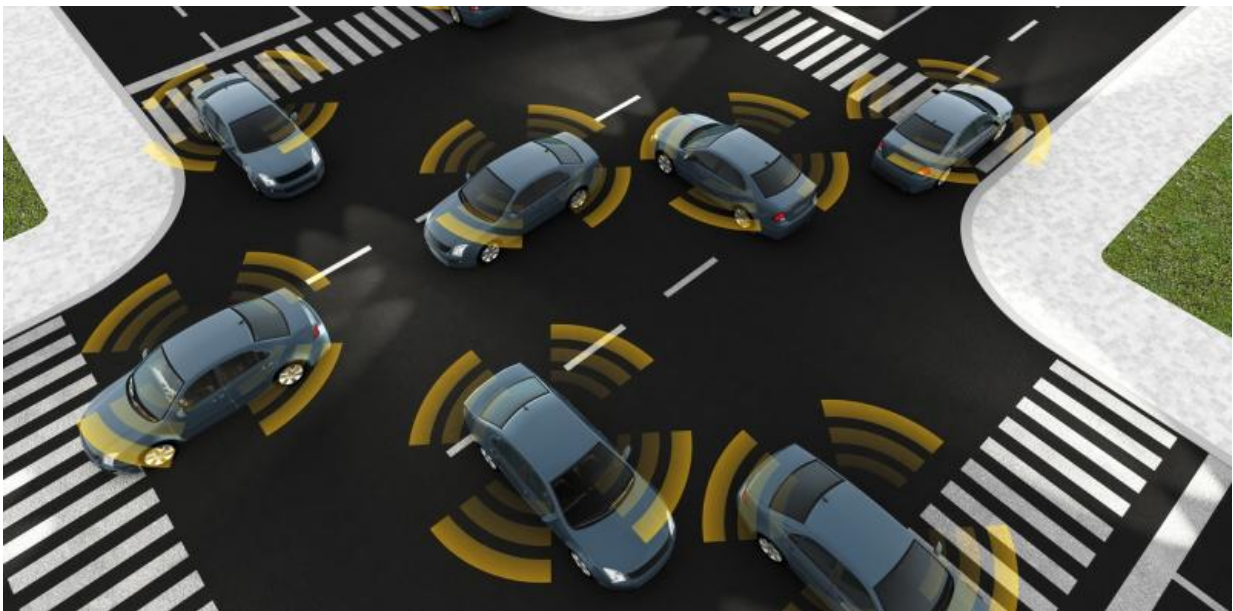


Driverless cars should learn lessons from crashes

April 13 2017, by Sean Welsh



Driverless cars still need to ‘learn’ how to drive on our roads, especially at busy junctions. Credit: Shutterstock/Karsten Neglia

Sean Welsh, *University of Canterbury*

Learning how to drive is an ongoing process for we humans as we adapt to new situations, new road rules and new technology, and learn the lessons from when things go wrong.

But how does a driverless car learn how to drive, especially when something goes wrong?

That's the question being asked of Uber after [last month's crash](#) in Arizona. Two of its engineers were inside when one of its autonomous vehicles spun 180 degrees and flipped onto its side.

Uber pulled its test fleet off the road pending police enquiries, and a few days later the vehicles were [back on the road](#).

Smack, spin, flip

The Tempe Police Department's report on the investigation into the crash, [obtained by the EE Times](#), details what happened.

The report says that the Uber Volvo (red in the graphic below) was moving south at 38mph (61kmh) in a 40mph (64kmh) zone when it collided with the Honda (blue in the graphic) turning west into a side street (point 1).



Uber crash - initial collisions. Credit: Alex Hanlon / Sean Welsh based on Tempe Police report

Knocked off course, the Uber Volvo hit the traffic light at the corner (point 2) and then spun and flipped, damaging two other vehicles (points 3 and 4) before sliding to a stop on its side (point 5).



Uber crash - subsequent collisions. Credit: Alex Hanlon / Sean Welsh based on Tempe Police report

Thankfully, no one was hurt. The [police determined](#) that the Honda driver "failed to yield" (give way) and [issued a ticket](#). The Uber car was not at fault.

Questions, questions

But [Mike Demler](#), an analyst with the [Linley Group](#) technology consultancy, [told the EE Times](#) that the Uber car could have done better:

It is totally careless and stupid to proceed at 38mph through a blind intersection.

Demler [said](#) that Uber needs to explain why its [vehicle](#) proceeded through the intersection at just under the speed limit when it could "see" that traffic had come to a stop in the middle and leftmost lanes.

The EE Times report said that Uber had "fallen silent" on the incident. But as Uber uses "deep learning" to control its autonomous cars, it's not clear that Uber could answer Demler's query even if it wanted to.

In deep learning, the actual code that would make the decision not to slow down would be a complex state in a neural network, not a line of code prescribing a simple rule like "if vision is obstructed at intersection, slow down".

Debugging deep learning

The case raises a deep technical issue. How do you debug an autonomous vehicle control system that is based on deep learning? How do you reduce the risk of [autonomous cars](#) getting smashed and flipped when humans driving alongside them make bad judgements?

Demler's point is that the Uber car had not "learned" to slow down as a prudent precautionary measure at an intersection with obstructed lines of sight. Most human drivers would naturally beware and slow down when approaching an intersection with obstructed vision due to stationary cars.

When it comes to [deep reinforcement learning](#), this relies on "value functions" to evaluate states that result from the application of policies.

A value function is a number that evaluates a state. In chess, a strong opening move by white such as pawn e7 to e5 attracts a high value. A weak opening such as pawn a2 to a3 attracts a low one.

The value function can be like "ouch" for computers. Reinforcement learning gets its name from positive and negative reinforcement in psychology.

Until the Uber vehicle hits something and the value function of the deep learning records the digital equivalent of "following that policy led to a bad state - on side, smashed up and facing wrong way - ouch!" the Uber control system might not quantify the risk appropriately.

Having now hit something it will, hopefully, have learned its lesson at the school of hard knocks. In future, Uber cars should do better at similar intersections with similar traffic conditions.

Debugging formal logic

An alternative to [deep learning](#) is autonomous vehicles using explicitly stated rules expressed in formal logic.

This is being developed by [nuTonomy](#), which is running an [autonomous taxi pilot](#) in cooperation with authorities in Singapore.

NuTonomy's approach to controlling [autonomous vehicles](#) is based on a rules hierarchy. Top priority goes to rules such as "don't hit pedestrians", followed by "don't hit other vehicles" and "don't hit objects".

Rules such as "maintain speed when safe" and "don't cross the

centreline" get a lower priority, while rules such as "give a comfortable ride" are the first to be broken when an emergency arises.

While NuTonomy does use machine learning for many things, it does not use it for normative control: deciding what a car ought to do.

In October last year, a NuTonomy [test vehicle accident was involved in an accident](#): a low-speed tap resulting in a dent, not a spin and flip.

The company's chief operating officer Doug Parker [told IEEE Spectrum](#) :

What you want is to be able to go back and say, "Did our car do the right thing in that situation, and if it didn't, why didn't it make the right decision?" With formal logic, it's very easy.

Key advantages of formal logic are provable correctness and relative ease of debugging. Debugging machine learning is trickier. On the other hand, with machine learning, you do not need to code complex hierarchies of rules.

Time will tell which is the better approach to driving lessons for driverless cars. For now, both systems still have much to learn.

This article was originally published on [The Conversation](#). Read the [original article](#).

Provided by The Conversation

Citation: Driverless cars should learn lessons from crashes (2017, April 13) retrieved 23 September 2023 from <https://techxplore.com/news/2017-04-driverless-cars-lessons.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.