Challenges in Autonomous Vehicle Validation

Keynote Presentation Abstract

Philip Koopman
Carnegie Mellon University & Edge Case Research LLC
ECE Dept. HH A-308, 5000 Forbes Ave., Pittsburgh, PA, USA
koopman@cmu.edu

ABSTRACT

Developers of autonomous systems face distinct challenges in conforming to established methods of validating safety. It is well known that testing alone is insufficient to assure safety, because testing long enough to establish ultra-dependability is generally impractical. That's why software safety standards emphasize high quality development processes. Testing then validates process execution rather than directly validating dependability.

Two significant challenges arise in applying traditional safety processes to autonomous vehicles. First, simply gathering a complete set of system requirements is difficult because of the sheer number of combinations of possible scenarios and faults. Second, autonomy systems commonly use machine learning (ML) in a way that makes the requirements and design of the system opaque. After training, usually we know what an ML component will do for an input it has seen, but generally not what it will do for at least some other inputs until we try them. Both of these issues make it difficult to trace requirements and designs to testing as is required for executing a safety validation process. In other words, we're building systems that can't be validated due to incomplete or even unknown requirements and designs.

Adaptation makes the problem even worse by making the system that must be validated a moving target. In the general case, it is impractical to validate all the possible adaptation states of an autonomy system using traditional safety design processes.

An approach that can help with the requirements, design, and adaptation problems is basing a safety argument not on correctness of the autonomy functionality itself, but rather on conformance to a set of ***safety envelopes***. Each safety envelope describes a boundary within the operational state space of the autonomy system.

A system operating within a "safe" envelope knows that it's safe and can operate with full autonomy. A system operating within an "unsafe" envelope knows that it's unsafe, and must invoke a failsafe action. Multiple partial specifications can be used as an envelope set, with the intersection of safe envelopes permitting full autonomy, and the union of unsafe envelopes provoking validated, and potentially complex, failsafe responses.

Envelope mechanisms can be implemented using traditional software engineering techniques, reducing the problems with requirements, design, and adaptation that would otherwise impede safety validation. Rather than attempting to prove that autonomy will always work correctly (which is still a valuable goal to improve availability), the envelope approach measures the behavior of one or more autonomous components to determine if the result is safe. While this is not necessarily an easy thing to do, there is reason to believe that checking autonomy behaviors for safety is easier than implementing perfect, optimized autonomy actions. This envelope approach might be used to detect faults during development and to trigger failsafes in fleet vehicles.

Inevitably there will be tension between simplicity of the envelope definitions and permissiveness, with more permissive envelope definitions likely being more complex. Operating in the gap areas between "safe" and "unsafe" requires human supervision, because the autonomy system can't be sure it is safe.

One way to look at the progression from partial to full autonomy is that, over time, systems can increase permissiveness by defining and growing "safe" envelopes, shrinking "unsafe" envelopes, and eliminating any gap areas.

CCS CONCEPTS

• **Computer systems organization → Embedded and cyber-physical systems**; *Robotics*; Robotic autonomy • **Software and its engineering →** Software organization and properties; *Extra-functional properties;* Software Safety

KEYWORDS

Autonomous vehicles; self-driving vehicles; validation; testing; adaptation; machine learning; safety envelope; software safety

ACM Reference format:

P. Koopman, 2017. Challenges in Autonomous Vehicle Validation. In *Proceedings of 1st International Workshop on Safe Control of Connected and Autonomous Vehicles, Pittsburgh, Pennsylvania, USA, April 2017 (SCAV 2017)*, 1 page.
DOI: 10.1145/123 4

SCAV'17, April 21-21 2017, Pittsburgh, PA, USA
ACM 978-1-4503-4976-5/17/04.
http://dx.doi.org/10.1145/3055378.3055379