IT’S TIME TO DRAFT SOME GUIDELINES FOR BUILDING SAFER AUTONOMY

MICHAEL WAGNER
Co-founder and CEO, Edge Case Research
mwagner@ecr.guru
We claim that we’ve identified all the risks,
Our safety specification mitigates the risks we’ve identified,
The system satisfies our safety specification, and ...

... that we built the system according to the best-practices of our industry.
Arguing that autonomy will be safe in all unspecified environments demands unbounded validation!

SO WHAT’S DIFFICULT ABOUT AUTONOMY?
Arguing that autonomy will be safe in all unspecified environments demands unbounded validation!

So we define operational design domains that constrain the situations in which autonomy must behave safely.
SO WHAT’S DIFFICULT ABOUT AUTONOMY?

Arguing that autonomy will be safe in all unspecified environments demands unbounded validation!

So we define operational design domains that constrain the situations in which autonomy must behave safely.

Unfortunately, domains that seem equivalent to humans may not be equivalent to autonomy.

Exploring the response of a DNN to environmental perturbations from "Robustness Testing for Perception Systems", RIOT Project, Carnegie Mellon University. Figures are DISTRIBUTION A.
THE IMPACT OF SMALL CHANGES CAN BE SURPRISING
THE IMPACT OF SMALL CHANGES CAN BE SURPRISING

(Szegedy et al., 2013)
AND WHAT ABOUT REQUIREMENTS ON DETECTING OBJECTS?

“Stop when there is a pedestrian in a crosswalk.”

Photo: driversed.com
We rarely have explicit definitions of the objects that the AV needs to respond to. We turn to machine learning, which uses no requirements. But we no longer have traceability.
A simple, verified Safety Controller can help mitigate risks in an unverified Complex Controller. DARPA HACMS program used it to address security risks.

The Decision Logic module is responsible for swapping in the Safety Controller before an accident is unavoidable.

Simplex approach is useful for supporting phased-deployment safety concepts.
Our team at Carnegie Mellon designed and submitted a patent for a two-channel architecture shown here, that segments autonomy architectures into checkable pipeline stages, and defines safing behaviors that are triggered when a primary-channel safety gate fails silently. (Figure is Distribution A, NREC Case #: 2016-01-19.)
SOME USEFUL IDEAS: EVOLVING SAFETY CONCEPTS

Model the operational domains and the function architecture of the AV to generate a fault tree.

Use the fault tree to identify risks and opportunities for mitigations.

Iterate safety concept to deploy more sophisticated technology in ever more complex operational domains (Schladover et al., 2001), (Bayouth and Koopman, 1998).
BUT WHAT ABOUT PERCEPTION?
“The associations learned by supervised learning algorithms are not guaranteed to reflect causal relationships.”

Verification is different than explainability. It is the independent determination as to whether the necessary casual relationships are in place.
"Shift from the simple, binary view of standards to a staged, dynamic evaluation resembling the approval process for drugs" (which also happen to be inscrutable black boxes). Design controlled experiments on the deep learner to:

• Generate data about the actual safe operational domain, and
• Find suitable spaces in which we can compare the similarity between given operating environments.

Successful “early trials” permit ever more expanded operational domains.

This isn’t just testing to rack up miles, these would be scientific and reviewable experimental programs.
NEXT STEP: LET’S ESTABLISH BEST PRACTICES
Code with confidence. Develop your team. Lower your risk.

EDGE CASE RESEARCH

MAKE ROBUST SOFTWARE
www.ecr.guru