Reports of Vanderbilt Student Summer Internships 2018

Edited by Janos Sztipanovits and David J. Hess

Partnerships for International Science and Engineering (PIRE) Program

National Science Foundation

OISE-1743772, “Science of Design for Societal-Scale Cyberphysical Systems”

Table of Contents

Tony Lin

Human Interaction with the Autonomous Vehicle

Mentor: David Käthner, Deutsches Zentrum für Luft- und Raumfahrt

1. General Problem and Context

Today there is a disconnect between connected autonomous vehicles (CAVs) and humans. For most people, the technology is a black box, and lack of understanding naturally leads to distrust. This is highly problematic because if the general population does not trust this technology, lawmakers would be less inclined to legalize CAVs for the road. This would mean reduced profits, and companies could lose the incentive to continue working in this area. Thus, the problem is creating trust such that both the general population and policymakers will understand enough to feel comfortable accepting connected autonomous vehicles. This would mean designing a system that is both transparent and secure in order to adapt to varying levels of scrutiny.

Some of the factors creators of connected autonomous vehicles need to account for include ethics, accounting for what the car would do when about to enter into an accident, privacy, masking data or collecting only certain data, transparency, letting users know what the car is doing, and giving users full control when requested. This report will focus more on solving the transparency and ethics issues by offering clear communication between the vehicle and user.

2. Description of the Specific Human-cyberphysical System Problem

Building trust between humans and cyberphysical systems boils down to understanding human psychology and designing systems that address human concerns. One way to understand the human mentality when driving is to analyze the eye movements of drivers. Eye movements reveal a lot about what humans are worried about in specific moments. For example, when doing a lane change, a human might tend to look more toward his or her left/right mirrors to check for incoming cars. A human might also look more at his or her speedometer when driving in an urban area. If one can successfully determine what concerns a human most in certain situations and build an interface that allows the car to communicate to the user that it is aware of the situation and will handle it in a safe manner, the amount of trust between CAVs and the general population will greatly improve. Thus, the specific human-cyberphysical problem this report covers is gaining important information from human driving behavior, with an emphasis on eye movement data.

3. The Challenges of Reaching a Functional System

Some of the challenges involved with reaching a functional system include acquiring unbiased, strong data for analyzing human behavior. Having misleading data could lead to misleading results, which might lead to poor interface design and to people becoming less accepting of CAVs. Another challenge would be making this interface universally friendly. People from different areas and backgrounds might yield widely different results, leading to adaptability issues. Overall, one is hopefully able to standardize this interface just as the manual car control system has been standardized. Another challenge would be meeting ideal expectations within

engineering limits. If the data analysis reveals something that humans are apprehensive about, such as hitting a deer in the middle of the night, the self-driving car may not be able to fully handle this fear. Certain factors are simply out of reach for today's technology, and this is a challenge for designers, who should develop a transparent interface between the vehicle and the user. There are many more potential challenges associated with this system, and the best way to narrow the scope would be to analyze human driving behavioral data and see what the results are.

4. The Technical Problem and the Research Setting

The research took place at the German Aerospace Center (Deutsches Zentrum für Luft- und Raumfahrt) under the Institute for Transportation Systems under the supervision of David Käthner. My supervisor obtained several datasets by placing eye trackers on a driving simulator and having the participants go through several driving scenarios such as changing lanes on the highway, driving through an urban area, and driving on a plain road. The technical problem was to take the data and find revealing information. As a result, the first step was figuring out how to best analyze eye tracking data. Unsurprisingly, modeling eye movement is a huge area of research and there are several important distinctions. A fixation is generally defined as a pause in eye movement for the human to concentrate somewhere, whereas a saccade is rapid eye movement that occurs when the human eyes move between fixations. The first technical problem is separating between these two eye movements. By analyzing their angular movement features and comparing the movements to threshold values gained from previous studies, a mostly clear separation was made between the two. From these two distinctions, one

generally wants to detect areas of interest (AOIs), which are places that the human looks at over time based on a number of fixations. AOIs are important because they convey what the human is most concerned about when driving. Having this information can directly tell us what a self-driving car should communicate when driving in a certain scenario. Thus, the best way to find revealing information is to accurately classify AOIs.

The design approach is to use unsupervised learning to obtain a data-based method for determining AOIs. In previous research, AOIs were frequently pre-defined with pre-set boundaries. This led to a lot of inaccuracies because experiments frequently did not conform to expectations. Thus, using previous data to learn about more accurate boundaries led to a more robust model. Unsupervised learning models specifically used are the Gaussian Mixture Model (GMM) and the custom iterative KMeans algorithm. The custom iterative KMeans model would loop through the entire dataset, collapsing a set time frame into a window of 2-D positional coordinates, and run the KMeans model several times on this map, with different starting parameters. By have changing parameters, particularly the number of clusters to search for, one can find an optimal solution by utilizing the elbow curve, which plots the accuracy score with respect to the number of input clusters. This would allow one to select the number of clusters that will not overfit or underfit the data. Both yielded promising results: the GMM was faster, but the KMeans method offered a deeper look into the data. The Kmeans method does iterate through the entire dataset, and it is also better for real time analysis.

The result is a way to distinctly classify different AOIs from one another. For example, the driving simulation had 5 AOIs with the left mirror, right mirror, speedometer, main window, and the display pad, which acted as a distraction humans tend to face when driving, i.e. the

radio. The method successfully made a distinction between all five, although the exact level of success depended on how the parameters were set up. As a result, this allows data scientists to classify a positional eye data point as being a fixation and to determine what area of interest this fixation contributes to. By having these as features, data scientists can now much more clearly see what concerns a human the most when driving in certain scenarios. This allows for more robust analysis, particularly with supervised learning and can reveal a lot of information about human driving behavior, necessary for effective communication between CAVs and users.

5. Future Research

In terms of the social aspect, trust is the main issue and needs to be built up over time. Tech companies and research facilities need to openly communicate their intentions and get society used to the fact that self-driving cars represent the future. This would not only increase potential revenue from the market but also increase the likelihood that policy is passed. Future research can be done in the human-cyberphysical interaction aspect by finding more ways to remove the black box between humans and self-driving technology. The interface itself needs specifications and could range from a display screen to certain warning lights on important car components. This would mean trying out different designs and running experiments to see what system humans find most transparent and trustworthy.

Saatvik Mohan

Decision Making in Smart Grid Cybersecurity

Mentor: Mathias Uslar, OFFIS

1. Problem and Context

The world is in the midst of a major transformation in electric power infrastructure, which affects not only governments and businesses but also homes and individuals. In order to increase efficiency, security, and privacy, there must be a concerted effort to understand and communicate fully the risks associated with different systems and the smart grid as a whole (NIST 2014).

With respect to efficiency, a lack of information for the players of the smart grid (governments, utilities, and businesses) leads to poor decisions by all parties. Therefore, in order to ensure that the smart grid functions efficiently, all players must be privy to the same total information. Moreover, security is important because the modern economy cannot function without proper availability and integrity in the power infrastructure. Equally important are the social and ethical implications of security. Without reliable energy, the most at-risk individuals and institutions of society will be affected the most. The result is energy poverty, a situation where the poorest individuals have the least access to power and are more likely to remain in poverty as a result of being unconnected (Nussbaumer et al. 2012). This issue is international, with stunted economic growth in areas with unreliable energy.

Finally, privacy is an issue that is becoming more and more pressing in a modern smart grid. Now that information-collection has become incredibly accurate, there must be effective strategies in place to protect smart grid-related data. The legal implications of breaches of privacy, especially in terms of smart meters, have been well-documented and have attracted organized advocacy groups. Although there are other concerns about the health risks, the majority of the opposition is aimed at the "pervasive anger at being forced to accept devices that can report on activities by appliance in a household" (Hess 2014). There should be expected privacy issues coming to the forefront of international conversation beyond the smart meter issue.

2. The Specific Human-Cyberphysical Problem

The specific human-cyberphysical problem in my project was how decision makers view the risks associated with different systems or types of systems. With more comprehensive and accurate knowledge, smart systems can better decide where to allocate resources for security. Not only have systems within the smart grid become more interdependent, there are an increased number of vulnerable entry points and more data that can be stolen. My project draws upon *Guidelines for Smart Grid Cybersecurity*, a comprehensive, three-volume, advisory report published by the National Institute of Standards and Technology (NIST) and the European Union's *Smart Grid Information Security*.

The NIST report presents "an analytical framework that organizations can use to develop effective cybersecurity strategies tailored to their particular combinations of smart grid-related characteristics, risks, and vulnerabilities" (National Institute of Standards and

Technology 2014). The primary goal of the report is to develop a high-level set of cybersecurity requirements that can be used by all stakeholders in the smart grid. As technology has evolved, so has cybersecurity. Cybersecurity must now address “not only deliberate attacks launched by disgruntled employees, agents of industrial espionage, and terrorists, but also inadvertent compromises of the information infrastructure due to user errors, equipment failures, and natural disasters” (ibid.).

3. Challenges of Meeting a Functional System

There are many challenges that make it difficult for decision makers. First, there has not been a formalized methodology to utilize the information concerning different systems. There is no consensus for classifying systems based on their threat, criticality, and impact because of the subjective nature of the smart grid. For example, while experts may agree that a certain system is more important than another, they may be basing that decision on different reasons. For example, the NIST report initially defines 49 actors, but the security characteristics are only defined for 46 because these systems are not disparate, and one can argue that there are more than 46 systems or fewer. As a result, any formalized process to define risk for decision makers will have to be reevaluated. Additionally, the metrics used in determining risk will have to be reevaluated along with the types of systems included. The NIST report only took into account three security characteristics (confidentiality, impact, and availability) among many possibilities. The primary issue is that most of these characteristics are very difficult to define for each individual system.

The final major issue affects decision makers directly: how does one decide which systems to fortify most even when the risk is known? Ideally, all risky systems would have mitigations in place to avoid any breach. But realistically speaking, decision makers have to balance maximizing safety with maximizing profit. Consequently, they must make difficult decisions in terms of which systems to fortify. Experts must simplify the risk methodology so that it is easier for policy decisions to be made. This transfer of information from expert to decision maker will be a major challenge going forward.

4. Technical Problem and Research Setting

I conducted my project at OFFIS in Oldenburg, Germany, under the tutelage of Dr. Mathias Uslar. There were two different risk formulas that were developed: one for the 46 actors that are defined in the Logical Inference Model (Formula 1) and one for 8 categories that were created from the 22 LICs (Formula 2, Table 1). These 8 categories were created to provide a joint risk for similar systems. Risk is a function of threat (how critical the system is), vulnerability (how difficult is it to breach), and impact (how devastating would an attack be). The risk formulas follow this form. As mentioned before, even though the NIST *Guidelines* document identifies the actors within the smart grid, it does not provide quantifiable threat levels. In order to do so, the Smart Grid Architecture Model (SGAM), found in *Smart Grid Information Security*, was used. Using SGAM's High Level-Guidance Table (Table 1), the systems were mapped onto it based on where they fell on the intersection of domain and zone (Figure 1), and are classified by threat level.

SGIS-SL HIGH LEVEL GUIDANCE*						
3 – 4	3 – 4	3 – 4	2 – 3	2 – 3	MARKET	ZONES
3 – 4	3 – 4	3 – 4	2 – 3	2 – 3	ENTREPRISE	
3 – 4	5	3 -4	3	2 – 3	OPERATION	
2 – 3	4	2	1 – 2	2	STATION	
2 – 3	3	2	1 – 2	1	FIELD	
2 - 3	2	2	1 - 2	1	PROCESSES	
GENERATION	TRANSMISSION	DISTRIBUTION	DER	CUSTOMER		
DOMAINS						

Table 1 – Threat Value Recommendation per Layer
(*Smart Grid Information Security* 10)

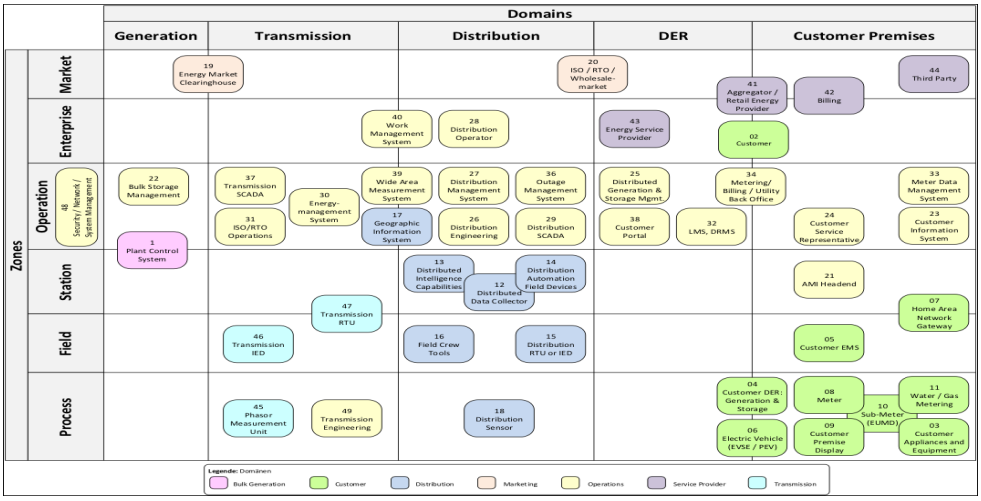

Figure 1 – Actors Mapped onto SGAM

To classify the systems and categories of systems' vulnerability, the number of interfaces and domains is connected to each system. A greater number of interfaces indicates that an actor is more vulnerable because there are more entry points from which an attacker can strike. Additionally, the more domains that an attacker can gain access to, the more vulnerable the entire smart grid becomes. Both of these factors were taken into account. The components of impact are the security characterizations defined for each system. The components of impact used are confidentiality, availability, and impact, which are defined by value in the NIST report. The final formulas are the following:

- Formula 1: Risk = (Threat Value*Threat of Bordering Actors)+((Number of Logical Interfaces/3)*Number of Domains)+(0.25 * (C Score+0.5)*(I Score+1)*(A Score+2))
- Formula 2: Risk = 2.25*Threat Value + (0.7)*(Number of domains+(Average Number of Actors/2)) + (0.125*(C Score+0.5)*(I Score+1)*(A Score+2))

The outcomes of the project are two formulas that can quantify the risks associated with particular systems and types of systems. In the Appendix to this report, the two tables detail how different scaled risk values are obtained using the formulas. These tables and formulas can help decision makers assess the risk of systems they're responsible for and control where mitigations are placed.

5. Future Research

Despite the positive contributions of the risk formulas, there are important areas of future research that need to be considered, especially in terms of the social, legal, and ethical issues, as well as the decision-making process. Currently, the formulas do not take into account the concept of energy poverty, but there may have to be some metric that considers more than the availability of resources. There should be a way to quantify the risk attached to energy availability for impoverished people. Additionally, the privacy metric, confidentiality, needs to be reevaluated for the weighting it has been assigned. Historically, its importance is designated as well below availability and integrity, but in the modern era, privacy issues are of utmost importance. Decision makers must take into account not only the technical requirements of cybersecurity, but also the social needs of their customers.

Appendix

	Actor	Threat	Threat of Bordering Actors	Number of Logical Interfaces	Number of Domains	Confidentiality Score	Integrity Score	Availability Score	Formula 1 (Unscaled)	Formula 1 (Scaled)
1	Plant Control System	4	5.00	3	2	1.00	3.00	2.00	28.00	5.68
2	Customer	3	2.50	4	2	2.50	2.00	1.25	17.48	3.28
3	Customer Appliances and Equipment	1	1.25	4	1	1.25	2.25	1.75	7.92	1.09
4	Customer DER: Generation & Storage	2	1.50	2	1	2.00	2.50	2.00	12.42	2.12
5	Customer EMS	1	2.17	12	3	1.89	2.42	1.60	21.51	4.20
6	Electric Vehicle(EVSE/PEV)	2	1.33	3	1	1.33	2.33	1.67	9.27	1.40
7	Home Area Network Gateway	2	1.92	13	3	1.88	2.55	1.56	24.35	4.85
8	Meter	1	1.71	7	4	1.78	2.83	1.44	18.57	3.53
9	Customer Premise Display	1	2.00	3	1	1.00	2.00	2.00	7.50	1.00
10	Sub-Meter	1	1.50	6	1	2.00	3.00	1.00	11.00	1.80
11	Water/Gas Metering	1	2.00	1	1	2.00	3.00	1.00	9.83	1.53
12	Distributed Data Collector	2	2.00	2	1	1.00	2.00	2.00	9.17	1.38
13	Distributed Intelligence Capabilities	2	2.00	1	1	1.00	2.00	2.00	8.83	1.30
15	Distribution RTU or IED	2	3.14	7	2	1.00	2.67	2.25	16.80	3.12
16	Field Crew Tools	2	3.57	7	2	1.29	3.00	1.86	18.70	3.56
17	Geographic Information System	5	3.25	4	2	1.50	3.00	1.88	26.67	5.38
18	Distribution Sensor	2	2.00	1	1	1.00	2.00	2.00	8.83	1.30
19	Energy Market Clearinghouse	4	4.00	5	3	3.00	3.00	2.00	35.00	7.28
20	ISO/RTO/Wholesale-market	4	4.25	4	3	2.75	3.00	2.00	34.00	7.06
21	AMI Headend	2	2.80	10	4	2.85	3.00	1.52	30.73	6.31
22	Bulk Storage Management	4	5.00	2	1	1.00	3.00	2.00	26.67	5.38
23	Customer Information System	3	3.00	11	4	2.45	2.73	1.42	33.09	6.85
24	Customer Service Representative	3	3.00	2	2	3.00	2.00	1.00	18.21	3.45
25	Distributed Generation & Storage Mgmt.	3	3.67	3	2	1.00	3.00	2.63	19.94	3.84
26	Distribution Engineering	4	3.00	2	2	1.00	3.00	2.00	19.33	3.70
27	Distribution Management System	4	3.47	15	5	1.49	2.93	2.10	46.89	10.00
28	Distribution Operator	4	3.33	3	2	1.00	3.00	2.00	21.33	4.16
29	Distribution SCADA	4	3.33	12	5	1.70	3.00	2.35	42.90	9.09
30	Energy-Management System	5	4.13	8	4	1.29	3.00	2.14	38.69	8.13
31	ISO/RTO Operations	5	3.91	11	5	1.54	2.88	1.75	45.30	9.64
32	LMS, DRMS	3	3.00	7	2	1.57	3.00	2.14	22.25	4.37
33	Meter Data Management System	3	3.00	4	2	2.50	3.00	1.50	22.17	4.35
34	Metering/Billing/Utility Back Office	3	3.00	4	3	2.75	3.00	1.58	24.65	4.92
36	Outage Management System	4	3.00	4	2	1.50	3.00	2.00	22.67	4.47
37	Transmission SCADA	5	3.75	12	4	1.00	3.00	2.34	41.27	8.72
38	Customer Portal	3	3.00	2	2	3.00	2.00	1.00	18.21	3.45
39	Wide Area Measurement System	5	4.25	4	2	1.00	3.00	2.13	30.10	6.17
40	Work Management System	4	3.50	6	2	1.33	3.00	1.83	25.03	5.01
41	Aggregator/Retail Energy Provider	3	3.67	9	4	2.48	2.89	1.65	33.57	6.96
42	Billing	3	3.00	8	3	3.00	3.00	1.11	27.89	5.66
43	Energy Service Provider	3	1.00	1	2	2.67	3.00	1.67	15.28	2.78
44	Third Party	3	2.33	3	3	2.00	2.50	1.50	17.66	3.32
45	Phasor Measurement Unit	2	5.00	3	2	1.00	3.00	2.50	18.75	3.57
46	Transmission IED	3	3.50	2	2	1.00	3.00	2.25	18.21	3.45
47	Transmission RTU	4	3.50	2	2	1.00	3.00	2.25	21.71	4.25
49	Transmission Engineering	2	3.50	2	2	1.00	3.00	2.00	14.33	2.56

Table A-1 – Formula 1

	Description	Threat Value	Number of Domains	Average Number of Actors	Confidentiality Score	Integrity Score	Availability Score	Formula 2 (Unscaled)	Formula 2 (Scaled)
1	Interface between control systems and equipment	3.70	3.00	10.00	1.00	3.00	2.50	17.30	6.6
2	Critical information exchange between utility and third party	4.25	4.00	6.67	1.67	3.00	2.00	19.03	7.7
3	Non-critical information exchange between utility and third party	3.23	4.00	6.50	2.00	3.00	1.50	16.72	6.3
4	Metering & billing	2.73	4.00	11.20	2.40	2.80	1.60	17.83	7.0
5	Distrbution domain	2.00	1.00	2.50	1.00	2.00	2.00	8.33	1.0
6	Controlled system to back-end system	3.33	4.00	8.00	1.67	3.00	2.00	17.43	6.7
7	Customer domain	1.63	1.00	8.00	1.00	2.00	2.00	9.41	1.7
8	Interface between security/network/system management consoles and all networks and systems	3.08	4.00	12.00	3.00	3.00	3.00	22.69	10.0

Table A-2 – Formula 2

References

European Commission. 2012. *Smart Grid Information Security*. European Commission Smart Grid Mandate, pp. 8–10.

Hess, David J. 2014. Smart Meters and Public Acceptance: Comparative Analysis and Governance Implications. *Health, Risk & Society* 16(3): 243–258.

National Institute of Standards and Technology. 2014. *Guidelines for Smart Grid Cybersecurity*.

Nussbaumer, Patrick, et al. 2012. “Measuring Energy Poverty: Focusing on What Matters.” *Renewable and Sustainable Energy Reviews* 16(1): 231–243, 2011.07.150.

Tiger Mou

Implementation of an Oriented Bounding Box Distance Sensor for Virtual Valet Parking

Mentor: Eike Möhlmann, OFFIS

I. General Problem and Context

Many researchers are working on autonomous vehicles to get them ready for public use. However, a major impediment to this deployment is that the autonomous driving software in these vehicles must be safe and robust. Unless we can somehow verify that the software works and will not cause any accidents, drivers will not feel safe handing over the responsibility to the software. Therefore, it is necessary to test this software thoroughly before releasing it to the public.

There are several ways to test the software for autonomous vehicles. One method is to simply run the vehicle in the real world and fix any issues that come up. However, this is costly and dangerous, and it requires having the physical vehicle ready for testing, which can be very expensive. Additionally, there is a chance that the vehicle may get into an accident. If the autonomous driving software injures or kills another person, the company or driver is then held liable for the damages, which may bring a negative attitude towards autonomous vehicles and the company responsible for the incident.

Another method to test autonomous vehicles is to run the software in a simulation. By testing the autonomous driving software in a simulation, there is an extremely low chance that anyone will get hurt. Additionally, there is no need to have a working physical vehicle because

only a software interface with the testing platform is required. The testing software could also target test specific cases and provide more rapid development feedback. The simulation could also be run faster than real time, making it more time efficient than just a physical test run. Since there is a limit to how realistic a simulation can be, testing should be done with both methods, as described by the ENABLE-S3 project.

It is necessary to start small and incrementally build up this testing software. By starting with a small use case, it is possible to significantly narrow the focus of the software and help simplify the software development and testing. However, the testing software would then only be useful for a very specific scenario. As a result, we need to use and write flexible software that can be easily expanded on and reused for other use cases.

2. Description of the Specific Human-Cyberphysical Problem

The use case for this software is valet parking with a parking area management. In this use case, a driver will drop off a vehicle at a designated location. The driver will hand the vehicle over to the local parking controller, and the parking controller will assign the vehicle to a parking spot. The vehicle will then autonomously navigate to the parking spot. When the driver wants the vehicle to return, the controller will notify the vehicle, and the vehicle will autonomously navigate back to a designated location. (The requirements that this use case is designed to follow is outside the scope of this report.)

3. Challenges of Reaching a Functional System

There are many challenges that this test software must overcome. For example, the simulation must be feature rich in order to simulate a variety of scenarios and to accommodate different approaches to autonomous driving. During the simulation, metrics must be accurately calculated and reported to the test manager. These simulations should also be fast and easy to run. Although it is not possible to test every scenario, software simulation can test a greater number of scenarios than physical tests. Nevertheless, it is very important to focus on the important scenarios rather than focus on every possible scenario. Additionally, this test software has to be as good as or better than running physical tests; otherwise, there would be no reason to use it. The code written for this test software must also be flexible so that it can be reused for other test cases.

4. Technical Problem and Research Setting

I conducted my research at OFFIS in Oldenburg, Germany, under the supervision of Eike Möhlmann for research on the valet parking scenario. The existing work used software bu VIRES Simulationstechnologie GmbH, which has software called Virtual Test Drive (VTD). This software is a very feature-rich 3D simulation platform for a variety of vehicles in a number of computer-generated scenarios. Although VTD can be used to test autonomous driving software, it is missing a few pieces that are necessary for quickly running comprehensive tests. Two of these missing pieces are a custom distance sensor and valet parking scenarios.

Within the VTD software, objects are represented as 3D bounding boxes. Each object has x, y, and z coordinates for position and heading, pitch, and roll for orientation. Given that

these bounding boxes have an orientation, a simple delta x and delta y calculation is not enough. The distance sensor processes a list of detected objects and calculates the distance between the bounding box of the sensor's vehicle and the bounding box of the detected object. This distance sensor must be integrated with the existing framework. For every frame of the simulation, the sensor will calculate the distances to the area that the sensor can detect, and then the sensor will forward the desired data to an external listener. Depending on how the sensor is configured, it may use different coordinate systems to represent the position and orientation of each object. The distance sensor has to understand these coordinate system settings and be able to switch between them. As the sensor was implemented, the results were verified by comparing the calculated points and distances with the reported locations and orientations of each object.

My primary role in the implementation of the valet parking use case was to implement an oriented bounding box distance sensor for the VTD software. This distance sensor was needed to run various calculations on the objects in the simulation. The sensor needed to be customizable for future research and development. The distance sensor was designed to be mounted on a simulation vehicle for running calculations in real time to report metrics to a test controller. Although the calculations could have been performed from outside the simulation, there were performance losses from streaming the data to an external computer or program.

A number of additional features for the sensor were requested. Some of these features included switching coordinate systems, having a sensor on each side of the vehicle, and filtering the reported data to send only the distance to the closest object or most critical object, where criticality is a function on the speed, distance, and direction of the objects to be defined in the

future. It was necessary to have a sensor on each side of the vehicle since collisions may occur on all sides of the vehicle. These features were tested to ensure that they work as expected.

My implementation of the distance sensor accomplished all the basic required tasks. It successfully read the distances of detected objects and sent the data through a predefined port to another computer on the local network. Because it was always possible that there was a bug somewhere in the code, some test code for this distance sensor was needed. Additionally, all of the extra features implemented in this distance sensor made it more complicated and introduced more possibilities that there was a bug somewhere in the code, all of which potentially slowed down the simulation.

5. Future Research

Although the distance sensor worked correctly, there is definitely more future work that can be done on the distance sensor and the valet parking testing scenario. The distance sensor could definitely be improved by adding even more features and by improving the efficiency of the algorithm. More calculations could also be added and reported, such as the criticality of the detected objects, collisions events if they happen, and the ID of the object at fault for a collision. Although criticality can be a very useful metric, it can be difficult to correctly and accurately calculate. As a result, an outline of the method with the required data structures was set up, and a basic computation was done and reported. In terms of the valet parking scenario, we could always try to make the simulation more realistic. For example, since a major concern for these autonomous vehicles is how they interact with and fit in with society, we could try to simulate scenarios where pedestrians try to interact with an autonomous vehicle. For example,

when a pedestrian is interacting with an autonomous vehicle in an unexpected way, the vehicle must still try its best to avoid causing any accidents. Additionally, it is important to understand how the autonomous software might respond to ethical dilemma situations. For example, it is entirely possible that the vehicle will run into some variation of the trolley problem. This leads to the problem of whether or not drivers would approve of the autonomous vehicle's decision making.

Joshua Petrin

Human-Machine Interaction Development in Autonomous Vehicles

Mentor: Johann Telsch, Deutsches Zentrum für Luft- und Raumfahrt

1. General Problem and Context

In the context of up-and-coming cyberphysical systems such as unmanned aerial vehicles and autonomous cars, the need for trustworthy human-machine interaction (HMI) is growing, even though it was already very large. HMI has been developed over the years in various societal engineering inventions, such as airplanes, automobiles, and even rentable bicycles (such as Ofo's). Recently, much research funding has been invested in HMI for autonomous vehicles (AVs) because in order for them to be integrated into society, they must first be able to interact with the humans.

Autonomous vehicles offer several benefits over human drivers. For example, they can objectively detect obstacles, they can control their motors as precisely as their programmers can, and they do not have to have windows. However, it is still difficult for human drivers to drive with them. One reason this is the case is because AVs obey all traffic laws to the best of their abilities. Accidents can occur between a human driver and an AV because the driver expected an AV to disobey a traffic regulation rather than to stringently obey it. Confusion can occur between a human driver and an AV when waiting at a stop sign, and the AV does not know if the human driver wants to yield the right-of-way.

The majority of AV-human driver collisions have occurred as a result of this type of confusion. In fact, Kia Kokalitcheva, an Axios journaler, writes that in the 38 accidents involving

moving-AV and moving-human cars, only one was the fault of the AV [1]. According to Peter Hancock, a psychology of professor at the Institute for Simulation and Training at UCF, the reason this could be is because putting AVs on the road creates a disturbance in the typical patterns of driving [2]. For instance, if an AV goes 30 mph on a 30 mph-designated road that most people go 45 mph on, there is a higher probability that it will get rear-ended. Also, if an AV is at an intersection in accordance with traffic guidelines, and if it inches slowly into the intersection, human drivers might perceive this as a yielding of the right-of-way.

2. Description of the Specific Human-Cyberphyscial Problem

There are several possible causes for AV-human confusion, but they are not limited to cars. Confusion can also occur between AVs and pedestrians at a crosswalk. How are pedestrians supposed to know if an AV sees them? And what if it does not? Is it still safe to cross the crosswalk, or will the AV hit them? These are important problems to consider when introducing AVs to the roadway.

HMI seeks to mitigate these confusions. If humans can tell what an autonomous car perceives, or if humans can know what the autonomous car is expected to do, then there will be fewer unknowns in interactions between humans and machines. Consequently, there will be less frustration, technology will be safer, and operation will be more effective.

3. Challenges of Reaching a Functional System

However, despite good HMI being so necessary for future AVs, there are several inherent setbacks for its development. The biggest setback is dangerous experimentation.

Many of the HMI interfaces on the exteriors of vehicles have the potential to seriously hurt people who misunderstand them or make poor judgements because of them. Ironically, the greatest setback to HMI development for humans is that prototype HMI cannot be deployed in a civilian setting. Instead, it must undergo verified research and development and must adhere to several standards before being integrated. Therefore, HMI development can be very slow, and sadly this delay often keeps it from being integrated into the systems that need it the most.

Virginia Institute of Technology performed an experiment that involved driving a costumed driver around in a vehicle that was painted like an autonomous car. The driver was disguised as the car's car seat so that he was invisible to any passers-by. Several HMI apparati were installed on the interior and exterior of the confederate vehicle. The premise was to make the car look like an AV, even though it was only being driven by a disguised human [3]. The goal was to see which configuration of HMI worked better with pedestrians crossing the road. Several other experiments such as this one have been performed, including several in California. The DLR has also wanted to perform experiments such as this one.

4. Technical Problem and Research Setting

Last summer, I was assigned to the HMI group at the Deutsches Zentrum für Luft- und Raumfahrt to work on car-to-driver and car-to-pedestrian indicators for autonomous vehicles. The Partnerships for International Research and Education internship is an NSF-sponsored opportunity to visit Germany and study societal-scale cyberphysical systems. The structure I worked on the whole summer was an LED strip they wanted to program for use on the outside of the vehicle. I wrote code for interfacing programmatically with the LED strip, although I did

not have time to complete the interface because of several setbacks, including the duration of my internship.

The HMI department at the DLR asked me to write code for an Arduino interface to the LED strips that were to be mounted to the external of an autonomous vehicle. However, they wanted the code to do something very specific. The timing protocol for the LED driver IC (the LPD1886) was unique to all other driver ICs, and the code that would be used for programming the Arduino for the LPD1886 could not be used for any other LED driver IC. As a result, Johann Telsch, my project supervisor, asked that I would create a program for the Arduino that would directly output whatever signal it received from the USB serial signal. By writing this program, he hoped that the HMI department would be able to program a system-based serial client for every LED strip they decided to use.

In addition to this, the HMI department asked me to design a GUI that could create bitmap files for LED strip animations. They asked me to use Unreal Studio because it was a powerful way to interface with the simulation software that the DLR uses; however, until they had not used it, so they wanted me to pave the way in Unreal Studio. This GUI would showcase an LED strip model and a timing diagram so animations could be created by scientists without them using an image-editing software or learning C++. Sadly, I failed to complete every project I worked on this summer, and all that I could submit to the DLR was unfinished bits of C++ code. There were a few reasons for this, but the main reason was that I ran out of time.

However, despite limitations, I was able to develop a good final work product. The entire backend of the LED strip GUI was completed, and it came in a complete, header-only library. Also, I created a good framework for the timing requirements of the Arduino

programmer, so any knowledgeable programmer who wanted to complete my library could read my existing documentation and do so.

5. Future Research

In the context of car-pedestrian HMI, the DLR wants to continue its HMI research. They plan on replicating the experiments performed by Virginia Tech and have been designing a car seat costume to use for experimentation. Once it is complete, they will use the costume in conjunction with the LED strip module in a social experiment.

More work will be made by the DLR in Unreal Studio. They want to use it to program the LED strip module. Programs will be made by psychologists without awareness of the code for the strip, and experiments for the types of animation colors used in the strip will be performed. The LED strip that I worked on will eventually be implemented in an internal and external HMI scheme that will interact with the passenger of the AV and the pedestrians outside of the AV. This scheme will improve interactions between pedestrians and AVs as well as prevent pedestrians from being harmed by unwittingly crossing autonomous cars.

References

[1] Kokalitcheva, Kia. “People Cause Most California Autonomous Vehicle Accidents.” Axios, 29 Aug. 2017, www.axios.com/california-people-cause-most-autonomous-vehicle-accidents-dc962265-c9bb-4b00-ae97-50427f6bc936.html.

[2] Hancock, Peter. “Are Autonomous Cars Really Safer than Human Drivers?” *The Conversation*, 18 Sept. 2018, theconversation.com/are-autonomous-cars-really-safer-than-human-drivers-90202.

[3] O'Kane, Sean. “Ford Hid a Man inside a Car Seat to Test Reactions to Self-Driving Cars.” *The Verge*, The Verge, 13 Sept. 2017, www.theverge.com/2017/9/13/16303720/ford-self-driving-car-test-seat-costume.

Eric Yeats

Improvements to Traffic Criticality Metrics for Highly Autonomous Vehicles

Mentor: Thomas Peikenkamp, OFFIS

1. General Problem and Context

There is a race within the automotive industry to develop technology for highly autonomous vehicles (HAVs), which rely on accurately perceiving and cooperating with their environment in order to be safely mobile without human input. HAVs have the future potential to improve passenger safety, traffic throughput, and universal mobility by removing humans from the control loop; however, they face a number of societal-scale issues before their large-scale adoption and associated benefits can become reality.

One such societal-scale hurdle that HAVs face is a considerable lack of consumer trust regarding their safe comportment. There have been recent high-profile autonomous vehicle accidents (Uber, in Tempe, Arizona, 2018 and Tesla, in Williston, Florida, 2016) [1, 2], which decrease consumer confidence in the safety of HAVs. Without the option of referencing a clean track record of HAV operation, it is difficult to communicate credible guarantees of HAV technology safety to the public.

As independent agents on the road that will interact directly with humans, HAVs are expected to behave ethically, as another human would. Numerous examples of ethical dilemmas including the infamous trolley problem are brought up in the context of HAVs [3]. Awareness of these issues intensifies the debate on how HAVs should interact with humans, and it brings this unanswered question into the public spotlight.

In the legal domain, HAVs face additional societal-scale challenges. Lawmakers must produce new legislation and standards for HAVs with little knowledge of their technical aspects and with little coordination on a national and international level. These specifications would need to govern the innate behavior of HAVs in a specific and measurable way, yet still be general enough to be applicable over a wide variety of situations.

In each of these societal-scale challenges faced by HAVs, there is a disconnect between the technical aspects of future HAV technology and the public perception of HAV technology. A way that car manufacturers could attempt to bridge these issues is to express safe HAV traffic behavior with tools that are scientific and measurable, yet still grounded in aspects of driving that are familiar to the public. One promising technique in the safety-critical systems domain is employing criticality metrics, which have additional value when used as a technique for communication with the public.

2. The Specific Human-Cyberphysical Problem

Traffic criticality metrics are well-established concepts in literature that relate the motion of an *ego* vehicle and its surroundings (other vehicles and objects) to a measure of danger for *ego* in the given scenario. A metric will often express a time left until a critical event (with lower values indicating higher criticality) or consider the speed of the participating vehicles to yield a measure of danger.

Each metric can represent a unique source of danger for *ego* or can be a composition of other existing criticality metrics. For example, a metric can represent the time to collision (Time to Collision, TTC) [4] between a current trajectory of *ego* and another vehicle, a metric can

represent the time a vehicle takes to occupy the space of a previous moving vehicle (Post Encroachment Time, PET) [4], or a metric can be an expression that considers both TTC and *ego*'s current speed to yield a measure of momentum change required to stop *ego* over a time period in a crash scenario (Criticality Index, CI) [5].

It is important to note that each criticality metric represents an aspect of danger in a critical scenario rather than a specific formula. Criticality metrics are calculated based on the motion model of a vehicle, and thus the formula used to determine them could change with different motion models and situations. Criticality metrics, as an aspect of danger in a scenario, also are not applicable in all scenarios. For example, in a trajectory in which *ego* is not predicted to crash has no TTC defined.

Examples of Criticality Metrics

Criticality Metric	Example Formula	Source of Criticality Metric
TTC	$TTC = \frac{X_o(t) - X_e(t)}{\dot{X_e}(t) - \dot{X_o}(t)}$	[4]
PET	$PET = t_{o,space} - t_{e,space}$	[4]
CI	$CI = \frac{v_e^2}{TTC}$	[5]

3. Challenges of Meeting a Functional System

Given the broad range of situations that can be encountered on the road, HAVs cannot be programmed to react to every concrete scenario. Rather, HAVs' behavior can only realistically be specified in abstract terms. Abstract traffic scenario modeling tools such as Traffic Sequence Charts (TSCs)[6] are being used and are continuously under development for this purpose. In order to reflect the level of abstraction of the design process, criticality for

HAVs would need to be characterized in abstract terms, and the formulas which define their measurement would also need to be expressed in a uniform manner.

Although criticality metrics can represent general sources of traffic-induced danger, their definitions in literature are often highly contextual. Their applicability in certain traffic scenarios is not always made clear by their definitions in literature, and their formulas are highly coupled with the underlying vehicle motion model and the small details of the traffic situation they are used in.

An example of the tight coupling of criticality metrics to situational- and motion-model-context can be found in an influential criticality metric publication [7]. The publication describes an algorithm for making collision mitigation maneuver decisions in the context of turning at a busy intersection. Using a "curved coordinate system" in order to simplify the motion model of *ego* turning through an intersection, the publication defines several criticality metrics to consider in the algorithm: a_{req} (required braking deceleration to avoid collision) and Time to Touch (TTT, time at which *ego* 'touches' another vehicle with zero relative velocity after undergoing a_{req}). Four permutations of the contextual example are considered in which the other vehicle comes to a standstill, evades *ego*'s predicted path, both, or neither. As a result, there are four different methods for calculating a_{req} and TTT for this single example. This inconsistency due to the tight coupling of criticality metrics to concrete contexts makes current criticality metrics cumbersome to use in abstract HAV behavior specification.

4. Technical Problem and Research Setting

In the summer of 2018, I worked at a research institution in Oldenburg, Germany, called OFFIS, which is oriented towards solving emerging issues in energy, health, and transportation. My assignment domain was transportation, and I helped Thomas Peikenkamp and other members of CrEST investigate the incorporation of criticality metrics into TSCs, their graphical traffic scenario language based on formal semantics [6].

In order to decouple criticality metrics from their situational context and motion model such that they are more suitable in abstract applications, a parameterization of criticality metrics can be made based on actor trajectories. This parameterization defines an actor trajectory T_{act}, which specifies an evolution of position for an actor over time.

$$T_{act} : \text{Time} \Rightarrow \text{Position}$$

Additionally, a collision relation *c* can be defined such that a Boolean 'true' value is returned if there is a collision between two actor trajectories over a time interval. If there is a collision, a time t_c specifies the time at which the collision occurs.

$$c(T_a, T_b) : T_a \times T_b \times \text{Time} \Rightarrow \text{Boolean}, t_c$$

The concrete implementation of the collision can be realized at a later time and can be specialized for specific contexts. A potential implementation could be to test possible trajectories from a reachable set based on an abstract maneuver class similar to what is described in the *Maneuver-based motion models* section of Lefèvre *et al*. 2014 [8].

With trajectories and a collision relation, expressions can be developed in an intuitive manner that represents criticality metrics. For example, if there is a collision between actor *a* and actor *b* over a time period $[t_0,t_1)$, TTC over time is defined as follows:

$$\mathrm{TTC}(t) = t_c - t$$

If there is a collision specified between actor *a* and the *instantaneous position* of actor *b* at time instance *t*, then PET(*t*) could be expressed as follows:

$$\mathrm{PET}(t) = t_c - t$$

And finally, if a collision is specified between *ego* and another actor over a time interval, the criticality index for *ego* at time *t* can be expressed as follows:

$$\mathrm{CI}(t) = \frac{|T_e'(t)|^2}{t_c - t}$$

Using this framework, criticality metrics can be defined in abstract terms yet still retain their quantitative meaning. Many other established criticality metrics besides TTC, PET, and CI can be interpreted in this manner.

5. Future Research

Criticality metrics have proven their value as a technical tool for safety-critical systems. However, when the social implications of their application to HAVs is considered, the technical value of criticality is enhanced and new applications for criticality metrics as tools for communication become possible. As established scientific measures of danger that are expressed in familiar driving terms, socially-focused criticality metric research could promote an understanding of HAV safety-critical behavior that is more accessible to the public.

With consideration of the societal-scale issue of ethical HAV behavior, criticality metric research could be applied to the consequentialist approach of ethics [3]. In this leading approach on implementable HAV ethical frameworks, ethical choices are determined by

assigning ethical costs to decisions. Criticality metrics, which reflect passenger safety, could be included in the cost calculation for the ethical decision.

Research of criticality metrics could be further improved with social-domain consideration by investigating their use as a communication tool. Criticality metrics could be used by lawmakers for writing measurable and explicit safety-critical standards for HAVs without requiring detailed knowledge of the HAV pathing and decision-making implementation. Additionally, criticality metrics are defined in terms of familiar traffic concepts and could apply to generalized traffic scenarios, reducing the amount of specific legislation for more concrete situations.

If the parameters of criticality metrics implemented in HAVs are explored with respect to the diverse driving preferences of different demographic groups, traffic throughput and comfort can be optimized. For example, criticality metric preferences between urban and rural populations could differ if urban users are willing to trade the higher criticality of a lower Post Encroachment Time for the higher traffic throughput of following vehicles more closely.

In conclusion, consideration of ethical, legal, and social issues associated with HAVs when researching criticality metrics would benefit both the automotive industry and the consumer. Grounded as a technical tool, criticality metrics can have applications as a tool for communication, promoting consumer trust and adoption of HAVs.

References

[1] Berboucha, Meriame. Uber Self-Driving Car Crash: What Really Happened. *Forbes: Science*. May 28, 2018. www.forbes.com/sites/meriameberboucha/2018/05/28/uber-self-driving-car-crash-what-really-happened/#44577af64dc4.

[2] Fleming, Charles. Tesla car mangled in fatal crash was on Autopilot and speeding, NTSB says. *Los Angeles Times*. July 26, 2016. www.latimes.com/business/autos/la-fi-hy-autopilot-photo-20160726-snap-story.html.

[3] Maurer, Markus, et al. *Autonomous driving*. Springer Berlin Heidelberg, Berlin, Germany, 2016.

[4] Van der Horst, Adrianus Rigardus Antonius. *A time based analysis of road user behaviour in normal and critical encounters*. No. HS-041 255. 1990.

[5] Chan, Ching-Yao. "Defining safety performance measures of driver-assistance systems for intersection left-turn conflicts." *Intelligent Vehicles Symposium, 2006 IEEE*. IEEE, 2006.

[6] Damm, W., et al. "Using traffic sequence charts for the development of HAVs." *Embedded Real Time Software and Systems-ERTS2018* (2018).

[7] Hillenbrand, Jrg, Andreas M. Spieker, and Kristian Kroschel. "A multilevel collision mitigation approach—Its situation assessment, decision making, and performance tradeoffs." *IEEE Transactions on intelligent transportation systems* 7.4 (2006): 528-540.

[8] Lefèvre, Stéphanie, Dizan Vasquez, and Christian Laugier. "A survey on motion prediction and risk assessment for intelligent vehicles." *Robomech Journal* 1.1 (2014): 1.