerging technology is being applied to a task (driving) where there is no clear 'right' answer to the problems an ADS will encounter:

- The regulator, in conjunction with industry, needs to agree what constitutes a safe and reliable ADS, the level of safety needed is not yet clear; but AVs will arguably need to be safer than human-driven vehicles;
- The regulator then needs to be confident that the AVs deployed on the roads are safe;
- The regulator may potentially face pressure from both government and the public to guarantee that AVs will not behave recklessly;
- The regulator will want to see a progressive improvement in AV safety;
- The regulator (and industry) need access to the appropriate tools to allow the assessment of the ADS to be carried out;
- The regulator will need the support of industry to create sensible and workable regulation that is then adhered to and is fair to all market participants;
- The regulator should encourage innovation and improvement of performance, within the boundaries of what is known to be safe and reliable;
- Industry needs a clear certification methodology, so that it can minimise risk and cost when certifying its products.

This study seeks to address the above challenges and to propose a way forwards to CCAV and DfT.

2.2 PROJECT APPROACH

The project was delivered over an eight-week period by a team drawn from the Transport Systems Catapult (TSC) with input from CCAV and the DfT International Vehicle Standards (IVS) team. The approach to the project was:

1. A review of public domain information plus prior reports published by the TSC;
2. A series of interviews with stakeholders to explore specific issues related to the project, held face-to-face wherever possible, and
3. Presentation and discussion of a 'strawman' on the use of simulation & modelling in AV regulation for UNECE WP.29 meeting.

The above activities provided a considerable amount of information and insight which have been documented in this report. A key challenge associated with the delivery of the project was that whilst the development of AVs is becoming a significant industry, there are still comparatively few people working in the area of simulation and modelling for AV verification. Often such people are 'hidden' in large organisations and 'finding them' can often be challenging.

It should be noted that the insurance industry is a major stakeholder in bringing AVs to market. It was agreed not to approach this stakeholder group as part of this exploratory project but ensure they are consulted in relevant follow-on work.

2.2.1 In Scope

The study considered the following:

- **Tools:** what tools and tool chains exist to support the simulation and test of AVs
- **Scenarios:** feasibility of developing a user case database against which testing and verification can take place – and suggest a potential methodology for developing them
- **Vehicle testing and response:** how the vehicle / system is tested against the scenarios, what the safe response is and what the vehicle's response is.
- **Data:** the type of data that needs to be collected from the vehicle, and the methodology for collecting it, against which the ADS should be tested. The requirement (or need) for data sharing in such a system should be considered and, in particular, whether data sharing should be mandatory in return for utilising the system for development / testing
- **Dynamic updating:** the potential for the scenarios in the system to update as the tests are conducted and / or as additional features are added to a vehicle
- **Robustness:** How we prove the simulation itself.

---

4 Safe defined as an 'absence of unreasonable risk'. This implies risks have been fully identified, assessed, and addressed.

2.2.2 Out of Scope

*The following was considered to be **out of scope** for this study:*

- The focus of this work was on the contribution of simulation and/or modelling techniques; however, this will be discussed in the wider context of real-world physical tests and wider test programmes;
- Although the ability to simulate connectivity is important, this was not a focus of the study, since the simulation of communications (e.g. radio frequency simulation) is its own complex domain;
- Simulation and testing specifically to detect cyber security vulnerabilities was not considered;
- The focus was on simulation and/or modelling tools and techniques that can be used to enable SAE level 3, 4 and 5. Any tools that are limited in scope to SAE 1 or 2 have been given less prominence;
- The focus of the study was on road-based AVs, and
- Due to time constraints, the project did not specifically focus on lessons and/or processes that could be transferred from other domains, such as aerospace, nuclear, or defence. Some of the stakeholders referenced experience from these domains and some of the tools (e.g. dSpace, Simulink) are used across a variety of domains so the findings of this report are influenced by non-automotive processes but this area would merit a further detailed exploration.

2.3 STRUCTURE OF DOCUMENT

The remainder of Chapter 2 is given over to defining the various concepts used throughout the report, together with providing a high-level overview of how the simulation and test of AVs may be approached. Chapter 3 summarises the system level challenges associated with the simulation and test of AVs, together with an overview of relevant standards and approaches to achieving safe systems. Chapter 4 describes the range of technical challenges that would need to be addressed to test an AV in a simulated environment and also proposes a potential test framework. Chapter 5 summarises the current state of the art associated with the various tools which could be used to develop a test and simulation solution, and this is complemented by Chapter 6 which describes the infrastructure required to allow the test and simulation solution to be implemented. Finally Chapter 7 summarises the conclusions drawn from the study.

2.4 CONCEPTS & DEFINITIONS

This section briefly introduces the most important terms and concepts for understanding the remainder of the document. A complete glossary of terms is provided in Appendix 8.1.

2.4.1 Key Definitions

Sensor processing is defined in this study as obtaining raw sensor data (in the form of, for example, a stream of RGB video data, or lidar data represented as point clouds) and extracting object properties from it. These object properties may include spatial location and orientation, velocity and acceleration, category (such as bicycle, lorry, SUV, and so on), colour, and others. Sensor processing will also include object tracking, i.e. associating object identity across data from different times, and the fusion of data from different sensors.

An ADS (autonomous driving system) is the hardware and software responsible for making autonomous driving decisions, and this includes the autonomy sensors, sensor processing, and path planning. The term ADS is defined in SAE J3016[5], the same standard that defines the commonly used 1-5 levels of autonomy, and is gradually gaining favour as the default term.

An object model is a map of all the dynamic elements an ADS's sensors can detect. It may include the location, orientation, velocity and acceleration of nearby vehicles and pedestrians, details of the environment such as the location and height of kerbs, what markings are present on the road, the location and status of traffic lights, and potentially other information. Building and maintaining an object model is the role of sensor processing in many ADS architectures.

For simplicity, we use *AV (autonomous vehicle)* to mean a L3, L4 or L5 vehicle.

2.4.2 ADS Overview

The ADS is defined to include all the components shown in Figure 1, where it's possible some of the components may be merged or re-ordered in some ADS architectures. The sensors capture a view of the world, SPUs (sensor processing units) convert that into object-level data, sensor fusion merges and resolves conflicts between data from different sensors. Then the planner performs a variety of functions to determine the trajectory the AV should take: it may predict the future movements of other actors, reconcile the long-term route goals of the ADS with the immediate road layout, and perform path planning around current and predicted obstacles. Finally, the control ECU translates a medium-term desired trajectory into immediate low-level commands for the vehicle actuators.
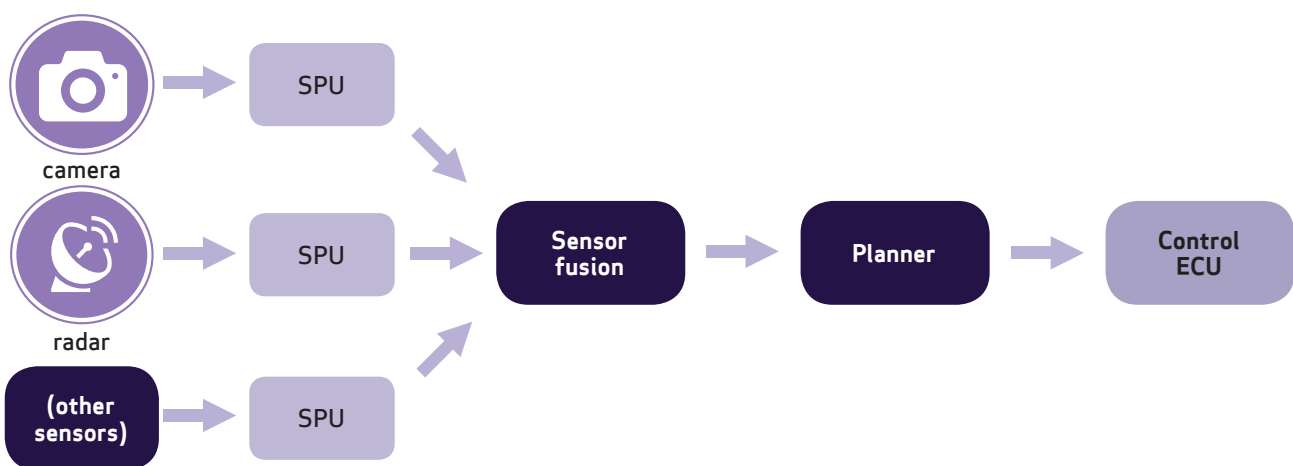

Figure 1: Components of a typical ADS. SPUs are sensor processing units, which may be integrated into the sensors.

We note that there is a general (if not complete) industry consensus, especially amongst the organisations closer to the development of a production ADS, that there will be a clear interface between sensor processing and planning, and that this interface will be in the form of an object model. However, one or two of our interview respondents expressed doubt that this separation will be universal.

A more detailed description of a possible ADS architecture is given in Appendix 8.4.

2.4.3 ADS Simulator Anatomy

An ADS simulator is a software tool or set of tools that enables a sensor-driven vehicle control module (which might be a complete ADS, or a subsystem, or a driver aid such as adaptive cruise control) to be developed and tested in simulation.

These are also known as scene generation tools, and they require several components:

• An ability to generate test scenarios (or handle those generated by other means).
• A 3D environment model. This defines the road network, including the geometry of junctions, number of lanes, width of lanes, road surface, and lane markings; the roadside infrastructure, such as street-lights, street-signs, crash barriers, buildings, pavements; the material and colour of each element of the model; and other actors in the system such as cars, bicycles and pedestrians. It may also include the weather, time of day and global coordinates, and other light sources in the scene. However, not all simulators will include all these elements in their environment models.
• A sensor simulation system, which may be a set of models for each type of sensor the tool supports. For example, given the 3D environment and view point as input, a lidar model will produce sensor data in the form of range and intensity readings from each beam of the lidar. Sensor models can be generic, configurable (for a camera, parameters may include the resolution, frame rate, and focal length), or specific to a particular product.

- A vehicle dynamics model for the ego vehicle[6](s). This will model how the vehicle reacts to a given control input, under a given set of conditions.
- A simulation controller (the term simulator is loosely used to refer to the controller, plus all the additional components needed to make it work, but excluding the device-under-test).
- A way of visualising the progress of a simulation, often a GUI providing an overhead view of the actors and environment.
- Tools and interfaces to set up all these components, and to integrate the ADS software.

The simulation of sensors and actor movement may use a common, underlying physics engine.

2.4.4 Introduction to Testing in Simulation

Cyber-physical systems (or robots) such as AVs pose particular challenges for testing. They consist of software which controls the physical actions of the system, and reacts to the environment, which means the only mature way to test the complete system is to test it in the real world. However:

- It is difficult to cover enough of the test cases in the real world to be sure the system works correctly – and this is arguably entirely impracticable in the case of AVs;
- Hardware testing is well established and can be performed independently of the software;
- While subtle interactions between the hardware and software are the cause of some bugs, most incorrect behaviour will be completely due to software errors.

Given this, it makes sense to increasingly test cyber-physical systems virtually, where the real environment is replaced by a simulation that "fools" the software into thinking it is operating in the real world.

Testing in simulation requires two key elements: a simulator and a device-under-test (DUT). For testing AVs, if you consider the complete vehicle to be the DUT, then the simulator must be a facility which encloses the whole vehicle, sends emulated data to all its sensors, and translates movements of the cars' wheels (drive, braking, and steering) back into the simulator. Warwick University's WMG has such a simulator, which we term vehicle-in-the-loop (VIL).

Figure 2 shows the basic elements for a VIL simulator, assuming the vehicle's only sensor is a camera. VIL simulators have several drawbacks, as described in Section 4.1.
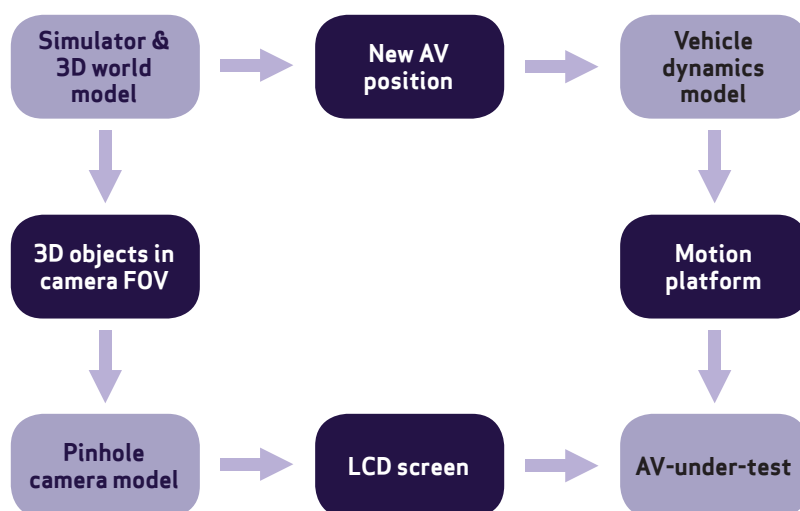
Figure 2: How a vehicle-in-the-loop simulator works, assuming a single camera sensor

However, the full benefits of simulation can only be achieved by running multiple simulations at the same time (potentially thousands), without human supervision, and without incurring high costs. To do this, a subsystem of the AV must be considered as the DUT, and the simulator must have defined software interfaces into this subsystem.

The ADS (excluding its physical sensors) is the obvious subsystem to test, and a system test framework for an ADS with a single camera as its sensor is shown in Figure 3.

[6] Ego vehicle is the vehicle itself under test
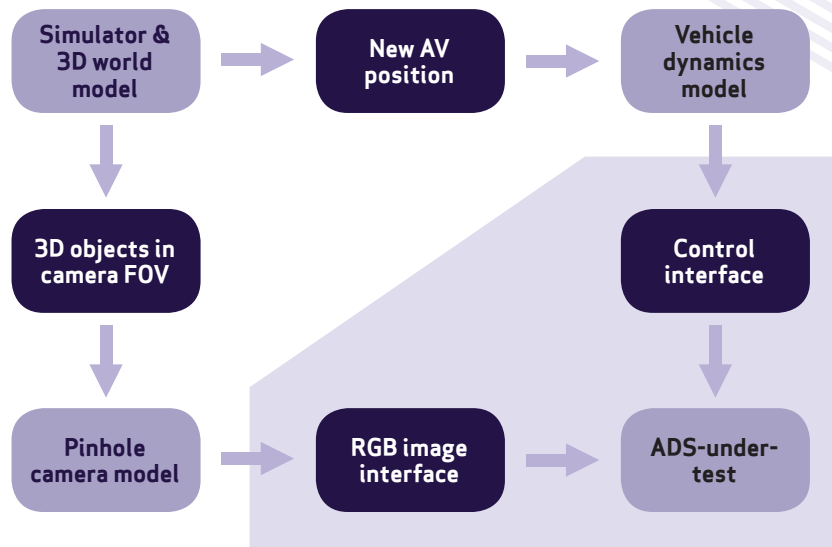
*Figure 3: Framework for a simulator that uses software interfaces to an ADS,
again with a single camera sensor*

This is broadly similar to Figure 2, the only differences are that the AV has been replaced by just the ADS, and the screen and vehicle platforms have been replaced by digital interfaces. The implications of this are:

- A software interface must exist for the inputs to the ADS.
- An interface must exist for the outputs from the ADS (although this is often present to some extent via the CANbus in the vehicle).
- The simulator no longer uses the physical parts of the sensors, but relies on all their behaviour being captured by the sensor models. However, this is essentially the case for VIL too, as the image on the screen is created by a sensor model.

It is also possible and useful to allow subsystems of the ADS to be tested in simulation, by changing the infrastructure so that it simulates the part of the environment responsible for input to the subsystem under test. An example would be testing just the planning subsystem of the ADS, using a simulator to directly generate an object map, which is the input to the planning subsystem. This would avoid the need to simulate the behaviour of the sensors themselves.

In summary, a Vehicle-in-the-loop simulator as depicted in Figure 2 represents one extreme of the range of possibilities, whereas a Software-in-the-loop simulator as depicted in Figure 3 represents the other extreme. The disadvantages of ViL are that it is limited to real-time testing, requires a physical space and can be expensive, which contrasts to the SiL approach where multiple parallel simulations can be run faster than real-time, provided fast, accurate sensors models are available. It is likely that pure ViL, pure SiL and many variants in between will all be needed as they have different strengths; however, it is the view of the authors that the SiL approach is likely to have the most benefit in addressing the testing complexity issues as introduced in Section 3.3.

Note: This view of the need for both approaches is broadly consistent with the 'validation' enabler line in the Automotive Council Intelligent Connected Vehicle roadmap[7] which predicts that the market is moving from 'Partial simulations' towards 'Full Environment, vehicle and sensor simulation' over a period of the next 10 to 20 years. The authors view is that this transition may well happen in a shorter timeframe.

---

[6] https://www.gov.uk/guidance/the-highway-code
[7] https://www.automotivecouncil.co.uk/wp-content/uploads/sites/13/2017/09/ICV-Roadmap.jpg

3 ROLE OF SIMULATION IN ADDRESSING THE CHANGING VERIFICATION CHALLENGE

ADS developers will need a programme of verification and validation activities to ensure safety and adequate performance of autonomous vehicles across the wide range of driving conditions and scenarios. This section explains the challenge posed from the driving task complexity, explains the deficiencies in the current functional safety process and introduces simulation methods as part of the solution.

3.1 CHANGING SAFETY ASSURANCE LANDSCAPE FOR AUTOMOTIVE

The current mechanisms for maximising road safety can be divided into three key areas: vehicle safety, operational safety, and environmental safety, as set out in Figure 4:
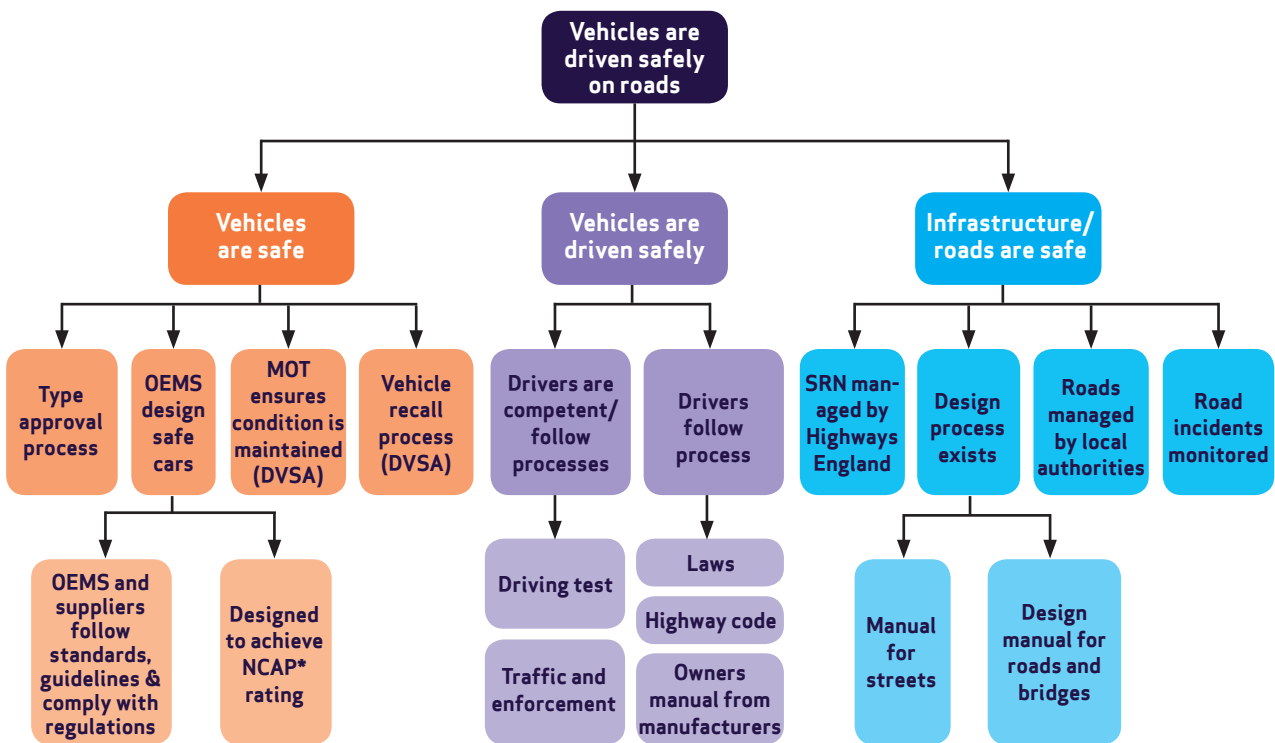
Figure 4 illustrates that there is currently a combination of technical, process and enforcement mechanisms to ensure that:

• Vehicles are 'safe';
• Vehicles are driven 'safely', and
• Infrastructure and roads are 'safe'.

For this report we are primarily interested in the role of simulation methods to verify that vehicles are driven safely (replacing the middle branch of Figure 4)[8]. With the introduction of AVs this driving task merges with the vehicle design tasks and becomes the responsibility of the system developer. Ensuring the performance of the driving function, currently controlled via the mechanism of the driving test and law enforcement, becomes a technical challenge.

Currently passing the UK driving test is a relatively low pass/fail barrier (~45 minutes in just one geographical area in one day's environment conditions). The new driver is typically inexperienced, and societies' expectation is that the new driver will considerably improve driving function 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.

---

[8] Required changes to the other mechanisms in Figure 4 as a result of AVs will also need consideration and may be able to benefit from simulation methods but are out of scope of this report.

3.2 CURRENT AUTOMOTIVE FUNCTIONAL SAFETY PRACTICE (ISO 26262 AND SOTIF PAS)

ISO 26262[9] is the automotive industry functional safety standard with the objective of minimising 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.

Weaknesses in the assessment of AV intended function can have consequences comparable to those of hardware and software failures, yet the means to formally ensure that equivalent safety in relation to fault failures is not yet mature. An attempt to address this issue for automated functions (but only up to SAE Level 2) has been made in ISO/AWI PAS 21448 *(currently in draft)*[10].

ISI/AWI PAS 21448 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 simulation techniques, the sections on verification and validation methods imply that simulations can be used to analyse the nature of the hazard or understand the effect of a sensor failure *(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'*

ISO 26262 itself does have some limited guidance on use of simulation in verification activities. Part 6, 'Product Development at the Software level' provides some guidance on verification of the software architecture design and recommends *'simulation of dynamic parts of the design'* (via the use of executable models). Part 6 further briefly notes that 'Software integration testing can be executed in different environments', and lists as examples:

- model-in-the-loop tests;
- software-in-the-loop tests;
- processor-in-the-loop tests; and
- hardware-in-the-loop tests

ISO 26262 Part 8 does have some guidance on the validation of software tools which may be applicable to simulation software tools. It defines procedures to classify, certify, or qualify software tools. To date, the authors are not aware of any common approaches for tool validation, certification, and qualification of simulation software.

There is also an introduction to Model-based system development in Part 6, Annex B but this reads as an add-on the main body of the standard and is not well integrated into the rest of the standard.

3.3 UNDERSTANDING THE COMPLEXITY CHALLENGE

An ADS is not like traditional automotive software. The cumulative complexity of the variety of driving tasks, variety of other vehicles behaviour, and types of environmental conditions result in an overall system functionality that cannot be described in a specification document.

For example, in the scenario of a rail level-crossing with barriers down, how would such a specification handle all the following complexity?

• If there's already a queue of vehicles at the barrier;
• If there's a lorry at the front of the queue, obscuring part of the barriers;
• If the flashing lights are not working;
• If it's nearly (but not quite) dark, and
• If it's nearly dark and there's a grey car at the back of the queue.

The ADS must respond to a large number of factors (e.g. the type and location of every actor in the scene, road layout, traffic signals etc) and different decisions may be required for each possible combination. It is not feasible to write an infinite number of test scripts and it is difficult to separate the problem into independent modules. The ADS needs to take everything in the environment into account when driving. This complexity is why current test processes are unlikely to provide complete safety assurance[11].

Without the use of simulation, there is much debate as to how many miles should be driven before AVs are *'proven'* to be safe. For example, one calculation estimates that to demonstrate that autonomous vehicles have a fatality rate of <1.09 fatalities per 100 million miles (i.e less than the human performance equivalent in US) using a fleet of 100 autonomous test vehicles driving 24 hours a day, 365 days a year, would take about 12.5 years[12]. Critically, any change to the ADS would then require an assessment of the need to repeat some or all of these test miles.

Automated systems also have a complexity that is at odds with the normal processes for development of safety critical software. A relevant article was published by Dr Steven Shladover in the June 2016 edition of Scientific American[13]:

> " Software for automated driving must therefore be designed and developed to dramatically different standards from anything currently found in consumer devices. Achieving these standards will be profoundly difficult and require basic breakthroughs in software engineering and signal processing. Engineers need new methods for designing software that can be proved correct and safe even in complex and rapidly changing conditions. Formal methods for analyzing every possible failure mode for a piece of code before it is written exist – think of them as mathematical proofs for computer programs – but only for very simple applications. Scientists are only beginning to think about how to scale up these kinds of tests to validate the incredibly complex code required to control a fully automated vehicle. Once that code has been written, software engineers will need new methods for debugging and verifying it. Existing methods are too cumbersome and costly for the job.

Steven Shladover, Scientific American

[9] This problem was identified in aerospace (defence) around 15 years ago – despite lots of research into formal methods, no mature solution has emerged. This supports the theory that a fundamentally different approach based around simulation is required.
[10] "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?" Nidhi Kalra & Susan M. Paddock, RAND Corporation 2016. https://www.rand.org/pubs/research_reports/RR1478.html
[11] https://www.scientificamerican.com/article/what-self-driving-cars-will-really-look-like/

This is particularly true for Deep Neural Networks (DNNs) and other artificial intelligence based approaches where the direct input-output relationship is obscured by the internal complex decision processes and many of the typically used checks and measures, such as limiting design complexity and code reviews, do not readily apply.

If it is accepted that 'typical' safety assessment tools are not directly suitable for assessment of ADS performance and that the driving task complexity means that even huge numbers of test miles is not statistically significant, then it becomes clear that exploring the role of simulation and modelling becomes vitally important.

Simulation is therefore proposed as a substantial complement to real world testing, and is a method of covering a wider range of challenging scenarios, in a repeatable and controllable fashion, and with no risk to the public or vehicle assets. Simulation can also help inform vehicle architecture and design, in optimising sensor locations and guiding specification requirements.

4 TECHNIICAL CHALLENGES

4.1 HIL OPTIONS

Figure 5 shows the ADS processing pipeline, and the SIL/HIL options for testing the ADS or subsets of it.
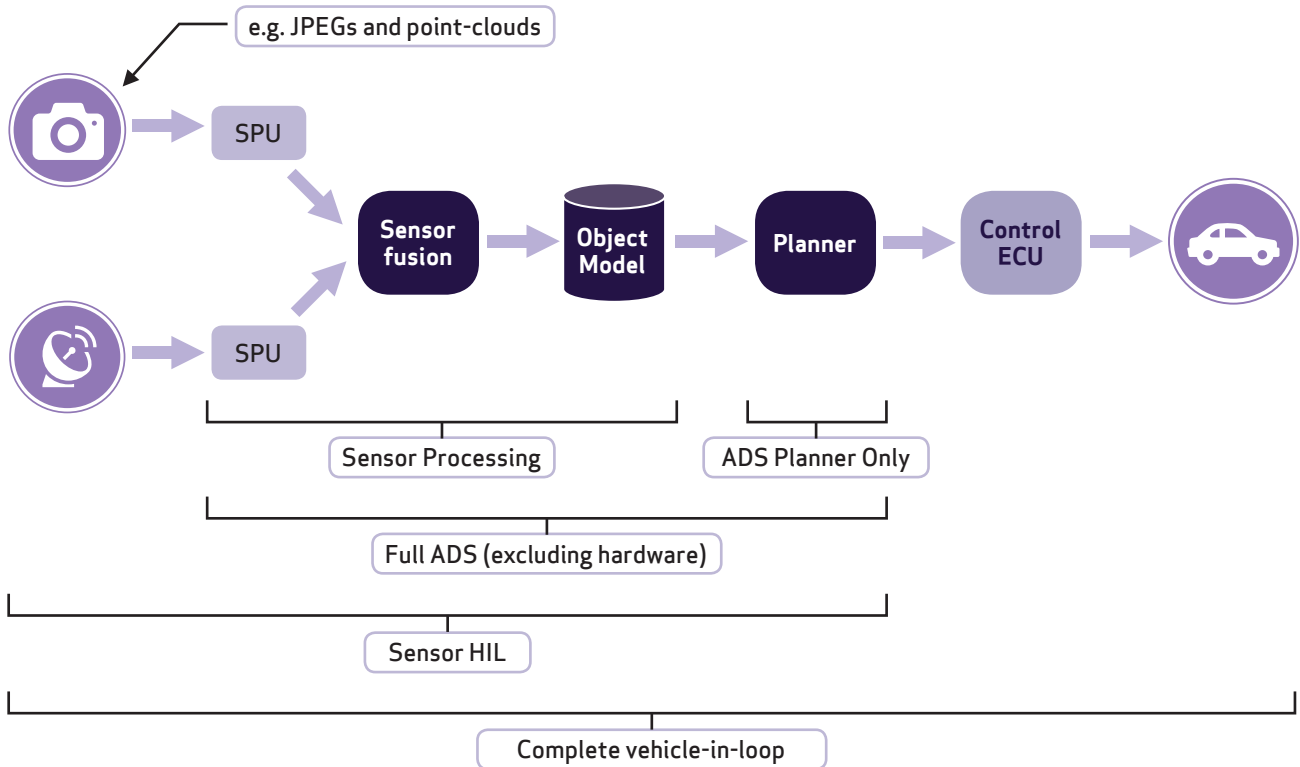
Figure 5: Options for SIL/HIL ADS testing

Following our engagement with the UK stakeholders, there is broad agreement that all options for SIL/HIL testing of the ADS have value for different purposes and points in the engineering lifecycle.

• Sensor processing tests will isolate the algorithms that identify and localise actors, including cars, motorcycles, lorries, bicycles and pedestrians. Given access to the output object model, it is relatively easy to determine the effectiveness of the ADS's sensor processing algorithms.
• ADS planner tests isolate the decision making, avoiding the problems with simulating the output of sensors. Instead, the simulator can keep a simple model of the environment, storing for example just the lane widths and road layout, and maintain an object model of actors within the ADS's field-of-view. This object model is used directly as the input to the ADS planner.
• Full ADS tests are one of the most useful options in this space, as they probe the performance of the complete system without any HIL which prevents accelerated-time, massively parallel testing.
• Sensor HIL tests include the physical sensors without needing a complete vehicle. This option is interesting because we envisage some Tier 1 suppliers might provide a complete ADS (including sensor hardware) to an OEM, and this setup would allow the ADS to be tested.
• Complete vehicle-in-loop (VIL) means placing the entire vehicle into a test facility. A current key area of research is to 'fool' all sensors in such a simulator in order that they provide a synchronised view of the virtual environment (i.e. fooling a sensor that it is looking at a real outside environment when in reality it is in a contained test area).

4.2 IDEAL ADS SIMULATION TEST FRAMEWORK

The simulator and ADS-under-test are only two components of the infrastructure needed to run full V&V in simulation – a diagram of the full system is shown in Figure 6.
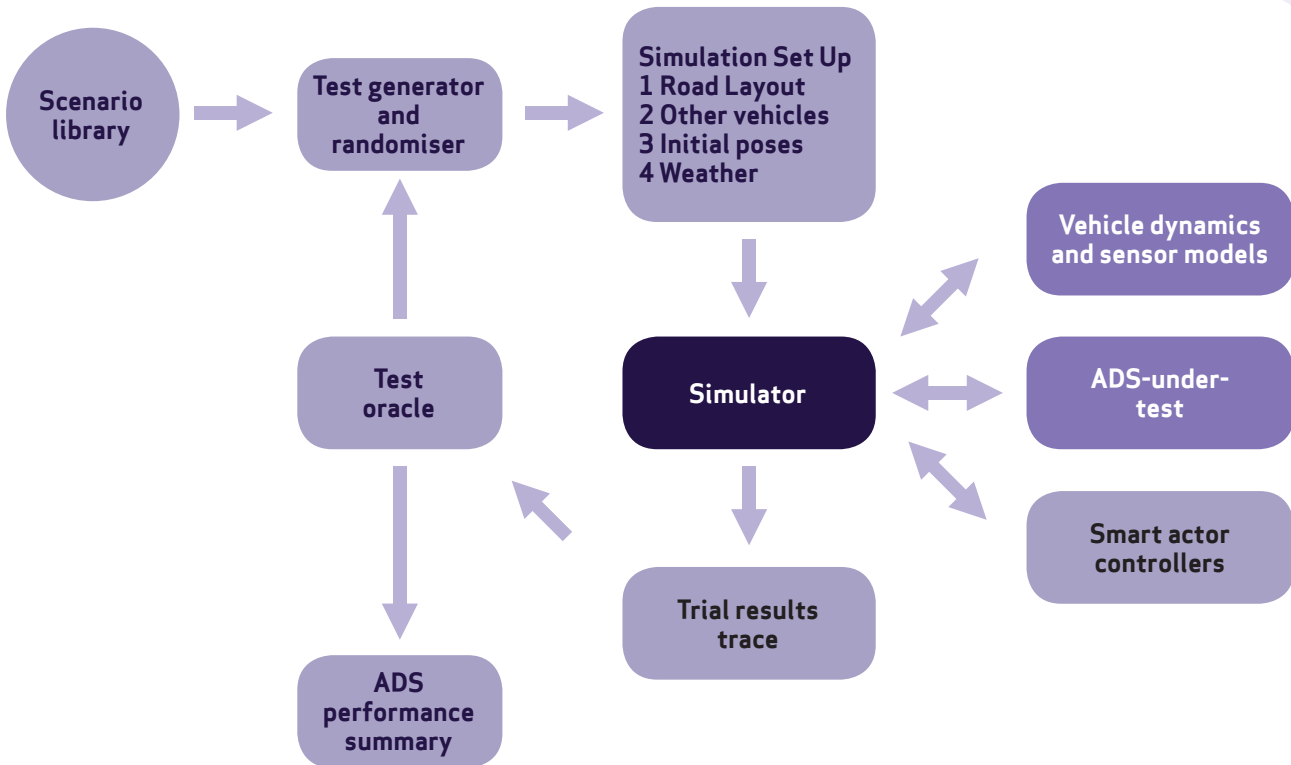
Figure 6: Overview of simulation testing infrastructure

4.3 FIDELITY OF THE SIMULATION

The fidelity of the simulation – i.e 'how well it matches the real world' – is a key issue when performing verification in simulation. If the results obtained in simulation are not the same as results from an equivalent real-world test, then there is limited value in doing the simulation testing. There are three main aspects of the simulation to examine: the vehicle dynamics model, the environment model, and the sensor model.

Modelling vehicle dynamics is well understood by the automotive industry, and can be done to very high fidelity. Some stakeholders have told us that using the most accurate vehicle models isn't necessary when testing an ADS in simulation, so given the cost involved (both financially and computationally), lower fidelity models are often used.

Sensor models are not yet as mature, but are the focus of significant improvement efforts from both sensor suppliers and simulation tool developers. For a "normal" scene, camera models are well understood, and of good fidelity. However, creating highly detailed images is computationally demanding, which means some compromise is often needed in simulations. Further, some aspects of scenes are harder to render, such as reflections, water features, precipitation and weather systems and their effects on the image. Other sensors, such as radar and lidar, are even further away from having efficient, high-fidelity sensor models available (although most ADS simulation tools do have reasonable radar and lidar models included with them).

The environment model includes the roads, surrounding infrastructure, and other actors. There are two cases to consider: if the environment is a digital twin, it should be highly correlated with its real-world twin. If the environment is artificial, realism is still an issue: does the environment include the camber of the road, bumps, potholes and cracks, different types of tarmac and their different reflective as well as friction properties? Does it only allow smooth-sided buildings, or are individual bricks modelled? Does it model the reflective properties of road markings, and allow them to be pristine, faded, or partially erased? Does it model the way suspended traffic lights can sway in the wind?

In both the artificial and digital twin cases, the level of detail and articulation possible in actors is important. For humans especially, many clues to their intentions can be picked up from body language, the way they're walking, where their head is directed, if they've just waved enthusiastically at a friend on the opposite side of the road. All these clues are only available if the simulation environment models the morphology of humans in enough detail, and includes a high-quality model of their dynamic physiology. Similar considerations apply to humans driving a car or riding a bike (can they look over their shoulder?), and even to vehicles – the indicators, front wheel angle, headlights, and horn should all be part of a high-fidelity simulation.

4.4 SCENARIO BUILDING AND UPDATING

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 (which are also referred to as *use cases* or *behavioural competencies*) 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 vehicle fleet of 30 million 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.

Normal driving is important to capture both the base case of controlling a vehicle driving along a road, and the everyday conventions of driving that human drivers do automatically. Examples of such conventions are inching forwards into a traffic jam to prompt other drivers to let you in, or driving slightly closer to a vehicle on a motorway to indicate you'd like to pass them.

The hope is that a common database/library of scenarios can be built, which will contain both international and country-specific scenarios.

Open questions are:

- How many scenarios do we need? Should scenarios be prioritised?
- What should the electronic format for scenario sharing be?
- What level of detail should scenarios be stored at? For example, complete coordinates and speeds, or just "vehicle from the left lane cuts in front"?

4.4.1 Relation to country-specific rules and regulations

It is likely that most edge case scenarios will be international, but many of the everyday-driving scenarios will capture traffic laws and conventions specific to each country or region. A key benefit of a common scenario library is that scenarios can be tagged with a list of the countries they are applicable in. This means that, for the purposes of verifying an ADS in a particular territory, it is easy to extract a test suite of the scenarios that encode the laws and conventions of driving in that territory.

4.4.2 Finding Scenarios

Sources for an initial scenario library could include:

- Expert knowledge:
  - TSC report "Taxonomy of Scenarios for Automated Driving"[14];
  - NHTSA "Federal Automated Vehicles Policy", September 2016[15];
- Pre-existing scenario repositories (both from CR&D projects and from industry);
- Pre-existing collision databases such as the German GIDAS database[16], or the (less detailed but still potentially useful) UK STATS19 data[17], and
- Data recorded from sensor-equipped vehicle fleets, such as fleets operated by local councils, or organisations closely aligned to government, such as the Royal Mail or Highways Agency. Data from these fleets would ideally be processed automatically to extract collisions, near-misses, and other undesirable incidents. Near-misses could be identified using heuristics such as brake pressure applied. Even given this, the candidate scenarios will probably need human review before they can be submitted into the library. The MOVE_UK CAV2[18] project is collecting a highly relevant dataset that could be used for scenario mining.

4.4.3 Refreshing the Scenario Library

As new challenging scenarios become known, these must be added to the scenario library. A key source of new scenarios is real-world collisions involving AVs, and both emergency services and the regulatory framework should mandate data gathering from these.

Other sources are:

- Near-misses involving deployed AVs. Mining scenarios from these faces some of the same challenges as using sensor-equipped fleets, as well as needing collaboration with the OEMs, and addressing any data privacy concerns with recording data on vehicles owned and used by members of the public, and
- Research projects or industry, who may often encounter scenarios that are difficult for prototype ADSs.

Finally, given the importance of capturing relevant new scenarios, it is desirable to allow an electronic submission system for updates to the scenario library. However, it is also important to verify the integrity and relevance of updates, so a mechanism for that would have to be developed.

4.4.4 Dynamic Scenario Generation

The "Test generator and randomiser" shown in Figure 6 is a key part of the system test framework for several reasons:

- Unlike humans, an ADS may be very sensitive to small changes in its environment. This means that while it may work perfectly for a scenario as it exists in the library, a small perturbation of the circumstances may cause it to fail. Examples of this could be a pedestrian wearing different coloured clothes; a vehicle it is overtaking being 0.03m closer to the kerb; or a crossroads having a small offset between the roads "opposite" each other, instead of being exactly in line;
- One goal of the test framework is to find situations where the ADS does not behave as it should do, both to show that such situations exist, and to feed them back to the ADS developer so they can address them. To do this, an effective technique is to focus the tests on areas where the system shows signs of weakness (which is the intention of scenario-based testing in general). However, for a specific ADS, there will likely be indications that tests are probing the limits of its abilities even for tests it passes; an example of this could be failure to identify a pedestrian until the AV is quite close to the pedestrian. A smart test framework could generate more scenarios which are similar to ones where the specific ADS may have difficulties, with the goal of uncovering a true deficiency, and

---

[14] https://s3-eu-west-1.amazonaws.com/media.ts.catapult/wp-content/uploads/2017/04/25114137/ATS34-Taxonomy-of-Scenarios-for-Automated-Driving.pdf
[15] https://www.transportation.gov/AV/federal-automated-vehicles-policy-september-2016
[16] http://gidas.org/en/willkommen/
[17] https://www.gov.uk/government/collections/road-accidents-and-safety-statistics
[18] http://www.move-uk.com/

- In addition to variations of scenarios available from the library, a test generation system could create completely new scenarios. This could be done by taking elements from existing scenarios (such as the road layout, or bicycle actors, or start positions and speeds of other motor vehicles) and combing them with other scenarios. A scenario could also be completely randomly generated. Such generation would require a robust knowledge of the rules of road layouts, and possible starting locations of actors within that.

The dynamic generation of scenarios also encourages the designers of the ADS take into account all possible situations the ADS may encounter in the real world, as opposed to focusing on scenarios from the library.

4.5 INTERFACES

In Figure 5, each of the test options requires two interfaces to the system-under-test: one to inject simulated input to the system, and one to capture the output. Most of these will be software interfaces; the only exceptions are generating input before the sensors (the far left of Figure 5), which will use OTA sensor emulation, and capturing output from the physical vehicle (the far right of Figure 5), which will probably use a rolling road and/or vehicle cradle system.

The software interfaces are challenging because:

- They require the OEMs to open their ADS software (to the extent that these interfaces access internal state), which they may be understandably reluctant to do;
- The different input/output points may require interfaces to different manufacturers' or suppliers' systems, complicating the arrangements;
- While suppliers' algorithms can always be used HIL, running pure software simulations may be tricky. This is because the supplier would have to provide a binary executable, which they may resist as it could allow reverse engineering of their IP, and additionally this executable might only run on very specific hardware;
- For some sensor suppliers, the physical sensor and the SPU may be combined onto a single hardware device, making it hard to insert simulation data between the sensor and SPU;
- Differences in the way different OEMs' ADSs work will mean creating a standard set of interfaces is difficult; the alternative to standard interfaces is that time and money will have to be spent creating adaptors between simulators and OEM ADSs, which is what tends to happen currently, and
- Some respondents have indicated to us that defining any kind of standard interfaces for the simulator to use is a 'pointless exercise', because the technology in this field is moving so fast that an interface which is useful now will be redundant in 6 months' time. However, the right level of abstraction (i.e. not over specifying) should minimise this issue.

4.6 MODELLING OTHER ACTORS

Most scenarios will require other actors in addition to the ego vehicle, such as other cars, or vans, lorries, motorcycles, pedestrians, bicycles and so on. In many cases, it may be sufficient for these actors to follow a pre-defined plan: for example, a pedestrian may sit on a bench and do nothing; another car may be travelling in the opposite direction to the ego vehicle and have no interactions with it.

However, the more interesting and challenging scenarios involve the ego vehicle interacting in some way with other actors. Examples might be to:

- Merge with other traffic when joining a motorway;
- Follow directions of a member of the public organising traffic following an accident;
- Give way to a pedestrian attempting to cross the road at a T-junction, and
- Overtake a cyclist.

In those kinds of situations, the other actor(s) should be actively controlled, so that they behave in a realistic way in response to the actions of the ADS-under-test. Ideally there should be a variety of behaviours that could be assigned to each category of actor, for pedestrians these might include "elderly", "drunk", "in a hurry", and for cars "other AV", "aggressive", or "learner".

This is a significant challenge for building a AV test simulation infrastructure, because the models controlling other actors must be (almost) as sophisticated as the ADS-under-test. The actor models should obey all rules of the road, react appropriately to all other actors, and behave according to their category label (e.g. learner driver). Some of these models exist already in the transport modelling community but currently at a high level.

While getting other actors to obey the rules and conventions of the road can sometimes be achieved with pre-defined trajectories, this is often not possible as the actions of the ADS-under-test are not known in advance. For example, if an actor is approaching a roundabout, whether it should give way to the ego vehicle approaching the roundabout from a different direction may depend on how fast the ego vehicle has driven over the preceding few seconds – which is entirely within the remit of the ADS-under-test.

Examples of scenarios where other actors should react to the ADS-under-test are:

• If the ADS mistakenly veers into the middle of the road, instead of staying in its own lane, oncoming traffic is more likely to steer out of its way than accept a head-on collision;
• If the ADS gives way unexpectedly, perhaps approaching a completely clear junction, a simulated human-driven vehicle might run into the back of it;
• If the ADS hesitates at the entrance to a roundabout, an actor vehicle to its left should probably be assertive and enter the roundabout, instead of giving way;
• When the ADS is attempting to merge into fast-moving traffic, actor vehicles should slow down to make space for it;
• When the ADS is attempting to merge into fast-moving traffic, occasionally a simulated human-driven vehicle should speed up to close the gap and deliberately prevent the ADS from merging, and
• If a cyclist hears a vehicle approaching from behind, they may look over their right shoulder at the approaching car, and therefore drift slightly into centre of road.

In many (but not all) of the cases where an actor should react to the ego vehicle, this reaction will prevent a collision that would have happened had that actor instead followed a predefined trajectory. This may make it seem that having reactive actors leads to a less stringent test for the ADS; however,

1. The decisions of the ADS must be tested in a simulation that accurately reflects the real world: otherwise, you aren't testing its behaviour in the real world, and
2. The impact of the ADS's actions on other actors must be part of the assessment of the ADS – bullying other vehicles out of the way cannot count as acceptable behaviour.

Also, in even moderately-busy traffic environments, it is critical that an ADS anticipates that other vehicles will often give way to it. If it doesn't, the AV will end up acting as a slow-moving roadblock. Therefore, having other vehicles that actually do give way to the ADS-under-test is a key requirement for a validation framework.

Finally, we note that a relatively easy way of running a simulation with multiple cars, each with a smart control algorithm, would be to have the ADS-under-test control multiple vehicles within the same simulation. An extension to provide even more realism is to test the ADSs from several different OEMs at the same time within a single simulation, so they can all interact with each other. However, this workaround does not provide controllers for human-driven cars, pedestrians, bicycles, or any actor types apart from ones where OEMs have submitted an ADS for testing.

4.7 MAPS AND TEST AREAS

Many (if not all) ADSs will require detailed maps of the areas they operate in, with far higher fidelity than simple road network maps of the type used by satellite navigation systems (such maps are often termed HD maps). This presents a challenge for simulation, as these maps must be generated by some external mechanism, such as driving a dedicated car equipped with many expensive sensors along the route several times, and then post-processing the data acquired.

There are two obvious methods for ensuring the ADS-under-test has an HD map of the environment it operates in:

1. Make the simulated environment a digital twin of a real-world location. In this case, the external mechanism used to create HD maps can be applied to the real-world location, and
2. Generate the HD map for a simulated environment in a known data format, either a format used by one of the existing suppliers of HD maps, or a standardised open HD map format.

The drawback of the first method is that you lose the ability to vary junction types and layouts, road sign positions, lane width, level of street lighting, kerb height, road surface, and so on in simulations – which dramatically limits the test coverage possible. The drawback of the second method is that effort has to be spent writing the HD map generator, probably exporting to several different map formats, plus it is unlikely that the HD map format used by every OEM's ADS will be publicly available. In reality a combination of both approaches will be used to maximise their respective strengths.

4.8 ANALYSING RESULTS (THE 'TEST ORACLE')

One of the key advantages of testing in simulation is that many, many simulations can be run in a short time (due to running multiple simulations in parallel, and potentially also running faster-than-real-time, and being able to run back-to-back scenarios instead of needing to physically reset vehicles). Requiring a human to examine the behaviour of the ADS-under-test in every simulation would severely reduce the advantages of simulation: Waymo, for example, run 25,000 simultaneous virtual vehicles[19], which would be prohibitively expensive if each needed to be monitored by a person.

To avoid this an automated test oracle is needed, which can examine a simulation run and tell if the ADS performed acceptably or not. This is a complex question, because collisions are not the whole story: first of all, sometimes a collision will be the correct choice for an ADS, if (for example) it chose to collide with a crash barrier to avoid hitting a pedestrian.

Other factors the test oracle might consider include:

- Did the ADS drive too close to a cyclist, parked car, or other moving vehicle?
- Did the ADS force another actor to take avoiding action?
- Did the ADS have to brake heavily at any point?
- If the ADS braked heavily, was that braking unavoidable, or should it have been able to anticipate the causing factors sooner?
- Did the ADS obey all applicable traffic laws?
- If the ADS disobeyed a traffic law, was that the correct action (for example, to get out of the way of an emergency vehicle)?

There will undoubtedly be many more factors the test oracle should consider, and the unpredictable nature of the real world makes it even harder: did the ADS make the right decision, but there was still a bad outcome because the world evolved in an unlikely way? It can be imagined that developing the test oracle will require significant effort and a process of gradual refinement, based on close comparison of what a human thought of an ADS's behaviour, versus the test oracle's report.

4.9 THE ROLE OF DATA

Within the AV community, there is significant interest in capturing sensor data from suitably-equipped vehicles currently on the road (even if these are not being driven autonomously). Indeed, some commentators believe that ownership of such data will constitute a critical business advantage going forward. This data is useful for:

- Training ML models:
  - This generally means object detection and classification models;
  - For some specific kinds of AI/ML control algorithms, it means ADS planner models;
- Understanding and recording collisions, near-misses and challenging situations that occur in everyday driving;

[19] https://www.theatlantic.com/technology/archive/2017/08/inside-waymos-secret-testing-and-simulation-facilities/537648/

- Understanding and capturing some of the unwritten conventions that drivers in each country use. Examples of this are:
  - How quickly do vehicles move back into their original lane after overtaking on a motorway?
  - Do drivers joining a main road always give way to pedestrians who wish to cross in front of them?
  - How aggressively do drivers push into a traffic flow?
- Capturing sensor data that can be used to validate sensor processing algorithms

It is not clear how useful data is for validating the ADS planner. The desired use-case is that you can take a real-world a situation where your ADS made the wrong choice, make improvements to the ADS software, and use the data from the real-world incident to test the improvement by replaying it. This would avoid the lengthy cycle of having to deploy the new code to a vehicle, take it to a test facility, and recreate the exact scenario that caused the problem.

The challenges here are firstly that data recorded on a specific vehicle can only be used to validate the ADS for a vehicle with an identical set of sensors: the same models of cameras, mounted in the same positions, and the similarly the same lidars/radars/ultrasonic/other sensors.

The second challenge is that, if you replay the same data into a modified ADS planner, as soon as the ADS makes a different decision from the original software the sensor data recorded post critical decision is no longer correct. The different decision will mean the AV is at a different place in space and time, so the fixed replay data is no longer what the ADS would see in the real situation. Therefore replays may have some value in confirming the modified ADS will make a different decision, but will not assure the ADS safely coped with the whole of the problem scenario.

While this sensor data may not be directly useful for validating ADS planners, it can be used to build a high-detail, high-fidelity digital twin of the real world. Doing this requires complex data alignment, fusion, and extrapolation, but having such a digital twin could be extremely valuable for developing and testing ADSs.

4.10 ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Almost all ADSs under development use artificial intelligence (AI) and machine learning (ML) methods, often within the sensor processing components. From a testing perspective, there are two aspects to the use of AI and ML:

1. Validation of the correctness of the complete system (i.e. the ADS, or even the complete vehicle).
2. Verification that subsystems of the ADS meet their specifications.

When validating the complete system, it must be considered as a black-box: you have no access to subsystems, or even knowledge of how they work. Instead, testers should ensure the complete system works as intended – if there was a bug in subsystem A, but this bug was counter-acted by a bug in subsystem B such that the complete system always worked perfectly, that would be acceptable (albeit fragile with respect to future updates). From this viewpoint, the technology used within the ADS does not matter, so there is no issue with the use of AI and ML.

For subsystem verification, the concern is that it is harder to do white-box testing of AI and ML systems, because tracing the code path provides very little information about the processing the system really does. Further, there is some suspicion in the community of the trustworthiness of deep learning systems in particular, as they are not well understood theoretically. Despite this, there has been recent progress on coverage-aware verification methods for deep learning systems[20], and the AI and ML community is aware of the importance of the topic. Also, almost by definition, simulation and modelling does not perform white-box testing so these issues fall outside the scope of this report.

In summary, the details of the software are not particularly important when using simulation to test autonomous vehicles; the challenge of automated driving is greater than that faced by any software system deployed today, and therefore the complexity of the ADS will be commensurate with the task, regardless of how it is implemented. System-level testing is vital given this complexity, and simulation is the best way to achieve test coverage for such complex cyber-physical systems.

[20] "DeepXplore: Automated Whitebox Testing of Deep Learning Systems", Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana, 2017. https://arxiv.org/abs/1705.06640

4.11 OTA UPDATES AND ONLINE LEARNING

Online learning is a way of enhancing machine learning software once it is installed on a vehicle. In normal machine learning, first a dataset must be gathered and labelled – for example, a video data from a vehicle could be gathered, and for every frame, a label added to indicate whether or not the frame contains a pedestrian. Then this data is used to train an ML model. Once the model has been trained, it can be deployed, and will (in this example) output whether or not any input video frame contains a pedestrian. With online learning, data gathered on a vehicle deployed in the field is used to update the ML model running on that vehicle. This might mean a particular AV learning a better model of pedestrians, based on the pedestrians it observes during its owner's commute.

Online learning poses a significant challenge for V&V, and DARPA has recently launched a funding call for research on methods to assure the safety of autonomous systems that incorporate online learning .

In the automotive world, our opinion is that online learning is unlikely to be used. Instead, data will still be gathered by vehicles once they are deployed, but this data will be transferred back to the OEM. Then new data from many vehicles will be examined, used to update ML models as appropriate, and these updated models will be merged into the next OTA software update. This avoids the following two issues with online learning:[21]

1. For data recorded on a vehicle, there are no independent ground-truth labels available. To explain this issue by example, the only way an ADS can recognise a pedestrian is using its existing pedestrian-detection algorithms; therefore, it cannot recognise previously-missed pedestrians and use those to help improve its pedestrian-detection algorithm, and
2. Doing online learning means the OEMs no longer have complete control over the software controlling their vehicles, and this probably represents an unacceptable risk for them.

However, we note point (1) is less true when the temporal domain is considered: if a system predicts the path of some vehicle, and the ADS then observes where that vehicle actually went, that data could be used to update the path-predicting model. Similarly, object identifications when the object is far away could be corrected using the identification from when the object is closer (and presumably, although not necessarily, identification is more reliable).

[21] https://www.fbo.gov/index?id=038cf3096667fd15e72f759e1c7b049d

5 CURRENT SIMULATION PRODUCTS AND PROJECTS

5.1 SURVEY OF OFF-THE-SHELF SIMULATION TOOLSETS

This section looks at current toolsets on the market for the development of automated and autonomous systems for the automotive industry, focusing on ADAS and ADS.

The current toolsets have been split into five categories:

1. ADAS focused simulators
2. Automotive engineering-focused simulators
3. Graphics development, including gaming engines
4. Traffic modelling systems
5. Physical driving simulators

Table 1 lists the leading toolsets providers, product names, country of origin and high level general notes. Detailed analysis for each company and product listed in the table can be found in Appendix 8.5

CATEGORY	COMPANY	PRODUCT	COUNTRY OF ORIGIN	NOTES
DAS focused Simulators	TASS (Siemens)	PreScan	Netherlands (Germany)	Sales and Support office: Stratford Upon Avon
	VIRES (parent:MSC software / Hexagon)	VTD – Virtual Test Drive	Germany	As of May 2017 MSC software parent is Hexagon AB with HQ in Sweden.
	MathWorks	MatLab and Simulink	USA	UK development and sales office in Cambridgeshire
	rFpro	Driving Simulation	UK	
	Gazebo & Robotic Operating System		USA	Gazebo and ROS are maintained by Open Source Robotics Foundation (OSRF) and are both open source development tools for robotics
	Carla Simulator	ADS Simulator	USA/Spain	Open source
	Metamoto	Simulation-as-a-service	USA	Not yet launched
Automotive engineering-focused Simulators	dSPACE	ASM Traffic	Germany	UK – Application office based in Cambridgeshire
	National Instruments	Veristand / PXI / HIL test system	USA	Has UK sales and technical base in Newbury, Berkshire
	IPG	CarMaker	Germany	Interfaces with rFpro
	Mechanical Simulation	CarSim	USA	
	AVL	AVL Team SUITE AVL InMotion	Austria	Application development office in Warwick Integrator of tool sets, working with CarMaker as part of their AVL InMotion platform
	Claytex	Write modules / libraries for Dymola	UK	System integrator of toolset for virtual systems development and V&V. Based in Leamington Spa. Claytex is a distributor of rFpro

CATEGORY	COMPANY	PRODUCT	COUNTRY OF ORIGIN	NOTES
Traffic Modelling Systems	Epic Games	Unreal Engine	USA	Game development tool
	Rockstar Games	Grand Theft Auto (GTA)	USA	Originally developed by Acme Software based in Edinburgh UK
	CVEDIA	SynCity	Netherlands	CVEDIA are AI optimisation company with Autonomous vehicles use case. SynCity is a platform for generating simulated worlds for AVs.
	PTV	Vissim	Germany	Porsche Automobil Holding SE acquired PTV in September 2017
	Aimsun (previously Transport Systems Simulation TSS)	Aimsun Next	Spain	Developing AV modules as part of Flourish CAV1 and HumanDrive CAV2 projects.
	DLR	SUMO – Simulation of urban mobility	Germany	Open-source system developed by the Institute of Transportation Systems, Berlin
	Improbable	Spatial-IO	UK	HQ in London, recently secured £390m investment. Building software enabling efficient large-scale simulation and modelling for science, research, defence applications
	Immense Simulation	ImSim prediction engine	UK	Based in Milton Keynes, Immense create city-scale simulations that deliver enhanced decision making capabilities across the mobility ecosystem
Driving Simulators	Ansible Motion UK		UK	Provide fully immersive Driver in Loop systems for automotive manufactures and Motorsport, Including: F1, NASCAR, IndyCar
	XPI Simulation		UK	Driving simulation capability. Created Warwick WMG's 3xD simulator
	OKTAL		France	Used for the Venturer simulator at the Bristol Robotics Lab. (Oktal also make the SCANeR driving simulation tool which can interface with Aimsun)
	Cruden	Panthera ePhyse	Netherlands	Driving Simulators aimed at automotive OEM's Used by JLR integrating IPG Carmaker and desktop simulation via the Cruden Panthera ePhyse Net external physics interfacing package
	DALLARA		Italy	
	Moog		USA	

Table 1: Summary of current toolset developers, products and company locations

5.2 CURRENT UK CAV FUNDED PROJECTS RELEVANT TO SIMULATION AND MODELLING

Feasibility studies and Collaborative Research and Development projects for both CAV1 and CAV2 competitions have included elements of validation using simulation. A non-complete selection of these projects are listed below (does not include CAV3):

5.2.1 CAV1 Feasibility Study

• **VEDAS** (Virtual Validation Environment for Driver Assistance Systems), project lead AVL Powertrain UK.

5.2.2 CAV1 CR&D

• **INTACT** (Innovative Testing of Autonomous Control Techniques): Project partners: RDM and WMG. Project focused on the testing and evaluation of ADS, using a Vehicle-in-the-loop simulator.

5.2.3 CAV2 CR&D

• **SAVVY** (Smart ADAS Verification and Validation Methodology): Project lead is AVL Powertrain UK.
– SAVVY is a CR&D follow-on project from three previous Innovate UK funded feasibility studies. SAVVY will use a combination of simulation and test technologies including: AVL tools, Vitaq.ai test automation tool suit from Vertizan, Deep learning AI technology from Myrtle Software[22], WMG's 3xD Simulator, and real-world testing facilities at Horiba-MIRA. Aim is to develop a cost effective and scalable approach to testing next generation SAE 3+ systems.

• **CAPRI**: Project lead AECOM.
Verification and Validation the safety and security of the next generation of PODs for both on and off-road environments.

• **HumanDrive:** Project lead: Automotive OEM.
Key innovative aspects will be the development of an advanced vehicle control system, designed to allow the vehicle to emulate a "natural" driving style using machine learning. Significant digital validation work package led by Cranfield University using MUEAVI instrumented road and a digital model.

• **MuCCA** (Multi-Car Collision Avoidance): Project lead IDIADA.
An important element of the project is the development of a simulation environment that mirrors the real-world testing environment, to allow the novel ADS algorithms to be tested and verified prior to physical trials.

5.3 NEW DEVELOPERS AND NON-TRADITIONAL AUTOMOTIVE DEVELOPERS

New entrants into the automotive ADS sector are developing in-house tools for verification and validation of their systems. UK ADS developers interviewed during this process, confirmed they have not found, mature, off-the-shelf tools to aid their V&V activities, therefore they have developed internal tools.

It is public knowledge that companies such as Waymo, Apple, Uber, Tesla are developing autonomous control systems for driverless cars; however, public information regarding their V&V framework is relatively limited. The limited information available on Waymo's test process is detailed in the section below.

5.3.1 Waymo

Waymo has a mature test framework, evolved over 7 years of developing ADSs. It consists of a private physical test centre codenamed **Castle** and a sophisticated simulator codenamed **Carcraft** (shown in Figure 7), with targeted transfer of scenarios between them[23]. Figure 8 shows a schematic of the test process, with situations that prove challenging for the ADS identified at each stage.

Waymo places increasing reliance on simulation, with 25,000 simulated AVs operating continuously, allowing the ADS to cover billions of simulated miles per year. According to a Waymo engineer, *"The vast majority of work done – new feature work – is motivated by 'stuff' observed in simulation"*[23]. Carcraft was developed in-house, and Waymo uses it mainly to test the ADS planner, rather than for testing sensor processing.
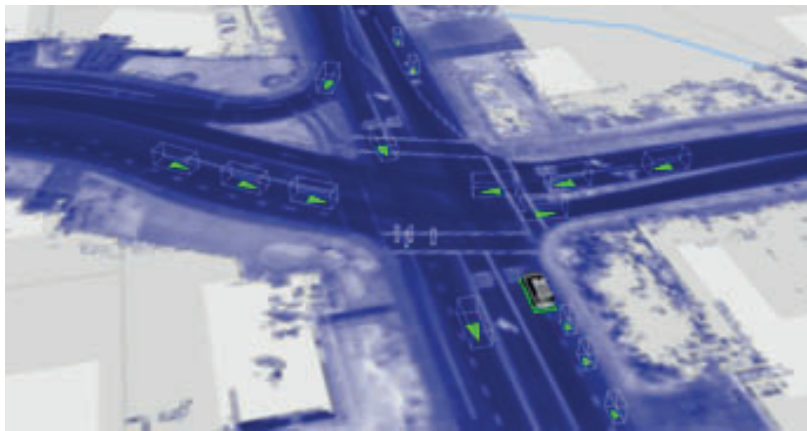
Figure 7: View of Waymo's Carcraft simulator[23]

Castle is used for setting up and running scenarios in a controlled environment, but one which replicates the real world, and can include pedestrians, bicycles, broken-down cars, roadworks, and other unusual obstacles. The scenarios are chosen to probe the ADS: for example, ones where it has to brake hard, or interact in close proximity to other vehicles, or has a restricted view of obstacles and vehicles moving in the scene.
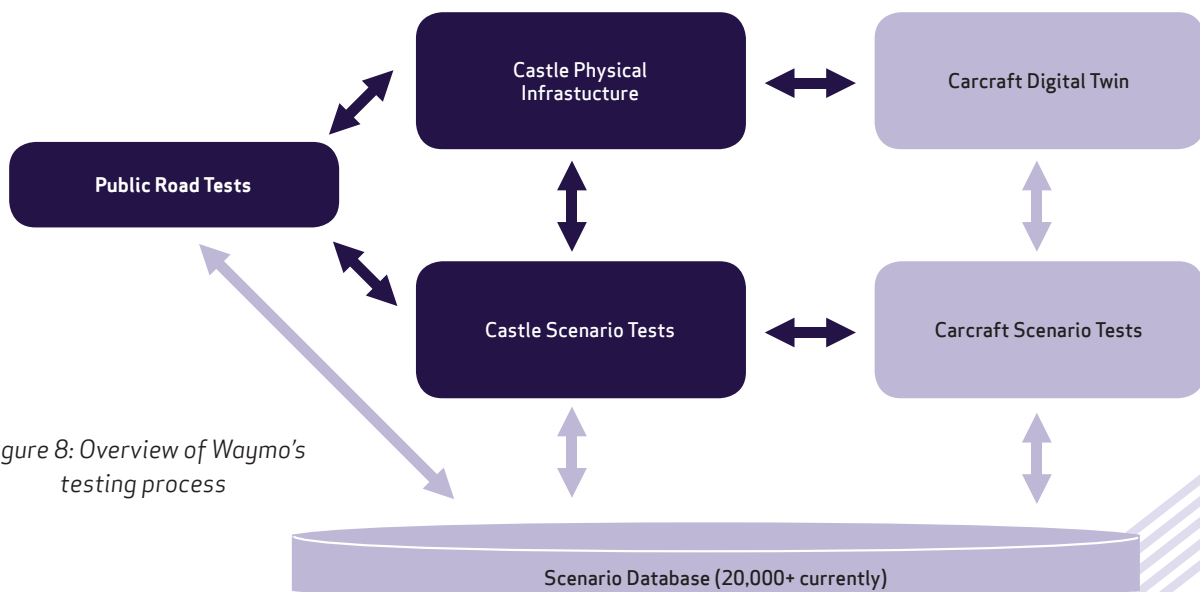
Figure 8: Overview of Waymo's testing process

```
Public Road Tests ⟷ Castle Physical Infrastucture ⟷ Carcraft Digital Twin
                     Castle Scenario Tests ⟷ Carcraft Scenario Tests
                     Scenario Database (20,000+ currently)
```

23 https://www.theatlantic.com/technology/archive/2017/08/inside-waymos-secret-testing-and-simulation-facilities/537648/

6 SUGGESTED INFRASTRUCTURE TO ALLOW ADS TESTING IN SIMULATION

This chapter describes how simulation and modelling can be used to test the behaviour of an ADS. The framework described will be useful both for the development of ADSs, and for assuring their safety from a regulatory point of view (see Section 6.6 for more details on the regulatory opportunities).

This study focuses on testing on-road behaviour, so does not consider other aspects relating to the safety of AVs. These other aspects include the HMI, moving the vehicle to a minimum risk condition when it exits its operational design domain (ODD), mechanical or software redundancy in systems, ensuring the ADS software is up-to-date, and many others.

6.1 PHASE 1

Given the structure of a simulation testing framework discussed in Chapter 4, we propose a two-part implementation effort, with a relatively well-defined Phase 1 followed by a more ambitious, loosely-specified Phase 2. The intention of Phase 1 is to be a short-term project with goals that can be achieved using existing technology, given a sufficiently focused effort. Figure 9 gives a high-level picture of the test process in Phase 1.

Controlled test facility with a digital twin

Conducted in Physical World
Conducted in Simulation

Perception layer: Does the sensor suite provide the right output?

Y/N

Does the ADS make the correct decision based on data received?
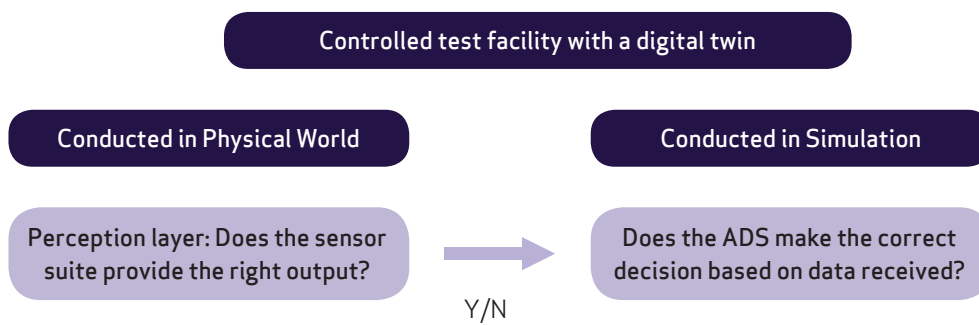
Figure 9: Overview of test process for Phase 1

6.1.1 Overview and Reliance on ADS Modularisation

The Phase 1 infrastructure performs separate testing of the ADS's sensor processing (which is done physically) and planning (which is done in simulation). The details of these two test processes are given below. Separate testing is feasible and advantageous if there is a clear interface between the two components, and this interface represents data as an object model, as shown in Figure 10.

Sensor processing   Object model   Decision making / Path planning

Figure 10: Separation of an ADS into sensor processing and planning, with an object model providing the interface between the two modules. Sensor processing picture © TU Delft, planning picture © Waterloo Autonomous Vehicles Lab.

For the simulation tests, significant simplifications in the test infrastructure are made possible by allowing sophisticated scenario definitions. Each scenario will be refined by hand to specify which variables are randomizable, what the behaviour of each actor within the scenario should be, and the acceptable behaviour of the ADS during the scenario. Further details of the scenario definition are given in Section 4.2.
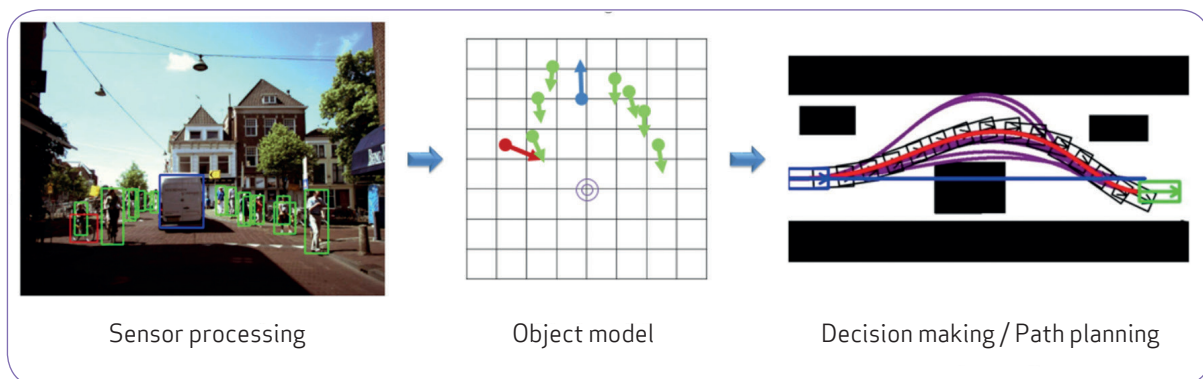
6.1.2 Sensor Processing

Sensor processing will be tested by instructing the AV-under-test to follow a fixed route within a controlled test facility or proving ground. This route will have a number of known targets along it, such as other vehicles, pedestrians, cyclists, animals (probably replica), temporary speed limit signs, and so on. Easy access to the object model output by the ADS's sensor processing means it is possible to record what the ADS thought it saw, and compare this to the targets.

Using a proving ground allows equipment to be installed to record the ground-truth of where every target was at every time, and this can be compared with the object map log. A clear and objective score can be created for the ADS, where false negatives (i.e. failing to see a target) should carry a higher penalty than classification or position errors.

The key advantage of testing sensor processing physically is that it avoids the need for highly accurate sensor and environment models. It is not possible to test in arbitrary weather conditions, but many proving grounds provide some ability to generate weather patterns such as rain, and a wide variety of targets can be used, with different colours and textures, and different backgrounds along the test route, to gain as much test coverage as possible.

6.1.3 Scenario Definition and ADS Planner Tests

The second pillar of Phase 1 is the ability to test many variants of each scenario from a database in simulation. This is made possible by putting effort into hand-crafting each scenario, allowing key aspects of the test to be encoded into the scenarios. The intention is to start with a small set of scenarios (perhaps 100) to make this feasible, but allow the scenario library to grow in the future.

Each scenario will specify the road layout, actors present, and the start positions and velocities of all actors and the ego vehicle. To allow for simple randomisation of scenarios, the definition will include permissible ranges for the start positions and velocities of the actors. An extension would be to allow a variable number of actors.

To allow actors to respond to the behaviour of the ADS, as discussed in Section 4.6, scenarios will provide a simple rule-based scripting language for actors. This will probably specify a primary trajectory for each actor, together with alternates to be executed if some condition is fulfilled.

Instead of requiring a general-purpose Test Oracle, we intend the acceptable behaviour of an ADS within each scenario to be part of the definition of that scenario. These behaviour boundaries will describe a geospatial bounding box for the acceptable trajectories for the ADS to choose, as well as defining the maximum time allowed for the ADS to reach its goal location, perhaps locations where the AV must slow down or stop, and other criteria.

The actual testing will use a low-fidelity simulator similar to Mathworks' Autonomous Driving Toolbox[24], where the environment is simplified to just the road layout (including junction layout), number and width of lanes, and traffic signals. Sensor processing is bypassed completely, and instead the location and velocity of actors within the simulated AV's field-of-view will be inserted directly into the ADS's object model. Vehicle motion however should be simulated to a higher degree of fidelity, so the control commands output from the ADS will use a vehicle dynamics model to ensure the movements of the AV are plausible.

To ensure the ADS has access to HD maps it needs to be able to operate within a specific area (see Section **4.7**), tests will be conducted in the digital twin of a controlled AV test facility.

The performance of an ADS will be determined mainly by the scenario boundaries, but also by threat measures, which would measure quantities such as the minimum of time-to-collision, and peaks of acceleration and jerk, during the scenario. Running scenario simulations would be completely automated, and a summary result would be produced after a specified number of variants of a specified set of scenarios had been completed.

[24] https://uk.mathworks.com/products/automated-driving/features.html#scenario-generation

6.1.4 Challenges and Limitations of Phase 1

The limitations of the proposed test infrastructure are:

• Testing the sensor processing and decision making separately means that bugs due to interaction between the two may be missed;
• Testing the sensor processing only physically makes it much harder to obtain the desired coverage of weather conditions. It also imposes limits on how many different combinations of road type, other vehicle types, pedestrian types, weathers, object colours, and so on, can be tested;
• Physical testing of sensor processing is slower and more expensive than simulation testing, and
• Basing the simulated tests in a digital twin of a real environment means each scenario must choose from a limited set of junction types and layouts, road sign positions, lane width, level of street lighting, kerb height, road surfaces, and so on.

We would like to highlight two technical risks, first that scripting language(s) to specify both the behaviour of actors, and the acceptable behaviour of an ADS in a given scenario, will be complex. This will affect the cost and duration of any development project, but will also make it harder to add new scenarios to the library going forward.

The second technical risk is that the trade-off between having enough scenarios for good coverage, and the amount of hand-crafting needed for each scenario, could lead to an overspend or functional deficiency in the infrastructure.

Finally, the most significant challenge with Phase 1 will be working with industry to ensure that the ADS-under-test has a read/write interface to the object model. The object model logically sits between the sensor processing and decision-making modules, but if the architecture of a particular ADS doesn't have a clear separation between those modules, it may require a significant amount of work by the ADS developer to create the required interfaces. Further, ADS developers will require stringent safeguards over their IP, as they will have to supply executable copies of a software system that is highly commercially valuable.

6.1.5 Rationale for Phase 1

The key driver for Phase 1 is that current technology levels are not sufficient for test results in a simulated environment to fully match results in the corresponding physical environment. Phase 1 overcomes this by separating testing of planning and sensor processing, thereby allowing valid test results for the planning component to be generated in simulation with existing technology. This opens the door to all the advantages of testing in simulation, primarily the ability to cover many more scenarios within a given time, but also much greater safety, control of the environment and repeatability.

The use of current technology and clear, achievable goals means a verification framework can be created on a similar timescale to the first releases of SAE level 3 cars, so that such cars can be put through a robust and independent verification process.

The other advantage of Phase 1 is that early engagement with the challenges of AV verification in simulation will lead to much faster progress in the field, and will give significant insight to where more fundamental research should be focused.

6.2 LONGER-TERM PROPOSALS

6.2.1 Motivation

Technology in this field is developing rapidly, and simulation solutions must keep pace. It is expected that Phase 1 will be a learning process, where close engagement with stakeholders and lessons from using a simulator in production will result in a clearer idea of the way forward.

A key motivation for improving on Phase 1 is over-the-air (OTA) updates: validation must be performed on a timescale compatible with the OTA update release cycle. If validation is not achievable on this timescale, either software that is not fully tested will be deployed, or else updates (which potentially contain safety fixes) will not be deployed since they have not been fully tested and approved for release. At this stage it is not clear what this release cycle is, but months or even weeks is possible[25]. Phase 1 relies on a relatively lengthy physical test process, so a goal for the longer term is to move to more simulation. Figure 11 shows an illustration of how we see the increase in the use of simulation (and corresponding reduction in the amount of physical testing needed) leading to safer AVs over time.
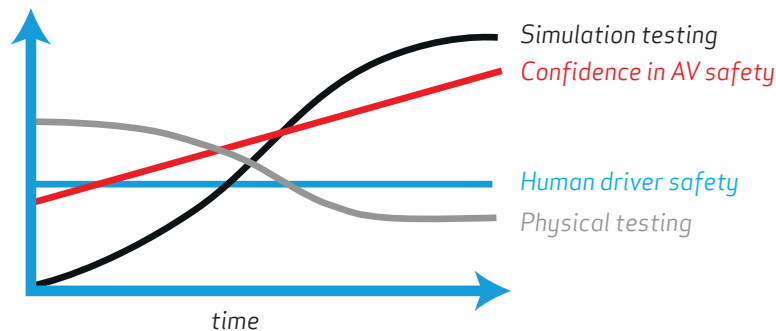

Figure 11: Stylised illustration of how increasing levels of simulation lead to higher confidence in AV safety over time. Note simulation can never completely replace physical testing.

The other limitations of Phase 1 are covered in Section **6.1.4,** and future work will ideally address all of them.

6.2.2 Longer-Term Suggestions

Potential areas for longer-term development are:

1. Phase 2 – develop the full system as shown in **Figure 6** from Section **4.2** (see below for more details).
2. Understanding the architecture and technologies commonly used in ADSs to help focus validation work. As autonomous driving is very new, little real-world data is available to identify the kinds of circumstances that cause AVs to make mistakes. An improved understanding of the underlying technology would allow a better intuition to be formed about this, which would indicate where testing should be focused to reduce real-world collisions.
3. Research into verification and validation of AI and complex systems is central to improving the safety of AVs, as even extensive, randomized, scenario-based testing cannot uncover all bugs in a system. Formal verification methods are one avenue to allow much higher confidence to be achieved on the correctness of subsystems.
4. Research into high-fidelity sensor modelling, to allow sensor tests in simulation to be fully representative of real-world behaviour.

---

[25] Tesla software on the Model S was updated every 11.5 days on average over the 44 month period to February 2016
(http://blogs.lse.ac.uk/businessreview/2017/01/31/with-software-updates-tesla-upends-product-lifecycle-in-the-car-industry/)

6.2.3 Phase 2 Overview

Phase 2 implements the full system shown in Figure 6 for simulation testing. While this will not remove the need for physical testing of an AV, testing end-to-end in simulation allows much more confidence to be gained from the virtual tests.

As discussed in Chapter 4, there are many complex aspects of the full system. The test oracle, smart actor controllers, and scenario generator are components that current tools only provide in a limited form. However, that will likely change in the future, so the Phase 2 infrastructure may be able to leverage off-the-shelf tools. Additionally, effort is needed to enhance and optimise sensor models, and advances will come from both industry and academia.

6.2.4 Challenges and Limitations of Phase 2

On a technical level, the main risk with Phase 2 is that the sensor and environment models' fidelity will not be sufficient for test results to translate to the real world. This is a critical issue, and although models are improving rapidly, they should be an area of focus for Phase 2.

In the current sensor simulation ecosystem, models for simulation are developed by two main groups: the sensor manufacturers themselves, and the developers of simulation tools. If sensor manufacturers create the model, it must take environment data as input, and the interface definition for this may be different for every different simulation tool (and potentially such interfaces will not even exist). On the other hand, if the simulation tool developers also create sensor models, they will have to be generic rather than reflecting the performance of a particular sensor model. If sensors from a vendor include custom, integrated sensor processing algorithms, these cannot be replicated in generic models provided with the simulation tools. Reaching a consensus on how sensor models should be created and interfaced with simulation tools would be a useful output of Phase 2.

There are also significant collaborative challenges with Phase 2, again revolving around the sensor models and integration of sensor processing with the rest of the ADS. For Phase 1, the interface point is the object model, but in Phase 2 the necessary interface is between the physical sensor and the sensor processing unit (SPU) – these are shown graphically in Figure 5. While we expect the object model interface to an ADS to be under the control of the OEMs, the sensor and SPU will often be provided by a Tier 1 or Tier 2 supplier. This means gaining agreement on the interfaces will involve more organisations, with complex relationships, and therefore will potentially be even more challenging than agreeing interfaces for Phase 1. To add a technical challenge to this, sometimes the SPU will be physically integrated into the sensor, which does not preclude the suggested Phase 2 infrastructure, but will complicate the process.

Finally, we note that providing robust and well thought out mechanisms for protecting industry's IP will be a key step for obtaining agreement on allowing access to sensor models and automated driving systems.

6.3 CONNECTIVITY AND CYBER SECURITY

Connectivity is out of the scope of this report. However, we feel it should be mentioned, because both V2V and I2V/V2I communication will likely be a core aspect of future AV deployments. At least one of our interviewees made the point that including connectivity and the resulting complex interactions between agents will massively increase the complexity of any simulation.

A related aspect is cyber security, which is important even for standalone AVs, but becomes a critical issue when connectivity is introduced. However, cyber security is similarly out of the scope of this report.

6.4 SAE LEVELS 3 AND 4

This report is concerned with the verification of autonomous decision making, i.e. without any human supervision or interaction. However, Level 3 and 4 AVs will sometimes encounter situations that they cannot cope with, and must either stop the vehicle, or hand control over to a human. Early research has suggested these handover periods are high risk relative to "normal" driving by either the ADS or a human [26][27], so methods for assuring safety during these transitions should be investigated as a matter of urgency. This includes aspects such as:

• Verifying there is no way for a L4 vehicle to leave its ODD when driving autonomously;
• Research to establish safe mechanisms and notice periods for transfer of control (in both directions) between the ADS and a human driver;
• Other HMI aspects, for example the best way(s) of clearly indicating whether a vehicle is under autonomous or manual control, to both internal occupants and external road users; and
• Techniques using simulation to verify adequate HMI has been implemented on an AV.

6.5 INTERACTION WITH PRIVATE-ROAD TESTS

One of the few points that every stakeholder we interviewed agreed on is that real-world testing of AVs will still be essential, regardless of improvements in simulation test tools.

Physical testing is necessary firstly because a simulation can never be perfect – the real world will always be different in some way – and secondly because it is important to test the complete system as opposed to isolated components, due to complex emergent behaviours. It is expected that physical tests will follow a similar path to the current automotive model, with private road tests to start with, and supervised public road trials as the vehicle nears production.

We see the simulation test infrastructure interacting with private-road tests in two main ways:

• Use simulation to focus physical tests
• Use physical tests to verify the model

6.6 POTENTIAL SYNERGIES BETWEEN DEVELOPMENT AND REGULATORY USES OF SIMULATION

Both system developers and regulators want to ensure the safety of AVs. There is a clear potential for both users to benefit from sharing a scenario database with standardised formats, and from using some of the same software tools.

There are however some differences in needs:

• Regulatory use would potentially need a smaller, more stable set of scenarios & simulation infrastructure;
• Developers would likely want to move beyond the minimum level of performance mandated in the regulation;
• Developers might want to interface to less-mature ADSs for development purposes;
• Developers would likely need faster test cycles and the ability for the test scenarios to integrate with other product design lifecycle tools (e.g. link to requirements in IBM Doors for example), and
• The output / reporting requirements may differ with regulators needing a simple high-level view of the system compliance, and the developers needing a much more detailed understanding of the system deficiencies.

[26] Venturer Planned Handover report, May 2017 http://www.venturer-cars.com/trial-1-results/venturer-trial-1-planned-handover-technical-report/
[27] Handover issues in autonomous driving: A literature review. Morgan, P., Alford, C. and Parkhurst, G. (2016). http://eprints.uwe.ac.uk/29167/

6.7 DEVELOPERS AND REGULATORS WORKING TOGETHER

Regulators and developers need to continue their discussions to agree on prescriptive or voluntary legislation.
This should draw upon best practise internationally, i.e.

NHTSA have this year switched to voluntary guidelines: *"The policy creates a less burdensome, flexible framework that allows companies to innovate first, focusing on performance based outcomes to improve safety and mobility, instead of focusing their resources on passing bureaucratic hurdles."* [28]

- Regulators and developers will likely need to accept a moving regulatory target. This is in terms of both the set of scenarios and the simulator implementation.
- Developers will likely have to provide some degree of access for regulators and/or independent reviewers to their ADSs (perhaps limited to black box level).
- In the global AV market, regulations will have to be harmonised internationally.

[28] https://www.nhtsa.gov/manufacturers/automated-driving-systems#automated-driving-systems-faq

7 CHAPTER SUMMARIES & CONCLUSIONS

The main takeaways from the key technical chapters are as follows:

Chapter 3 – The Role of Simulation in the Changing Verification Challenge

- Current mechanisms for maximising road safety focus on three areas:
  - Vehicle safety
  - Operational safety
  - Environmental safety
- The introduction of AVs merges the vehicle and operational safety areas
- Limited regulation and standards currently exist for AVs, ISO26262 provides the automotive industry with a reference point for functional safety associated with safety-critical situations arising from hardware and software malfunctions. This does not address the so-called Safety of Intended Functionality (SOTIF) whereby performance limitations or losses of key AV system elements (e.g. Lidar range) could be considered in the same manner as a system failure. ISO/AWI PAS 21448 seeks to address this failing.
- ISO26262 offers some (limited) guidance on the use of software tools which may be applicable to simulation software tools.
- AV software is sufficiently complex to make it impossible to accurate specify exactly how an AV should behave in all scenarios.
- Without simulation, there is much debate as to how many miles should be driven before AVs are proven to be safe.
- Simulation is likely to be a substantial complement to real world testing and may provide a means to covering a wide range of challenging scenarios in a repeatable and controllable fashion, with no risk to the public or vehicle assets.

Chapter 4 – Technical Challenges

- The project has proposed a target architecture for an ADS simulation test framework.
- A simulation framework features three key elements:
  - A vehicle dynamics model – well understood by industry, low fidelity models are sufficient for ADS testing
  - Sensor models – not yet mature
  - Environmental models – encompassing 'digital twins' of real world locations. Questions exist as to the level of fidelity needed in environmental models
- Scenarios (or Use Cases) are a key element in the simulated testing of AVs.
- Scenarios are central to assuring the safety of AVs: they are easily understood by humans, are easy to convert to automated test cases, and can capture the characteristics of situations that are challenging to an ADS.
- Scenarios should cover both normal driving and edge cases.
- Sensitivity testing of scenarios should be incorporated into simulation testing, to identify areas of weakness in ADS solutions that would not normally be found through physical testing.
- There should be an element of randomisation in the selection of scenarios that an ADS-under-test is subjected to, to prevent 'gaming' of the test.
- Standardised interfaces will be required within any integrated test environment; there are several open issues that need to be resolved:
  - How many scenarios are required?
  - What should the electronic format for scenario sharing be?
  - Should certain scenarios be prioritised?
  - What level of detail should scenarios be stored at? (e.g. complete coordinate and speeds, or just "vehicle from the left lane cuts in front")
- A range of sources for scenarios already exist, there is a need for industry and regulators to collaborate and share insight to enable a central scenario repository to be established. These should be curated by a neutral third party.
- As new scenarios are discovered, these should be added to the central scenario repository.
- Near misses involving AVs should be included in the central scenario repository.
- A range of actors should be considered in each scenario, the interaction of the vehicle under-test with these other actors should be considered, together with the impact of the ego vehicles actions on the environmental model.
- A Test Oracle to analyse the results of testing is recommended, this removes the need for a human to monitor each vehicle under test, thereby speeding up the process, removing human error and enabling more scenarios to tested in a shorter time.

- A solution to providing map data to AVs in simulation is required, this could potentially be a digital twin of a standard test area.
- Simulation based testing is expected to be important in evaluating OTA updates pragmatically and enabling key updates to be pushed out to a growing fleet of AVs.

Chapter 6 – Suggested infrastructure to allow ADS testing in simulation

- The project has proposed both short term and longer term interventions that could be made to provide a framework for the test and simulation of AVs.
- Phase 1 addresses the short term need to provide regulators with a means to assess the performance of an ADS, it relies on
  - sensor processing being evaluated in the real world at a controlled test facility or proving ground
  - a simulated environment (e.g. a well characterised digital twin of a test facility) to evaluate the decision-making components of the ADS
  - a central repository of scenarios that the ADS will be tested against (within the simulated environment)
- Phase 2 addresses the longer term needs of regulators and industry, building upon Phase 1. It integrates sensor testing into a simulated environment that will ultimately enable the timely test and evaluation of OTA updates to AVs.
- It is anticipated that in time the dependence on physical testing will decrease, whilst the extent and confidence in simulated testing grows.
- It is expected that an element of physical testing will always remain as part of the evaluation of AVs.

7.1 CONCLUSIONS

There is clearly a fast-evolving market for simulation tools in the CAV domain. Due to the large number of potential competing players adapting products from related areas, the market can expect to see consolidation and continuing alliances between tool vendors / providers to perform certain 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.

For the regulatory purpose, two phases of simulation testing have been proposed:

> **Phase 1** infrastructure performs separate **physical testing** of the ADS's **sensor processing** and simulated testing of the **planning** tasks. This is a practical first step but potentially has a weakness associated with any characteristics of the sensor processing that then follow through to a linked failure of the planner. Testing them separately rather than holistically would not uncover this type of failure.

> **Phase 2** implements the full system of **simulation testing.** While this will not remove the need for physical testing of an AV, testing end-to-end in a validated simulation allows much more confidence to be gained from the virtual tests and a quicker coverage of test scenarios.

Despite the stakeholder interviews, it is still unclear if the key simulation tools for AV design will be primarily bought off-the-shelf or developed in-house by ADS developers. For regulation purposes, tools or frameworks that are independent of the system developers are likely to be required.

There is general relevant UK strength in this area, particularly the motorsport industry, complex engineering, engineering testing services, computer games industry and the wider software industry. More specifically, the stakeholder work underpinning this study shows there are several UK organisations with relatively mature plans for contributing to simulated V&V for CAVs.

8 APPENDICES

8.1 GLOSSARY

Acronym	Concept (alphabetical order)	Description
	ADAS Simulator (ADS Simulator)	A simulation tool geared towards developing and testing ADAS and ADS systems. This means it must provide high-fidelity sensor models, as well as a 3D environment, physics engine, and vehicle dynamics models.
	ADS-under-test	The ADS being validated by the AV test infrastructure.
AI	Artificial Intelligence	A collection of algorithms and methods for creating computer software that makes decisions in an intelligent manner. Often machine learning is viewed as a sub-field of AI.
ACS	Automated Control System (Autonomous Control System)	Same as ADS.
ADS	Automated Driving System	The software controlling a L3-L5 AV. In other words, the smart software that processes sensor data and makes driving decisions. (SAE J3016 term)
AV	Autonomous Vehicle	A L3, L4 or L5 vehicle being controlled by the ADS
	Behavioural Competency	Same as Scenario. Term used by NHTSA.
	Black-Box Testing	Testing a system without having any knowledge of how it is implemented.
	Certification	Procedure in which authorised party assesses and verifies the attributes of a product or service in accordance with regulation or standard.
	Connected Vehicle	A vehicle that uses some form of mobile networking to connect to either other vehicles, roadside infrastructure, or both. By implication such a vehicle is not also autonomous.
CAD	Connected and Autonomous Driving	The task of controlling a CAV.
CAV	Connected and Autonomous Vehicle	A vehicle that, as well as being autonomous, uses some form of mobile networking to connect to either other vehicles, roadside infrastructure, or both.
CDV	Coverage-Driven Verification	A testing process that attempts to cover the complete space of possible inputs to the device-under-test, to ensure the correctness of the DUT.
	Decision Maker	Same as Planner.
DL	Deep Learning	A form of machine learning that uses neural networks with many hidden layers. Deep learning requires a lot of training data compared to other ML methods, but has proven extremely effective over the last few years.
DUT	Device-Under-Test	In a simulation-based test infrastructure, the device (including pure software devices) that is being tested.
	Digital Twin	A virtual environment that replicates a physical environment exactly.
DIL	Driver-in-the-loop	Test infrastructure where a human driver interacts with the simulator and any other devices-under-test.
	Driver Model	Generally used to mean the ADS – the model (in Simulink terms) for driving a vehicle. Occasionally also used to mean a model of human driver behaviour.

Acronym	Concept (alphabetical order)	Description
	Driving Simulator	A simulated intended for use by humans – it simulates driving a car to the human. Therefore, it will contain screens to present the virtual environment to the human, and a steering wheel and pedals so the human can control the car.
DDT	Dynamic Driving Task	The task of driving a car, including detecting and responding to events and objects in the external world. (SAE J3016 term)
	Edge Case	A situation or scenario which will only occur very rarely on the road, but which presents particular challenges for an ADS.
	Ego Vehicle	In a simulation, the vehicle you are driving or controlling. In more sophisticated simulations, there may be more than one ego vehicle, as you may want to actively control multiple vehicles.
	Emulation (Emulator)	Essentially means the same as Simulation, except when used specifically to mean OTA Emulation.
	Formal Verification	A technique that uses mathematical modelling of the behaviour of a system to prove certain properties. An example property might be: the software never engages reverse while the vehicle is travelling forwards.
	Hard Test	Same as Physical Test
HAD	Highly Automated Driving	Same as Dynamic Driving Task
HAV	Highly Automated Vehicle	Generally means a L4 or L5 autonomous vehicle
HD Map	High Definition Map	A map containing more detail than the simple layout of the road, in other words, more detail than a standard satnav map. This detail may include, for example, the exact width of every lane, the location and type of all lane markings, the speed limit, and the location of street signs and traffic lights.
HIL	Hardware-in-the-Loop	A simulation setup where one or more components of the DUT run on their final deployment hardware. An example setup could have the simulator itself and the ADS code running on normal desktop computers, but the ADS output being sent to the real vehicle controller ECU, and output from that ECU sent back into the simulator.
	Homologation	Gaining regulatory approval to market a particular vehicle model.
HMI	Human-Machine Interface	The user interface used to communicate with and control a machine (in this context, the machine is an autonomous vehicle).
	Lidar	A type of sensor that uses time-of-flight of laser beams to estimate the range to objects in its field-of-view. Lidar is often said to come from LIght Detection And Ranging. Lidars can be classified into 1D, 2D and 3D units: a 1D lidar provides distance along a single axis, whereas a 3D lidar creates a full depth image over its field-of-view, in the same way as a visual camera creates a full RGB image. 2D and 3D lidars often contain spinning components to minimise the number of lasers required.
LDM	Local Dynamic Map	LDM is defined by ETSI in TR 102 863 V1.1.1 (2011-06). It essentially means the same as Object Model.
	Localisation	The process of an ADS working out its exact location in the world, relative to a known map. Often GPS alone will not give the level of accuracy required by the ADS. In this report, we generally group localisation in with sensor processing.

| Acronym	Concept (alphabetical order)	Description
ML	Machine Learning	Using a large quantity of labelled data to train a software program that encodes a statistical model of some system. An example statistical model could be: do the properties of this image belong with the set of images containing cats, or the set containing dogs? Labelled data means that the training data must have a label specifying whether (for this example) the image is a cat or a dog. In machine learning, acquiring a large amount of labelled training data is a central part of the challenge.
NN	Neural Network. Also: DNN - Deep Neural Network CNN - Convolutional Neural Network MLP - Multi-Layered Perceptron DL - Deep Learning ANN - Artificial Neural Network	A neural network is a particular type of model used in machine learning, where the model is made up of many layers of units, in a very rough analogy of human neurons. Deep learning uses deep neural networks, which are fundamentally the same as "shallow" neural networks, but have more layers.
	Object Model	A map of the objects (vehicles, non-vehicle actors, roads, lane markings, traffic lights,…) in the vicinity of the AV.
	Online Learning	Continued learning for an ML system using data received during the deployment phase. This might mean a particular AV learning a better model of pedestrians, based on the pedestrians it observes during its owner's commute. As well as verification issues, this is challenging as ML usually needs ground-truth labels for the data it is presented with (e.g. it cannot learn to recognise post-boxes unless you tell it which part of the training images represent post-boxes).
ODD	Operational Design Domain	The set of conditions under which an ADS can operate safely. These conditions may include geographic limits, limits on the type of road (e.g. motorways only), weather conditions, and others. (SAE J3016 term)
	OTA Emulation (OTA Sensor Emulation)	Over-the-air emulation of sensor data: sending artificially-generated signals "over-the-air" to a sensor in order to present a view of the world. For example, placing a computer monitor in front of a camera, and displaying a view of the 3D simulated environment on the monitor, to spoof that camera's view of the world.
	OTA Updates	Over-the-air software updates, in particular updates of the ADS software.
	Planner	ADS module that decides on the (short-term) path an AV should follow, given an Object Model (and desired medium-term waypoint) as input.
	Plant Model	Same as Vehicle Dynamics Model
	Physical Test	A physical test with a complete vehicle, either on a public road or at a private test facility.
RGB	Red-Green-Blue	RGB is used to refer to digital images, which are actually stored as the amount of red, green, and blue to mix to represent the colour of each pixel in the image.
	Real World Test	Same as Physical Test. Sometimes used to mean a test on public roads, as opposed to a private test facility.

Acronym	Concept (alphabetical order)	Description
	Scenario (situation, scene, vignette)	A description of a short interaction between an AV and other road users and/or road infrastructure. A scenario may be specified at various levels of detail, for example at the highest level, "drive within a lane on a motorway, surrounded by other traffic" might represent a scenario. At the lower levels a scenario might specify the location, heading and speed of every actor at all times during the scenario, and also include the exact road geometry, layout, surrounding buildings, and so on.
	Scene Generation Tool	Same as ADAS Simulator.
	Sensor	A device that measures some quantity of the world, and converts it into a digital form. An active sensor (such as radar or lidar) transmits a signal and examines the return. A passive sensor (such as a camera, GPS or thermometer) simply captures existing signals and/or measurements.
	Sensor Fusion	Fusing data from multiple sensors to create a single coherent picture of the world. The intention is this fused picture will be more accurate than any of the sensors individually.
	Sensor Processing	The analysis of raw data from sensors to identify objects and infrastructure represented. For an ADS, the goal of sensor processing is usually to find objects, their location, their orientation, their type, and ideally their velocity.
SPU	Sensor Processing Unit	Hardware that performs sensor processing. This may be integrated into the sensor, or be a custom-dedicated ECU which is separate from the sensor itself.
	Simulation	Replicating the behaviour of a part of the real world, by using a computer model.
	Simulator	A (generally software) system that performs simulation.
SIL	Software-in-the-Loop	Generally used to mean a simulation using software only, without any HIL.
	Spoofing	Same as OTA emulation. Often used in the context of malicious manipulation, e.g. spoofing an ADS camera so that it sees a National Speed Limit sign instead of a Give Way sign.
TA	Type Approval	The mechanism which provides legal approval for the sale of a vehicle model.
TAA	Type Approval Authority	An organisation that manages the process of granting type approval to vehicle models.
	Validation	Is the spec correct, and does the completed system do what we want it to?
VCS	Vehicle Control System	All systems, hardware and software concerned with controlling the vehicle; as well as the ADS, this would include the engine ECUs, physical sensors, ABS controller, stability controller, etc.
	Vehicle Dynamics Model	A model describing how a vehicle will react to a given situation and set of control inputs. For example, how fast will the vehicle accelerate, given the application of 50% throttle, a certain initial road speed, passenger/luggage weight, and road surface and gradient?

Acronym	Concept (alphabetical order)	Description
VeHIL	Vehicle Hardware-in-the-Loop	Sometimes used to mean VIL, and sometimes used to mean HIL (where the hardware that is in-the-loop is identical to the hardware on the real vehicle).
VIL	Vehicle-in-the-Loop	A test infrastructure where the complete vehicle is tested within a simulator.
	Verification	Has the system been implemented correctly, does is perform exactly as specified? c.f. formal verification.
V&V	Verification & Validation	Often used to refer generally to the final stages of testing to assure the system works correctly. Also refers to doing both Validation and Verification.
vDVP	Virtual Design Validation Plan (Virtual Validation)	NOTE we have aimed to avoid the use of the word "virtual" in this report, as this word has proved confusing in a regulatory context.
	Virtual Testing	Used in general to mean any testing performed virtually, as opposed to in the physical world. Used by vehicle regulators to specifically refer to testing a mechanical component using a computer model of that component. Such testing will have a tightly-defined specification and pass criteria.
	White-Box Testing	Using detailed knowledge of how a system is designed and implemented to help devise and execute test cases.

8.2 TEST AND VERIFICATION LANDSCAPE

8.2.1 Current approach to automotive testing (up to SAE L2) and how products are brought to market

The prevailing method for developing ADAS features (and many engineering products) is the Systems Engineering 'V' model, as shown in Figure 12, in which the vertical position represents how high level (i.e. a broad scope but with little detailed analysis) or low level (i.e. detailed design within the narrow scope of a small portion of the overall system) the activity is, and the horizontal position represents the timing within the program (with time passing from left to right).

Systems Engineering 'V Model'

Figure 12 - Systems Engineering V Cycle
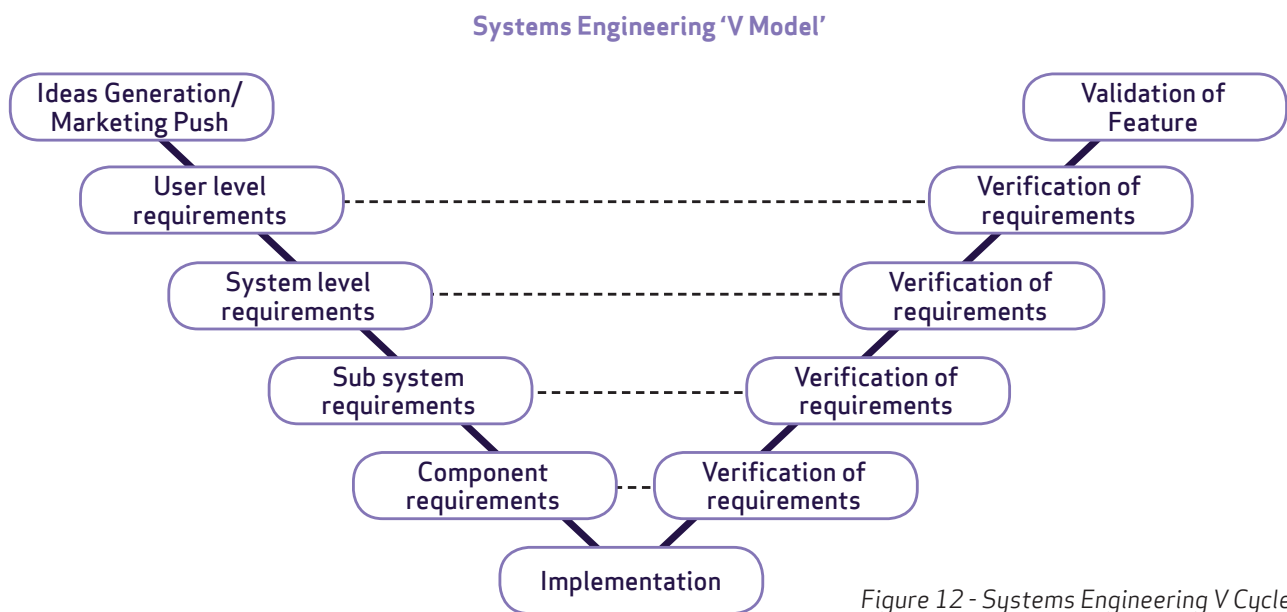
It should be noted that complete subsystems will often be delivered by a tier 1 supplier, and therefore an OEM will have limited visibility of how subsystem requirements are broken down into component level requirements, or how the component is developed (often by a tier 2 supplier contracted by the tier 1 supplier). The OEM will therefore be dependent on some form of Service Level Agreement where the supplier is obliged within the contract to meet a range of KPIs (Key Performance Indicators).

Modelling activities can be used during the system design to check that the functionality described by the requirements will result in a feature that delivers the required performance, and changes may be needed as a result. Such modelling will typically use off-the-shelf software such as **IPG Carmaker** or **TASS PreScan** to provide the virtual environment, with the feature itself being modelled via a rapid prototyping method such as **Stateflow** and Simulink (which are both incorporated into Mathworks Matlab software), or by using State Machine or Sequence Diagrams to generate code in a Systems Modelling Language (SysML) platform such as **IBM Rational Rhapsody**, this code then being used within the model. Such platforms also include means to quickly test for logical flaws within the **SysML** model.

The 'V' shouldn't be seen as strictly linear, as many iterations will be needed on both sides of the 'V' as errors are found and corrected; indeed, some companies utilise 'Agile' development methods such as Extreme Programming (XP) or Scrum, where basic versions of software are created quickly, and then the results are assessed and the whole process is iterated until suitable performance is obtained.

However, the underlying principle remains; a finite set of use cases and their corresponding requirements and test cases are considered, meaning that it is possible to precisely evaluate performance. This is feasible for driver assistance features only intended to function in a finite range of scenarios, such as active safety features that will protect road users in the most prevalent or serious use cases only, or automation features that have a very limited Operational Design Domain. However, for higher levels of automation, such prescriptive approaches become challenging due to the vastly increased range of possible scenarios that the vehicle has to be able to deal with.

8.3 REGULATORY BACKGROUND

8.3.1 Regulatory and EuroNCAP precedents for testing in simulation

Although European Union and United Nations Economic Commission for Europe (UN ECE) Type Approval Regulations are currently heavily biased towards physical testing, there are some precedents for the use of testing via virtual means, often backed up with some physical testing to provide an overcheck or validate the model, where it is uneconomic or otherwise impractical to rely solely upon physical testing:

- **European Union Type Approval Framework Directive (2007/46/EC):** Article 11 contains provisions for simulation data to be used to agree with the test authority upon a vehicle specification for physical testing that is representative of the worst case for the vehicle type (paragraph 2), and for certain tests to be conducted entirely by virtual means (paragraph 3, referencing Annex 16) – this requires details of the model used to be supplied, and requires any mathematical model used to be validated by physical tests

- Examples of this are **UN ECE Regulations 94 and 95 (Frontal and Side Impact crash testing respectively),** both of which are required for type approval under 2007/46/EC; it is commonly accepted practice for simulation data to be presented by the manufacturer to justify selection of the worst case for physical testing, as it is not feasible to crash test every possible variant and version of a vehicle type. Some simpler regulations to assess, such as UN ECE Reg 49 (Devices for Indirect Vision, i.e. mirrors) are permitted to be tested entirely via virtual means under Annex 16 of 2007/46/EC, and are commonly assessed via CAD models/ drawings.

- **UN ECE Regulation 29 (Truck Cab Strength) paragraph 5.1.6**: none of the physical tests need to be conducted if the manufacturer can show via "computer simulations or calculations of the strength of component parts or by other means to the satisfaction of the technical service" that the vehicles would pass the prescribed tests. This leaves a lot of room for interpretation of what methods are allowable!

- **UN ECE Regulation 66 (Bus and Coach Rollover) section 5.4**: At the discretion of the manufacturer, equivalent test methods can be chosen, which include "Computer simulation – via dynamic calculations – of the basic rollover test on a complete vehicle". Note that a full physical test would be particularly expensive in this case, hence the need for a pragmatic compromise. Again, there is a lot of room for interpretation of what is acceptable

- **EuroNCAP Pedestrian Protection Protocol**: The bonnet and bumper are marked with a grid and the manufacturer is required to provide information of how the vehicle will perform for each grid location – this may be derived from simulation and/ or from physical tests. EuroNCAP then choose test points based on expected worst-case positions (legform tests) or random selection (headform tests), to confirm the manufacturer data. Where there are discrepancies, a correction factor will be applied, up to a limit where further investigation will be required

- **EuroNCAP City and Inter-Urban Autonomous Emergency Braking (AEB) Protocol**: this will, from 2018, require the system to respond to a static or slower moving car target not just at a range of speeds but also a range of overlaps (i.e. differing relative offsets between the two cars), creating a two-dimensional matrix of test scenarios. Manufacturers are required to provide data (either from simulation or from physical tests) for each scenario, and then some scenarios are tested at random to confirm the accuracy of the data provided. As per EuroNCAP pedestrian protection, a correction factor is applied to account for any discrepancy

- **European Union Pedestrian Protection Regulation 78/2009/EC**: this sets out the required performance when impacted with a headform for two different zones. The Technical Service doing the tests will then select a number of points within each zone; there is no grid system, so a point can be selected anywhere, and in practice is selected at points expected to be particularly challenging. The manufacturer is not required to provide any prior evidence to show performance at any test points, but it is up to them to be confident that they can pass, knowing they could be tested in any location. This would be less suitable for EuroNCAP, as it would be unfair to publicly compare manufacturers being tested in completely different points (a function that the EuroNCAP grid system is able to provide), but is regarded as acceptable for regulations because it provides a reasonable indication that the minimum requirements have been met, and a clear incentive for OEMs to ensure they can meet the standard at all points.

8.3.2 Current Regulation for AVs

At the time of writing this report there is limited regulation in place that is specific to autonomous and highly automated vehicles, however there is a strong recognition that this is a situation that needs to move quickly to keep pace with fast developing technology.

Currently within the UK, AVs are bound by the existing legal requirements for general road users. Recognising the need to encourage the development of AVs, the Department for Transport published its Code of Practice for Testing of Automated Vehicle Technologies[29] in July 2015. The Code of Practice provides guidance to anyone wishing to conduct testing of automated vehicle technologies on public roads or in other public places in the UK. Updates to the Code of Practice are anticipated as technology advances and requirements change.

A key element of regulation for AVs is that of Type Approval. Production vehicles sold in EU member states require EC type approvals, which are issued on the basis of Directive 2007/46/EC[30]. This directive forms an umbrella to a range of ECE regulations which cover the type approval for the majority of systems found within a car. The ECE regulations are developed in line with the 1958 ECE Agreement, which aims to standardise the technical requirements for vehicles and their associated parts across member states.

Fundamental to automated driving is ECE Regulation 79 (Reg. 79) which contains the requirements for automated steering systems. Currently Reg. 79 limits the "automatically commanded steering function" to assisting the driver in following a particular path, in low speed manoeuvring or parking operations, limited to 12kmh (10kmh +20% tolerance). This regulation is currently under review with a goal of increasing the maximum speed at which AVs may be driven on public roads. Once agreed then increasing levels of autonomous and highly automated driving may be permitted on UK roads, utilising automated steering.

---

[29] https://www.gov.uk/government/publications/automated-vehicle-technologies-testing-code-of-practice
[30] http://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX%3A32007L0046

8.4 EXAMPLE OF AN ADS ARCHITECTURE

Figure 13 shows an idea of what a full ADS architecture might look like, expanding slightly on the view provided in Section 2.4.2.
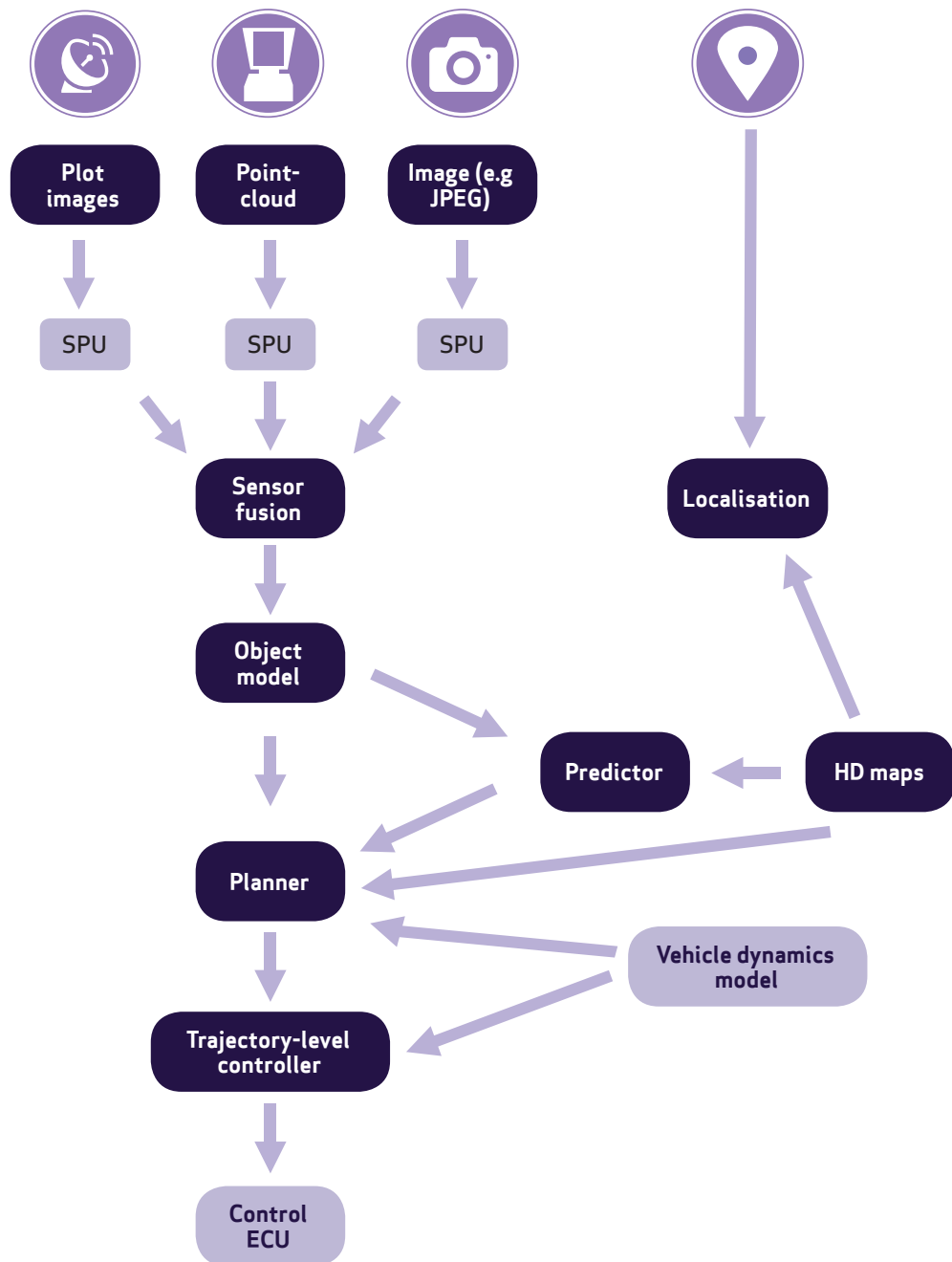

Figure 13: A more detailed view of a possible ADS architecture

In Figure 13, the sensors, SPUs (sensor processing units), sensor fusion, object model, and control ECU are the same as described in Section 2.4.2. The boxes immediately below the three example sensors are intended to provide a clearer idea of the raw data output from those sensors, and the GPS/GNSS is only used for localisation. Localisation means finding the location of the vehicle, and for an AV high accuracy is needed (of the order of centimetres), as junctions, lanes, turnings, and other map items need to be matched to sensor data. Conversely, if the AV has an HD map of the area, it may use the correspondance of sensed features with map features to help determine where it is.

The object model and road layout are used by the predictor to estimate the future path of each actor in the scene – for example, if the object model reports a car in front is indicating right, and there is a right turn 50m ahead, the predictor may estimate the car will take that turning. The predicted future paths, road layout from the 3D map, and object model are all used by the planner to determine the best trajectory for the AV to take. Finally, a model of the vehicle dynamics is used by the planner to ensure it plans paths that are feasible for the vehicle, and by the controller to calculate the commands which will produce a particular behaviour from the vehicle.

8.5 SUMMARY OF CURRENT SIMULATION TOOLS AND SIMULATOR PRODUCTS DEVELOPMENTS

This section supports Section 5.1 above and is a survey of **off-the-shelf** simulation products.

8.5.1 ADAS-focused simulators

TASS PreScan[31]– Netherlands (Siemens: Germany)

PreScan was initially developed by TNO, and then spun out into TASS. In August 2017, it was announced that Siemens had acquired TASS international.

PreScan has focused on sensor simulation and Automated Driver Asist Systems (ADAS). PreScan is a simulation platform consisting of a GUI-based pre-processor to define scenarios and a run-time environment to execute them. The engineer's primary interface for making and testing algorithms includes MATLAB and Simulink. PreScan can be used from model-based controller design (MIL) to real-time tests with software-in-the-loop (SIL) and hardware-in-the-loop (HIL) systems. PreScan can operate in open-loop & closed-loop, and offline & online mode. It is an open software platform which has flexible interfaces to link to 3rd party vehicle dynamics model and 3rd party HIL simulators/hardware.

PreScan works using four steps:

1. Build scenarios
2. Model Sensors
3. Add Control Systems
4. Run Experiments

VIRES Virtual Test Drive (VTD)[32] – Germany

Like PreScan, VTD is a toolchain focused on sensor simulation and ADAS modelling. VTD enables the creation, configuration, presentation and evaluation of virtual environments in the scope of road based simulations. It is used for the development of ADAS and automated driving systems as well as the core for training simulators. It covers the full range from the generation of 3d content to the simulation of complex traffic scenarios and, finally, to the simulation of either simplified or physically driven sensors. It is used in SiL, DiL, ViL and HiL applications and may also be operated as co-simulations including 3rd party or custom packages. By its open and modular design, it can be interfaced and integrated.

Again, as with PreScan, VDT splits its process into four steps

1. Creation of virtual worlds
2. Configuration of virtual worlds
3. Simulation of Virtual worlds
4. Customisation of virtual worlds

Mathworks[33] – USA & UK

Mathworks produce the Matlab and Simulink tools which are very widely used in the automotive industry. In particular Simulink models are used in the development of most pre-ADAS vehicle controllers, and can even be deployed directly to ECUs (following a code generation process).

Recently Matlab has extended its capabilities for ADS development with the Autonomous Driving Toolbox, available from the 2017b SW release. This toolbox provides algorithms and tools for designing and testing ADAS and autonomous driving systems. It allows engineers to automate ground-truth labelling, generate synthetic sensor data for driving scenarios, perform multi-sensor fusion, and design and simulate vision systems.

Integration to 3rd party developer tools is generally seamless with the MathWorks toolset. Mathworks has a UK-based sales and development team.

---

[31] https://www.tassinternational.com/prescan
[32] https://vires.com/vtd-vires-virtual-test-drive/
[33] https://uk.mathworks.com/

48

rFpro[34] – UK

rFpro provides driving simulation software, and 3D content, for Deep Learning Autonomous Driving, ADAS and Vehicle Dynamics testing and validation. rFpro is focused on ground based road vehicle simulation.

rFpro started 10 years ago as a motorsports simulator company concentrating on vehicle dynamics and engineering evaluation tools for F1. According to rFpro website:

"rFpro is being used for driving simulation by 6 of the top 10 OEMs as well as many smaller OEMs and T1s for Virtual Test programmes of vehicles", subsystems, ADAS and Deep Learning Autonomous Driving.

Their standard environment modelling solution represents the road surface as a 1cm grid in x,y, with sub-millimetre accuracy in z. Figure 14 shows an example.

Existing partners include:

1. Motion Platform providers
2. Vision System providers
3. Vehicle model integration HIL and SIL system providers

Figure 14: Example of photo-realistic scene from rfPro, ©rfPro

Gazebo[35] and ROS[36] – USA (open source)

Gazebo is an open-source simulator offering the ability simulate populations of robots in complex environments. Includes a physics engine, high quality graphics and graphics interface. The development of Gazebo is managed by the Open Source Robotics Foundation (OSRF), but as an open source product, Gazebo's development is supported by a diverse and active robotics community. Gazebo has been used for CAV simulation in limited contexts[37].

Gazebo is informally part of the Robot Operating System (ROS), a set of open source software libraries and tools enabling the user to build robotic application. ROS is considered as an early research and development tool kit for robotics systems, but to address this the team at the OSRF are developing ROS 2.0. This new release will aim to cope with some of the issues that ROS 1.0 has and expand its capabilities and use cases, such as real-time systems and production environments[38].

ROS and Gazebo are becoming more popular for AV development, with BMW presenting at ROSCon 2015 [39] and presentations at ROSCon 2017 focusing on vehicle and city simulation for the development of CAVs.

The TSC prototype Basic Autonomous Control Systems (B-ACS) for the LUTZ pods has been developed using ROS and simulated using Gazebo. Figure 15 shows an early version of the pod model in Gazebo.

[34] http://www.rfpro.com/
[35] http://gazebosim.org/
[36] http://www.ros.org/
[37] http://robohub.org/simulated-car-demo-using-ros-kinetic-and-gazebo-8/
[38] http://design.ros2.org/articles/why_ros2.html
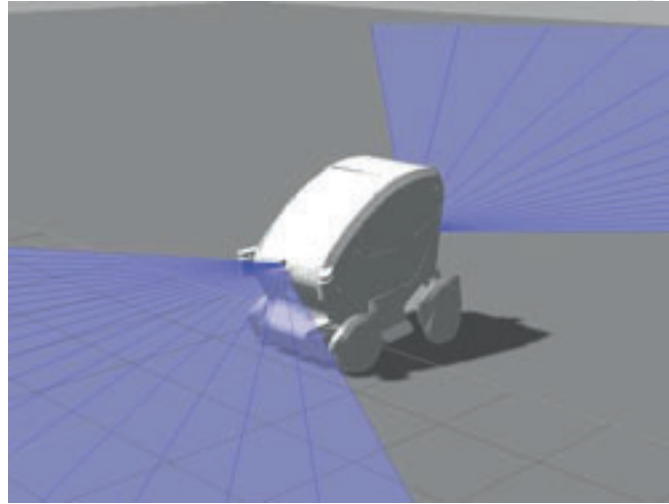[39] http://www.ros.org/news/2016/05/michael-aeberhard-bmw-automated-driving-with-ros-at-bmw.html

Figure 15: The TSC's LUTZ pod as simulated in Gazebo

CGA Simulation[40] – UK

CGA use computer game technologies to visualise and solve problems from the real world.

Relevant case studies listed on their website include:

1. Traffic Simulation with an anomaly analyser to detect suspicious behaviour,
2. Autonomous Robotic Simulator

Carla Simulator[41][42], – USA, Spain (open source)

Carla is an open-source simulator developed by Intel Labs, Toyota Research Institute and Computer Vision Centre. It was first released in November 2017, and is built on Unreal Engine (see Section 8.5.3). It aims to provide sensor, environment, and weather models suitable for ADS development, adding the functionality of other ADS simulators to a gaming engine.

Metamoto[43] – USA

Metamoto is a start-up aiming to provide ADS validation in simulation as a service. While they do not appear to have a product available yet, they are attempting to address the challenge described in Section 4.2, including randomisation of scenarios and high-fidelity simulation of multiple sensor modalities.

8.5.2 Automotive engineering-focused simulators

dSPACE[44] – Germany

dSpace is one of the leading suppliers for development and testing toolsets for the automotive industry and they offer a range of tools for automotive development lifecycle. In November 2017 dSPACE announced the introduction of a toolset for ADAS and autonomous driving system development[45].

Development of dSpace product is carried out in Germany. dSpace have an application office based in the UK.

NI (National Instruments)[46] – USA, UK

NI have a product line devoted to HIL testing, competing with dSpace. They offer a single, software-centric, platform-based approach to automotive testing that can natively integrate all the I/O types you need to test today's and tomorrow's ADAS technology[47].

---

[40] http://www.cgasimulation.com/#section-projects
[41] http://www.carla.org/
[42] http://proceedings.mlr.press/v78/dosovitskiy17a/dosovitskiy17a.pdf
[43] https://www.metamoto.com/
[44] https://www.dspace.com/en/inc/home.cfm
[45] https://www.dspace.com/en/inc/home/applicationfields/our_solutions_for_driver_assistance_systems.cfm
[46] http://www.ni.com
[47] http://www.ni.com/en-gb/innovations/automotive/advanced-driver-assistance-systems.html

IPG CarMaker[48] – Germany

A tool that provides a modelling environment that includes:

1. **Vehicles:** Passenger vehicle models that display realistic behaviour to the limits of vehicle dynamics
2. **Road:** Real roads or user-specific test tracks with a detailed environment
3. **Driver:** (IPGDriver) An adaptive driver model with artificial intelligence
4. **Traffic:** Complex traffic scenarios with countless traffic objects

While CarMaker was originally focused on vehicle dynamics modelling, it has recently included more sophisticated sensor and environment modelling, bringing it closer to tools like PreScan and VIRES VTD. IPG has recently announced a collaboration with rFPro[49].

CarSim[50] – USA

CarSim is a commercial vehicle dynamics software toolset. It provides methods for simulating the performance of passenger vehicles and light-duty trucks. With circa twenty years of real-world validation by automotive engineers, CarSim claims to be used by 7 of the 10 largest automotive OEMs.

CarSim works as a standalone vehicle dynamics application and has a standard interface to MatLab/Simulink allowing users to build complex scenarios and test event sequences. Supports: SiL, MiL, HiL and DiL, vehicle sensors integration and interactive traffic for Vehicle-2-Vehicles and ADAS development.

AVL[51] – Austria

Powertrain expertise, but also developing solutions for Automotive ADAS and ADS systems, including the AVL In Motion product. This software development is in collaboration with CarMaker. AVL InMotion is integrated with CarMaker to deliver scalable real-life simulation solutions for different test environments. AVL InMotion makes manoeuvre-based testing integral.

Claytex[52] – UK

Claytex is a UK based consultancy, developing and providing systems engineering tools. Claytex provides a comprehensive suite of solutions for automotive and motorsport, enabling all aspects of a vehicle to be modelled, simulated and tested in a virtual environment. These solutions include support for testing ADAS and autonomous vehicles using immersive virtual test environments allowing vehicle model, sensors and control systems to be fully immersed into a virtual world.

8.5.3 Graphical / gaming engines

Unity[53] – USA

Unity is a widely-used game development platform. It allows 3D environments to be defined using an external editor (such as 3DS-Max), and provides high-quality camera views of this environment. It also allows objects to be controlled by a physics engine, and is fully configurable and controllable using scripting.

Therefore, while it isn't intended to work as an ADAS simulator, it is a useful starting point for one.

Unreal Engine[54] – USA

Unreal Engine is, like Unity, a popular general-purpose games development engine. It provides a scripting engine, physics engine, and highly realistic video capabilities. As Unity, it therefore provides many of the components needed in an ADAS simulator.

[48] https://ipg-automotive.com/products-services/simulation-software/carmaker/
[49] https://ipg-automotive.com/news/article/testing-automated-driving-functions-with-an-unprecedented-level-of-detail/
[50] https://www.carsim.com/products/carsim/index.php
[51] https://www.avl.com/-/avl-inmotion-real-life-simulation
[52] http://www.claytex.com/
[53] https://unity3d.com/
[54] https://www.unrealengine.com/

Grand Theft Auto (GTA) – USA

GTA V is produced by Rockstar Games. The game features an open world design, containing more than a thousand roaming characters, 262 different vehicles, multiple different weather and lighting conditions and complex infrastructure, such as bridges and tunnels.

In 2016 Researchers at Intel labs and Darmstadt University developed a method to extract training data from GTA 5, using software to classify different objects in the road scenes and feed this information back to a machine learning algorithm. The learning algorithm uses this information to recognise cars, pedestrians and other objects.

News article links below:

https://www.bloomberg.com/news/articles/2017-04-17/don-t-worry-driverless-cars-are-learning-from-grand-theft-auto
https://eandt.theiet.org/content/articles/2017/04/driverless-cars-learn-road-safety-from-grand-theft-auto-5/

CVEDIA[55]: SynCity[56] – Holland

CVEDIA is a free cloud-based service that simplifies image dataset preparation and management. SynCity is a product of CVEDIA.

Limited information is available for SynCity and CVEDIA. CVEDIA's CEO is based in the Netherlands.

Improbable[57] – UK

HQ in London, recently secured £390m investment. Building software enabling efficient large scale simulation and modelling for science, research, defence applications.

Immense Simulation[58] – UK

Founded in 2016 as a spin out from the Transport Systems Catapult, they are an applied technology start-up aiming to meet the challenges of the global transportation market. Their prediction engine is powering Intelligent Mobility, MaaS and the integration of Connected Autonomous Vehicles to enable partners to create tomorrow's sustainable, efficient, customer centric transport solutions.

8.5.4 Traffic Modelling Software

PTV Vissim[59] – Germany

Porsche Automobil Holding SE acquired PTV in September 2017.

PTV Vissim is a microscopic multi-modal traffic flow simulation software package allowing the simulate of traffic patterns. Including; transportation planning, signal timing, public transport and urban planning, for motorised private transport, goods transport, rail and road related public transport, pedestrians and cyclists to a 3D visualisation.

Vissim displays all road users and their interactions in one model providing realistic modelling of all road users.

TSS: Aimsun Next[60] – Spain

Aimsun Next is a traffic modeling software that allows the user to model anything from a single bus lane to an entire region, enabling high speed simulation and fusing travel demand models, for static and dynamic traffic assignment, with mesoscopic, microscopic and hybrid simulation. Aimsun features a V2X specific API that models vehicle to vehicle and vehicle to roadside systems communication, with the implementation of associated protocols (CAM, DENM, SPATEM, MAPEM messages) and the capability of introducing perturbations like latency and packet loss; this API is being used within the Flourish CAV1 project to implement rules engines at vehicle, roadside unit and traffic control centre levels. For ADAS simulation Aimsun can interface with SCANeR, developed by Oktal.

---

[55] https://cvedia.com/
[56] https://syncity.com/
[57] https://improbable.io/
[58] https://imsim.co.uk/
[59] http://vision-traffic.ptvgroup.com/en-us/products/ptv-vissim/
[60] https://www.aimsun.com/

DLR SUMO[61] (Simulation of Urban Mobility)

Simulation of Urban MObility (SUMO) is an open source, road traffic simulation package designed to handle large road networks. SUMO is developed by the Institute of Transportation Systems, Berlin.

The complete suite includes tools for importing road networks, generating routes from different sources, and traffic simulation, allowing modelling of intermodal traffic systems including road vehicles, public transport and pedestrians.

8.5.5 Driving simulators

Ansible Motion[62] – UK

Ansible Motion provides fully immersive Driver-in-the-Loop simulator solutions for vehicle constructors and suppliers around the world. Including: professional motorsport (supplier to F1, NASCAR, Indycar and Le Mans organisations), specialty transportation constructors (virtual driving experience for specialty vehicles). Systems include motion and visual immersive cueing for human drivers, vehicle physics models, virtual environments and virtual proving grounds.

XPI Simulation[63] – UK

XPI Simulation design and manufacture a range of driving simulators, from desktop solutions to full-scale converted vehicles for commercial, military and institutional clients.

In conjunction with its range of standard and bespoke driving simulator platforms, XPI provides simulator-based driver training services through the Driving Simulation Centre (DSC).

XPI Simulation created the 3xD simulator at Warwick University's WMG centre.

XPI is a wholly-owned subsidiary of Thales UK.

Figure 16 - Example of virtual scene from XPI Simulation (image courtesy of XPI simulation)

OKTAL[64] – France

Oktal software runs the Venturer simulator at the BRL. Their focus is on simulators for human factors experiments, and driver training, and their products include a full-motion platform simulator. The simulator includes 3D environment modelling, scripts to control all actors, vehicle dynamics, the ability to interface to Matlab/Simulink, and they also supply the ADAS development market.

[61] http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/
[62] https://www.ansiblemotion.com/
[63] http://www.xpisimulation.com/
[64] http://www.oktal.fr/en/automotive/range-of-simulators/range-of-simulators

Cruden[65] – Netherlands

Cruden is a designer, manufacturer and integrator of professional open architecture driving simulators for the automotive, motorsport, marine and motorcycle industries.

Used by JLR integrating IPG Carmaker and desktop simulation via the Cruden Panthera ePhyse Net external physics interfacing package

DALLARA[66] – Italy

Provides driver simulator for all Dallara cars. Specialist in motorsport, from Formula 3 to IndyCar.

Moog[67] – USA

Moog provide driving simulators for automotive OEM's and Motorsport. Their simulator products span Space, Aerospace and Marine sectors.

[65] http://www.cruden.com/#innerShape481_Shape481
[66] https://www.dallara.it/wps/portal/en/expertise/Vehicle-Dynamics/Driving-Simulator#.Wgx9f8dLHlU
[67] http://www.moog.com/markets/automotive-test-simulation/automotive-performance-testing/driving-simulator.html