EDITOR **PHIL LAPLANTE**
The Pennsylvania State University;
plaplante@psu.edu

UL 4600: What to Include in an Autonomous Vehicle Safety Case

Philip Koopman, Carnegie Mellon University

UL 4600 provides an umbrella for coordinating software development practices and computer-based system safety standards to make sure nothing is left out when assuring safety.

Autonomous vehicles (AVs) will not see widespread use until we can be sure that they are acceptably safe. That remains a big challenge, but robust support from safety standards can help. To illustrate what is involved in ensuring AV safety, this article provides an overview of the approach taken by the ANSI/UL 4600 AV safety standard.[1]

Digital Object Identifier 10.1109/MC.2023.3236171
Date of current version: 3 May 2023

MITIGATING UNUSUAL HAZARDS

While AV safety involves software, hardware, sensor technology, and more, it is software safety that is by far the biggest challenge. Creating life-critical software goes well beyond writing high-quality code. Even perfect software implementations might have safety issues due to requirements defects and encounters with novel operational environments.

Adding the complexity of machine learning-based technology brings into play issues such as avoiding training data bias and ensuring a sufficiency of data samples for rare events. While progress on AV functionality to date has been impressive, it might be said that deploying the first hundred vehicles is the easy part. Ensuring the lifecycle safety of a truly driverless AV, deployed at scale across a wide variety of operational conditions, will require significant engineering effort that has only just begun.

Conventional vehicles place a significant safety burden on the driver for handling unusual circumstances

and equipment failures. Being able to drive down a road in normal conditions without collisions is only the start of safety. Gracefully dealing with unusual situations and inevitable equipment failures will also be required of automated driving systems once the human driver is out of the loop.

A common enough attribution for a crash is that the driver failed to respect road infrastructure limitations or was unable to compensate for a vehicle equipment failure. For example, drivers are expected to be able to stay on the road even if lane marking lines have been obscured, avoid driving into flooded roadways, and brake effectively even if the vehicle's antilock braking system is inoperable.

Gracefully dealing with unusual situations and inevitable equipment failures will also be required of automated driving systems once the human driver is out of the loop.

Exceptional operational conditions might be rare when considering an individual vehicle and driver. But such situations happen often enough across all deployed vehicles to be a concern, given the impressive human driver capability of almost 100 million mi between fatality crashes.[2] While human drivers might further improve by avoiding drunk driving, and infrastructure changes could make a big difference, at these comparatively low rates, any fatal crash is unusual. We can similarly expect that crashes for mature AV technology will involve unusual circumstances, as well.

While it is often said that AVs will not drive drunk, they will fail in other different ways than human drivers. Those different ways will tend to involve rare circumstances that are poorly handled by potentially brittle automated driving systems.

Perhaps the most significant challenge for AV safety is dealing with long-tail infrequent events that nonetheless pose unacceptable risk. Humans are remarkably effective, albeit imperfect, at dealing with novel unstructured situations. The machine learning-based approaches used for AVs can achieve impressive results when dealing with commonly seen inputs. But machine learning is at its worst when dealing with novel low-probability events. Because of the extremely low fatality rate that must be met by AVs, low-probability events—many of which will never be seen in public road testing at all—will form the practical limit on real-world safety.

UL 4600 approaches the problem of safety validation of ultradependable systems by having developers go beyond the extensive simulation and testing needed to create reasonable driving behavior. While such validation approaches are essential, it is impracticable to scale them up to the billions of miles of real-world road testing that would be required for life-critical system assurance. (Even if you do a billion miles of simulation, how do you know the simulation software and models are essentially perfect?) More is required in the form of careful attention to safety engineering and lifecycle management.

STANDARDS-BASED AV ENGINEERING

A key challenge in creating an AV safety standard is avoiding a premature mandate of specific technology approaches. UL 4600 does this by not standardizing how the AV itself is built nor even the engineering methodology used. Rather, UL 4600 standardizes a way to ensure that an explanation of why safety is acceptable for a particular AV is comprehensive, consistent, and compelling.

Existing standards provide a substantial starting point for AV safety. ISO 26262 is a functional safety standard for conventional automotive electronic features.[3] It provides a framework for identifying and mitigating hazards, which is a core activity of safety engineering. ISO 21448 adds to this by providing a way to mitigate rare adverse events as they are encountered in testing, using an iterated improvement approach.[4]

While the ISO 26262 and ISO 21448 pair of safety standards can and should be used for any AV design effort, the standards have their limitations. One limitation is that they provide a hazard mitigation process but leave it up to developer experience to identify which hazards to mitigate. Another is that many aspects of system-level safety beyond automated driving are out of scope. For example, neither standard will guide a design team to identify safety issues related to poorly secured cargo and the mitigation of postcrash hazards to emergency responders.

Issued in April 2020, UL 4600 fills these gaps by providing a comprehensive umbrella standard for AV system-level safety. That includes extensive lists of potentially relevant hazards (informally, #DidYouThinkofThat? lists). Examples include clothing color affecting object classification accuracy (yellow construction vests can confuse perception systems), situations requiring judgment calls (should a 10-year-old passenger be able to override a robotaxi destination midtrip?), and how to deal with impaired passengers (what should a robotaxi do about a passed-out passenger in the back seat?).

The standard does not require specific solutions but, rather, helps ensure that the right questions have been asked during design to reduce problematic surprises during

operation. In the longer term, this structure of identifying hazard topics can form a basis for information sharing across companies. The industry should avoid having users of each different AV design suffer harm from a novel (to that company) hazard that has already been discovered by another design team.

THE SAFETY CASE

The centerpiece of UL 4600 is an instrumented safety case. A safety case is a reasoned argument, supported by evidence, as to why an AV is acceptably safe for deployment. This includes listing questions the safety case must answer (for example, is there evidence that all identified hazards have been mitigated?) as well as what threats to the validity to the safety case's argument must be considered (for example, did the design team leave some hazards off the list that should have been considered?).

Safety performance indicators (SPIs) provide instrumentation to detect whether any claim within the safety case is falsified during design, simulation, testing, and deployment. As an example, vehicle-level safety might be based, in part, on an argument that phantom (false alarm) panic braking will happen at some low but tolerable rate. But what if during deployment, metrics show phantom braking is happening twice as often as it should? The SPI approach requires both collecting such data and reconsidering the safety case in light of metric anomalies.

Pervasive use of SPIs can enable improvements before significant accumulation of harm. A key concept here is not just defining metrics but also pairing them with claims in the safety case so that metrics have a concrete relationship to safety.

An important objective of UL 4600 is to play well with other safety standards. UL 4600 is designed to be compatible with ISO 26262 and ISO 21448, involving minimal redundant effort. For projects in the government systems world, it can also be used with Military Standard 882.[5]

UL 4600 CONTENT OVERVIEW

UL 4600 requires addressing the following high-level topics in the context of the overall safety case. This list illustrates how far ensuring safety goes beyond just doing simulation and road testing:

- *Argument sufficiency and validity*: Are the assumptions made in the safety case reasonable? Are all claims supported by evidence? Is there a strong safety culture ensuring that reality matches the safety paperwork?
- *Hazard identification and risk mitigation*: Have all relevant hazards been identified according to a reasonable fault model? Have risks been mitigated sufficiently to achieve acceptable safety?
- *Interaction with people and road users*: Have people across the full range of population demographics been considered? Have all types of road users been considered, including those with unusual characteristics and behaviors? What about justifiable rule breaking by the AV?
- *Safety of autonomous features*: Each stage of an autonomy pipeline brings its own safety considerations, including sensing, perception, use of machine learning technology, planning, prediction, motion control, and computational resource management. Effective architectural redundancy approaches and a robust operational environment definition are also essential.
- *Software and system engineering processes*: These processes must be both defined and followed to ensure the sufficient quality of not only the software but also engineering analysis and other work products.
- *Dependability and redundancy management*: AVs must remain operational even after a failure in the driving computer, because there might be no human driver to take over. Doing this successfully requires close attention to redundancy management and degraded operational modes. Hazards due to malicious faults must also be considered (cybersecurity).
- *Data and networking*: Data transmission and storage must provide end-to-end integrity encompassing the collection of training data, design process data, and operational data. Road infrastructure integrity (both digital and physical) must also be considered.
- *Verification, validation, and test*: Various types of testing and runtime monitoring will each make a different contribution to understanding safety, but the limitations of those contributions must be accounted for. Corrective actions must be triggered in response to each test failure and other "surprise."
- *Tool and legacy code qualification*: Would you trust your life to a free computer vision library downloaded from the web? Was the software inside the lidar from that hot new startup company developed to life-critical safety standards? Does the

> UL 4600 standardizes a way to ensure that an explanation of why safety is acceptable for a particular AV is comprehensive, consistent, and compelling.

optimizer in the compiler used to build the simulator have code generator defects? To the degree that the safety argument is based on using simulation to displace road testing, that tends to make the simulation models, tool quality, and externally developed component software engineering practices much more critical, as well.

- *Lifecycle concerns*: The pilot fleet is just the start. Safety issues can arise from release to manufacturing, supply chain failures, operational issues, and even retirement/disposal issues.
- *Maintenance*: Ensuring that maintenance is performed as required will be essential for AV safety. Especially at first, acceptable practices might look like a required schedule of aircraft maintenance performed only by qualified personnel.
- *SPIs*: Key claims in the safety case should be monitored to see if they remain true. At some point, every system design will experience this situation: "The safety case says this can never happen … and yet, it just happened." Responding quickly to SPI violations is an opportunity to improve before a collision makes headlines and forces big recalls.
- *Assessment of conformance to the standard*: The team's safety engineers create and self-assess their safety case. Then, an independent assessor checks that both the form and substance of the safety case look good. Independent assessors are not required to be external but must have an independent arms-length relationship with the design team.

DID YOU THINK OF THAT?

Perhaps the most distinctive aspect of UL 4600 is its innovative prompting structure. Rather than an endless list of "shall" statements, there is a moderate number of such requirements, with each requirement getting its own subsection in the standard. Each of those subsections has a set of bulletized prompt elements.

The purpose of prompt elements is not to spell out, in unambiguous detail, exactly what should be in the safety case. Rather, the idea is to prompt a reasonable engineer to consider factors that are in scope for the requirement and that might otherwise have been missed. In other words, this is not a conclusive list of all possibilities but, rather, an approach of "be sure to consider this class of possible hazards," often accompanied by representative examples.

For instance, there is no list of all possible vehicle types that must be considered when ensuring that all types of road users have been accounted for in planning and prediction algorithms. Rather, there are prompts to ensure that safety engineers have considered diverse potentially relevant types, such as micromobility users, horse-drawn vehicles, farm equipment, and aircraft operating on the roadway.

In another departure from typical standards practices, the publisher of UL 4600 has made the full text of the currently active standard publicly viewable in its entirety at no charge.[6] A book by this author is also available that gives a chapter-by-chapter tour of the standard in less stylized form than the prompt element format of the standard itself.[7]

THE FUTURE

The evolution of UL 4600 continues. An update to more specifically address autonomous heavy trucks is being prepared for release in 2023. A similar approach is being considered for autonomous aircraft, and it is possible the future will see extensions that encompass off-road vehicles, such as mining and agricultural equipment.

Removing the human driver is a game-changing capability for AVs of all types. As much as that might disrupt transportation models, it also will disrupt safety engineering approaches. UL 4600 is there to help ensure that potential assurance gaps get filled when making AVs acceptably safe for large-scale deployment. C

REFERENCES

1. *Standard for Evaluation of Autonomous Products, UL Standard* ANSI/UL-4600, Mar. 2022.
2. "Early estimates of motor vehicle traffic fatalities and fatality rate by sub-categories in 2021," Nat. Highway Traffic Saf. Admin., U.S. Dept. Transp., Washington, DC, USA, DOT HS 813 298, May 2022.
3. *Road Vehicles – Functional Safety,* ISO 26262, 2018.
4. *Road Vehicles – Safety of the Intended Functionality,*" ISO 21448, 2022.
5. F. Fratrik and K. Nocera, "Leveraging ANSI/UL 4600 to ensure adequate MIL-STD 882 safety," Int. Syst. Saf. Soc., St. Paul, MN, USA, Aug. 17–19, 2021. [Online]. Available: https://system-safety.org/store/viewproduct.aspx?id=18973194
6. *Evaluation of Autonomous Products*, UL Standard 4600, 2022. Accessed: Jan. 3, 2023. [Online]. Available: https://www.shopulstandards.com/ProductDetail.aspx?productid=UL4600
7. P. Koopman, *The UL 4600 Guidebook*, 2022. [Online]. Available: http://amazon.com/UL-4600-Guidebook-Include-Autonomous/dp/B0BNKXF3Z7

PHILIP KOOPMAN splits his time between teaching safety-critical embedded systems at Carnegie Mellon University, Pittsburgh, PA 15213 USA, and helping companies around the world improve the quality of their embedded system software. Contact him at koopman@cmu.edu.