DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Formal Verification of Applied AI/ML based Autonomous Vehicles

Bachelor's Dissertation: ACPS LAB

Name | **Roll Number**

Niranjana R. Nair | 210003049

Thesis Supervisor:

Dr. Gourinath Banda

September 2, 2024

Contents

1 Introduction **2**

2 Preliminaries **3**

2.1 Autonomous Systems 3

2.1.1 Definition 3

2.1.2 Key Componenets 3

2.1.3 Levels of Autonomy 3

2.2 State of the art 4

2.3 Formal Verification 4

3 Errors in Autonomous Systems **5**

3.1 Taxonomy of Errors in OEDR Systems 5

3.2 Survey of Errors in Autonomous vehicles 6

3.3 Formal Verification Tools 7

4 Implementation **8**

4.1 Verification Tools 8

4.2 Simulators: IPG CarMaker 8

4.3 Results 8

5 Conclusion **9**

1 Introduction

Autonomous cars and aircraft are rapidly transitioning from futuristic concepts to tangible realities. These systems promise significant advancements in transportation by offering efficiency, safety, and convenience. However, as they become more prevalent, ensuring their reliable and safe operation is critical.

The backbone of these autonomous systems is artificial intelligence (AI), particularly machine learning (ML) algorithms, which enable them to make real-time decisions. However, while these AI-based technologies have shown remarkable capabilities, they also present challenges in terms of safety and correctness. The systems must not only avoid ”bad behaviors” that could lead to accidents or failures (safety) but also exhibit ”good behaviors” consistently to perform their intended functions effectively (liveness).

A significant concern is that many ML technologies currently lack formal verification, meaning their behavior cannot be rigorously proven to be safe and correct in all scenarios. This gap in verification introduces the risk of errors, which could have severe consequences in safety-critical applications like autonomous vehicles and aircraft.

This work aims to address these challenges by proposing a framework for understanding and categorizing errors in AI-driven autonomous systems. The core of this research is to develop a taxonomy of errors that occur in these systems, identifying which classes of errors can be formally verified. By systematically mapping these error classes, this work seeks to bridge the gap between the current unverified state of many ML technologies and the need for robust formal verification.

Recent accidents involving autonomous vehicles and aircraft underscore the urgency of this research. By analyzing these incidents through the lens of the proposed taxonomy, this study will demonstrate how certain errors align with specific classes that can potentially be formally verified. This approach provides a structured way to understand past failures and guides future efforts to prevent them.

The ultimate goal of this research is to identify those classes of errors in autonomous systems that are amenable to formal verification. By doing so, the work aims to contribute to the development of safer and more reliable AI-based technologies, ensuring that autonomous systems can be trusted to perform correctly under all conditions. This ambition aligns with the broader goal of advancing the field of AI and formal methods, pushing the boundaries of what can be formally verified in complex, real-world systems.

2 Preliminaries

2.1 Autonomous Systems

2.1.1 Definition

Autonomous systems are machines or software that can perform tasks or make decisions independently, without human intervention. These systems rely on advanced sensors, processors, and algorithms to perceive their environment, process information, and execute actions. They are employed in various domains, with autonomous vehicles (AVs) and aircraft being among the most prominent.

2.1.2 Key Componenets

Sensors: These include cameras, LiDAR, radar, GPS, and ultrasonic sensors, which provide the system with data about its surroundings.

Perception: The process by which the system interprets sensor data to understand the environment. This includes recognizing objects, understanding road conditions, and predicting the behavior of other entities (e.g., vehicles, pedestrians).

Decision-Making: Based on the perception, the system decides the next course of action, such as steering, accelerating, braking, or changing lanes.

Control: Executing the decision by sending commands to the vehicle's actuators, which control movement.

Communication: Autonomous systems often communicate with external entities, such as other vehicles, infrastructure, or cloud services, to enhance their functionality.

2.1.3 Levels of Autonomy

Autonomous vehicle systems are classified into six levels by the SAE, ranging from Level 0 (no automation) to Level 5 (full automation):

- Level 0 (No Automation): The driver controls everything, with only basic alerts or warnings from the vehicle.
- Level 1 (Driver Assistance): The vehicle can assist with either steering or speed control, but the driver remains fully engaged.
- Level 2 (Partial Automation): The vehicle can control both steering and speed simultaneously under specific conditions, but the driver must monitor the environment and take over if needed.
- Level 3 (Conditional Automation): The vehicle can handle all driving tasks in certain situations, but the driver must be ready to take control when the system requests.
- Level 4 (High Automation): The vehicle can operate autonomously within specific conditions or areas (geofencing), without requiring driver intervention, but might need human control outside these conditions.
- Level 5 (Full Automation): The vehicle is fully autonomous and can handle all driving tasks in any environment or situation without any human intervention.

As automation levels increase, the need for human involvement decreases, shifting responsibility from the driver to the vehicle's systems.

2.2 State of the art

As the automotive industry expands, many companies have started to shift their focus to autonomous vehicles.

- Tesla and General Motors have already launched their autonomous driving technology to the general masses. GM cruise control and Tesla autopilot are already popular among the masses. Tesla is currently working towards Full Self Driving (FSD) systems that can be implemented in their vehicles.

- Google's Waymo has successfully started operating in several cities in the United States with level 4 automation. They use advanced machine learning algorithms to mimic human driving behaviour.

- Automotive giants such as Volvo have introduced Pilot Assist systems in their vehicles. These systems are equipped with services such as collision avoidance, emergency braking, and so on, and they are constantly improved using machine learning methodologies.

2.3 Formal Verification

Autonomous systems must make complex decisions in real time, often in unpredictable environments. While ML algorithms can learn from data and improve over time, they can also behave unpredictably or make errors in rare or unseen scenarios. Formal Verification is a method used to prove the correctness of systems through mathematical and logical reasoning. In the context of autonomous systems and machine learning, formal verification can be used to ensure that the algorithms and models operate reliably and safely under all conditions. This is particularly important in safety-critical applications like autonomous driving.

3 Errors in Autonomous Systems

3.1 Taxonomy of Errors in OEDR Systems

These types of errors represent the most significant risks in autonomous vehicle operation. Each of these errors can contribute to incidents or accidents, depending on the severity and context in which they occur:

- False Positives: These errors are critical in determining the vehicle's ability to identify and react to its environment correctly. In mathematical terms,

  $$P(\mathrm{FP}) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(\hat{Y}_i = 1 \land Y_i = 0)$$

  Where:

  - $\hat{Y}_i$ is the prediction of the DNN for the ith input.
  - Y_i is the true label.
  - N is the total number of predictions.
  - $\mathbf{1}(\cdot)$ is the indicator function, which equals 1 if the condition is true and 0 otherwise.

- False Negatives: While false positives can lead to unnecessary actions, false negatives are more dangerous as they represent missed threats. In mathematical terms,

  $$P(\mathrm{FN}) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(\hat{Y}_i = 0 \land Y_i = 1)$$

  Where $P(\mathrm{FN})$ is the proportion of instances where the DNN predicts no threat ($\hat{Y}_i = 0$) when there is an actual threat ($Y_i = 1$).

- Failing Unspecified Edge Cases: These scenarios test the limits of the system's design and highlight the importance of comprehensive testing and robustness in real-world conditions.

  For a DNN, failing unspecified edge cases occurs when the network encounters inputs that fall outside the training data distribution, leading to unpredictable or erroneous outputs.

  Mathematical Representation:

  $$P(\mathrm{UEC}) = P(X \notin \mathrm{ODD}) \times P(\hat{Y} \mid X \notin \mathrm{ODD})$$

  Where:

  - X represents input features.
  - ODD (Operational Design Domain) is the distribution of data the DNN was trained on.
  - $P(\mathrm{UEC})$ represents the probability that the DNN fails when encountering an edge case.

- Misclassification: This can cause the vehicle to make decisions based on incorrect assumptions, leading to inappropriate actions. Mathematically speaking,

  $$P(\mathrm{MC}) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(\hat{Y}_i \neq Y_i)$$

  Where $P(\mathrm{MC})$ is the proportion of incorrect classifications by the DNN.

- Sensor Failure: This compromises the vehicle's perception, leading to potential safety risks due to the inability to detect and respond to the environment accurately.

  $$P(\mathrm{SF}) = 1 - \prod_{j=1}^{n} P(S_j)$$

  Where:

  - S_j represents the jth sensor input to the DNN.
  - n is the total number of sensor inputs.
  - $P(\mathrm{SF})$ represents the probability of a sensor failure affecting the DNN's input.

- Processing Delay: Even small delays can have significant impacts, especially in high-speed or complex driving environments, where real-time decision-making is crucial. Processing delay in a DNN context refers to the time taken by the network to produce an output after receiving an input.

  $$P(\mathrm{PD} > T_{\mathrm{threshold}}) = P(T_{\mathrm{compute}} > T_{\mathrm{threshold}})$$

  Where:

  - T_{compute} is the time taken by the DNN to process an input and generate an output.
  - $T_{\mathrm{threshold}}$ is the maximum allowable time for processing.
  - $P(\mathrm{PD})$ represents the probability that the processing time exceeds the acceptable threshold.

- Lack of Fallback: The absence of a fallback strategy in case of failure is a critical issue, as it can lead to situations where the system is unable to safely mitigate errors.

  $$P(\mathrm{LF}) = P(\mathrm{Error}) \times P(\mathrm{No\ Fallback} \mid \mathrm{Error})$$

  Where:

  - $P(\mathrm{Error})$ is the probability of the DNN making an error (e.g., misclassification, FP, FN).
  - $P(\mathrm{No\ Fallback} \mid \mathrm{Error})$ is the probability that no fallback mechanism is triggered after an error.

Note that these errors can overlap, and most fatal crashes involving autonomous vehicles involve more than one of these major errors.

3.2 Survey of Errors in Autonomous vehicles

- Death of Elaine Herzberg, Tempe, Arizona (March 18, 2018):

  Incident: A pedestrian with a bicycle was jaywalking across the road when the OEDR vehicle repeatedly misclassified the person, oscillating between identifying them as an unidentified object and a vehicle. The system assigned different trajectories to each classification until it was too late to avoid the collision. The Autonomous Driving System (ADS) shut down, but the human operator, who was distracted, failed to perform a fallback manoeuvre, resulting in a fatality.

  Errors: Failing Unspecified Edge Case (3) Misclassification (4) Lack of Fallback (7)

- Tesla ADAS Crashes

- Williston, Florida, USA (May 7, 2016):
  Incident: A Tesla in Autopilot mode crashed into an 18-wheeler truck painted bright white. The ADAS system failed to recognize the white truck against a brightly lit sky.
  Errors: False Negative (2) Failing Unspecified Edge Case (3) Misclassification (4) Lack of Fallback (7)

- Mountain View, California, USA (March 23, 2018):
  Incident: A Tesla crashed directly into a concrete barrier that its sensors could not recognize.
  Errors: Failing Unspecified Edge Case (3) Lack of Fallback (7)

- Kanagawa, Japan (April 29, 2018):
  Incident: A Tesla crashed into pedestrians and motorcycles after the vehicle in front of it changed lanes.
  Errors: Failing Unspecified Edge Case (3) Lack of Fallback (7)

- Delray Beach, Florida, USA (March 1, 2019):
  Incident: An incident identical to the Florida 2016 crash occurred, where a Tesla in Autopilot failed to recognize a white truck against a bright sky.
  Errors: False Negative (2) Failing Unspecified Edge Case (3) Misclassification (4) Lack of Fallback (7)

- Key Largo, Florida, USA (April 25, 2019):
  Incident: A Tesla failed to detect flashing red lights and stop signs, causing it to hit another car, which then struck two pedestrians.
  Errors: Misclassification (4) Sensor Failure (5?) — If sensor malfunction led to the failure to detect the lights and stop signs.

- Opal, Virginia, USA (July 19, 2023):
  Incident: A Tesla crashed into the underside of a truck.
  Errors: Processing Delay (6) Lack of Fallback (7)

- Snohomish County, Washington, USA (April 19, 2024):
  Incident: A Tesla in Full Self-Driving (FSD) mode failed to slow down when a motorcyclist ahead did.
  Errors: To be categorized (potentially related to sensor failure, processing delay, or lack of fallback).

3.3 Formal Verification Tools

To address these errors, it is essential to employ rigorous verification techniques that can prove the correctness and safety of autonomous systems.

- Formal Methods: Introducing formal methods such as model checking, theorem proving, and formal specification languages that are used to verify the correctness of autonomous systems.
- Verification Tools: A survey of state-of-the-art tools like SPIN, UPPAAL, and TLA+ that are used to formally verify the safety and reliability of AV systems.
- Case Studies: Examples of how these tools have been applied to detect and mitigate errors in real-world autonomous vehicle systems.

By exploring these tools and methods, we aim to enhance the safety and reliability of autonomous vehicles, ensuring that they can handle real-world challenges with minimal errors.

4 Implementation

4.1 Verification Tools

4.2 Simulators: IPG CarMaker

4.3 Results

5 Conclusion

Me when skibidi

Here is a citation: [1]

References

[1] Albert Einstein. “Zur Elektrodynamik bewegter Körper. (German) [On the electrodynamics of moving bodies]”. In: *Annalen der Physik* 322.10 (1905), pp. 891–921. DOI: `http://dx.doi.org/10.1002/andp.19053221004`.