Generating and Characterizing Scenarios for Safety Testing of Autonomous Vehicles

Zahra Ghodsi[1], Siva Kumar Sastry Hari[2], Iuri Frosio[2], Timothy Tsai[2], Alejandro Troccoli[2],
Stephen W. Keckler[2], Siddharth Garg[1], and Aimashree Anandkumar[2]
[1]New York University, [2]NVIDIA Corporation

***Abstract*—Extracting interesting scenarios from real world data as well as generating failure cases is important for the development and testing of autonomous systems. We propose efficient mechanisms to both characterize and generate testing scenarios using a state-of-the-art driving simulator. For any scenario, our method generates a set of possible driving paths and identifies all the possible safe driving trajectories that can be taken starting at different times, to compute metrics that quantify the complexity of the scenario. We test our method to characterize real driving data from the Next Generation Simulation (NGSIM) project, as well as adversarial scenarios generated in simulation. We rank the scenarios by defining metrics based on complexity of avoiding accidents and provide insights into how the AV could have minimized the probability of incurring an accident. We demonstrate a strong correlation between the proposed metrics and human intuition.**

I. Introduction

Research and development of Autonomous Vehicles (AV) has surged in recent years with many companies starting on-the-road testing, thanks to advances in perception and planning. However, comprehensive end-to-end testing on the road is prohibitively time consuming because unsafe conditions occur rarely. Significant advancements have been made in developing driving simulators that provide a scalable and safe environment for testing AVs. Existing methods to generate scenarios for safety testing consider a scenario useful if it involves a collision or a near collision [3], but a more detailed analysis reveals that not all scenarios that end in an accident are indeed useful, because the AV may not be able to avoid the accident given processing time limitations and physical constraints. Therefore, scenarios should be characterized based on the level of complexity (a synonym for possibility of avoiding collision) such that the scenarios can be prioritized for AV testing, learning methods, or other applications.

Our contributions in this work are twofold. First, we develop a method for characterizing driving scenarios used for safety testing of AVs. To the best of our knowledge, characterizing unsafe scenarios based on the avoidability of the accident has never been explored. Our characterization defines 5 metrics based on several factors such as the number of safe driving paths and total paths in the scenario, narrowness of safe paths, and the effort required to follow each safe path. We enumerate and store the set of paths using a computationally efficient tensor representation, and calculate the above metrics in the tensor space. We show that our characterization can extract interesting scenarios from real driving data [1]. Second, we develop a method for generating unsafe scenarios which takes an initial executed real or simulated staring condition as input, and introduces perturbations to increase the likelihood of an unsafe condition. Our method models dynamic actors who act as attackers for a fixed time aiming to create an unsafe condition by reducing their distance to the AV. Our method can generate approximately 240 scenarios per hour on a single system with up to 80 accidents, with more than 90% of them avoidable when appropriate actions are taken by an ideal AV system 2 seconds before the collision.

II. Methods to Characterize and Generate Scenarios

A *driving scenario* includes the description of the environment, the initial states of all actors at time t_0, and the driving policies of all actors except for the AV. Such a definition allows using a scenario to test or benchmark different AV policies. We use the term *sequence* to indicate a temporal succession of states of all the vehicles. A sequence can be obtained by executing a scenario on a simulator using a specific AV policy or obtained directly from real data.

Starting from time t_0, we compute a set of possible AV positions (EGO vehicle w_E in Fig. 1(a)) in time t_{i+1} by calculating the annulus sector (light blue in Fig. 1(a)) with all the locations of the center of the AV, based on the maximum acceleration (positive or negative) and steering angle of the AV in its state at t_i. We discretize the 2D sector with a grid with 0.5m spacing between the cells. Cells with centers inside the annulus are valid next states while the rest (off-road or collisions with other actors) are discarded. For each new valid state, we iterate the procedure until the end of the scenario at t_{end}.

To compute all safe paths, a natural approach would be to build a tree with all valid (safe) locations of w_E, and safe trajectories would correspond to paths from the root to the leaves of the tree. We implemented this approach and found it to have limited scalability, high storage needs, and challenging parallelization. To overcome these issues, we proposed a tensor based approach which quantizes space, speed, and the orientation of the AV. In this representation, a 4D tensor keeps track of all possible states for the AV at t_i, as shown in Fig. 1(b). For each valid state (marked in green) at t_i, the possible states at t_{i+1} are computed as explained before, which requires checking for road boundary and collision with other actors. To speed up this computation, we pre-compute a map matrix for the scenario and a collision tensor for each t_i in the

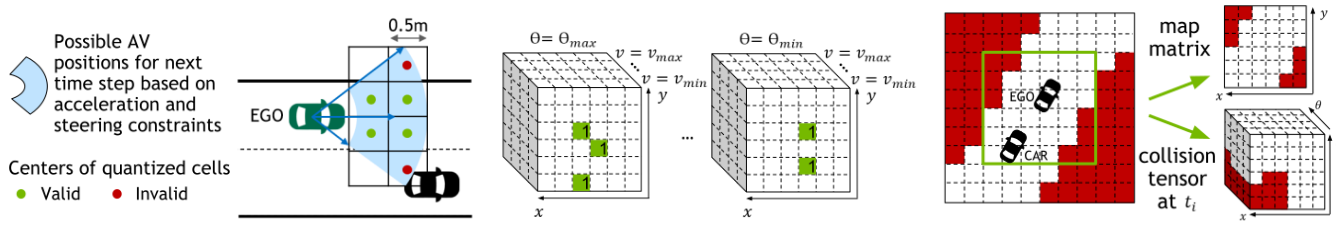

Fig. 1: Scenario scoring based on search for safe driving policy. (a) shows the quantized region on the map where the AV can travel in the next time step based on maximum allowed steering and acceleration, (b) is tensor representation of AV state at time t_i during the scenario roll-out, and (c) depicts pre-calculation of map matrix and collision tensor from a scenario.

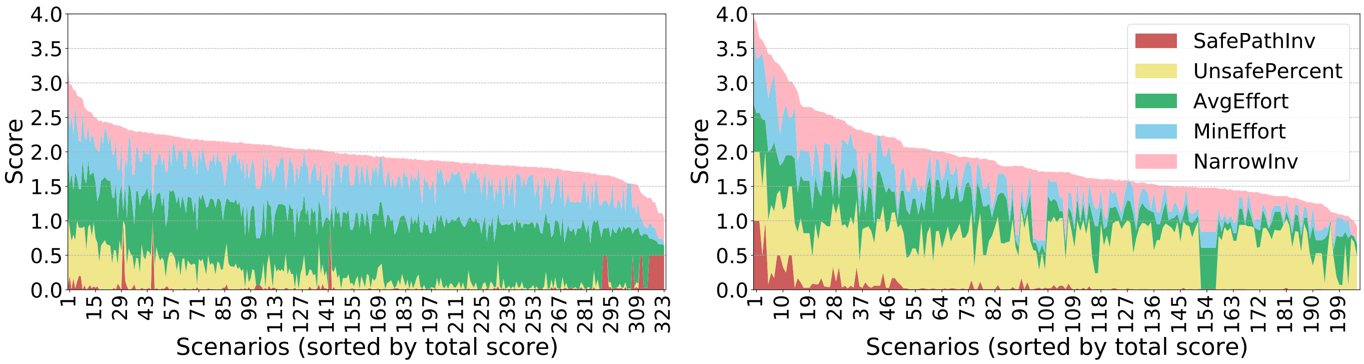

Fig. 2: Score and metric break down of (a) scenarios from NGSIM, and (b) generated adversarial scenarios with accidents.

scenario as shown in Fig. 1(c) to store invalid states for the AV. The final tensor at t_{end} gives the number of safe paths. Similarly, we use propagation in the tensor space to compute total on-road paths. We define the metrics below, with the first two metrics computed directly from the number of safe paths and the number of on-road paths, and the rest computed using a similar propagation in tensor space:

- **SafePathInv** ($1/\#p$)**:** Inverse of the number of safe paths ($\#p$) available to the AV until the end of the scenario.
- **UnsafePercent** ($p_c\%$)**:** The percentage of paths leading to a collision within a given scenario, among all the paths that do not lead the AV off-road, in absence of other vehicles.
- **AvgEffort** ($E[e_s]$)**:** We compute the effort to navigate from a parent to a child cell as the sum of the absolute values of the steering and acceleration controls. To compute the effort needed to navigate a safe path (e_s), we accumulate cell effort values. We compute the average effort for all the safe paths to obtain AvgEffort ($E[e_s]$).
- **MinEffort** ($\min[e_s]$)**:** The minimum effort required to navigate through a safe path.
- **NarrowInv** ($1/E[\min[c_s]]$)**:** We measure how narrow a safe path is by computing the minimum number of children for each of the nodes in a safe path (s). We refer to it as $\min[c_s]$, where c_s indicates the number of children of a node along the path. $E[\min[c_s]]$ is the average of the minimum number of children each safe path; its inverse (referred to as NarrowInv) is one (or close to one) when the AV has only one (or few) options at each node to navigate through a safe path. A narrow path suggests that an evasive maneuver requires precision during execution.

These metrics are defined such that a higher metric value indicates a more challenging scenario. Using this set of metrics, we can characterize scenarios either extracted from real world driving datasets, or synthetically generated for testing. Creating metrics for scenario characterization allows us to compute a difficulty score for scenarios in testing, as well as to measure their diversity for creating non-trivial datasets. The definition of a distance function, e.g., based on the characterization metrics introduced in our paper, is left for future investigation.

Using this set of metrics, we characterize scenarios extracted from real world driving data in NGSIM [1]; to do so we extract 3s scenarios from Interstate 80 Freeway and pick each time a random vehicle as ego.

We also develop a fast and scalable method to generate adversarial scenarios in NVIDIA DRIVE Sim [2] simulation platform. A *base sequence* is simulated by spawning an AV and other vehicles on the map at random initial states (positions and velocities), and assigning to each vehicle (including the AV) a simple driving policy described by a Markov chain where vehicles change lane or speed with probability p after every N time steps, and otherwise drive on the same lane.

To generate an *unsafe sequence* from the *base sequence*, we change the policy of a vehicle that is closest to the AV. For a subsequent period (randomly selected between 3 to 5 seconds), the selected vehicle violates safety and sets its steering and acceleration to decrease the distance to the AV and increase the chance of a collision (e.g. simulating a distracted driver). The driving policy for this vehicle during this time is determined by computing acceleration and steering such that the distance to the AV decreases. To ensure that the generated adversarial scenarios are more realistic, we can seed this generator from real driving scenarios. Starting from a real scenario, we can modify the behavior of one driver for 3 seconds to generate an incident. It is worth noting that even if this may be a pessimistic scenario, it is still acceptable for benchmarking, as it can compare the performance of different AV systems without requiring that all scenarios must be safely navigated.

III. Results

We compute the five characterization metrics for the extracted and generated scenarios. We also assign a *score* as the sum of the metrics, after normalization based on the range of values over all scenarios. The break down of our metrics for NGSIM and generated scenarios is depicted in Fig. 2, sorting scenarios by the score. The generated adversarial scenarios have higher scores compared to real driving data, with higher UnsafePercent for all scenarios. This is reasonable due to the presence of the attacking vehicle. NGSIM contains crowded highway data, which explains the larger AvgEffort and MinEffort compared to generated scenarios with sparse actors. For adversarial scenarios, SafePathInv is large only for the most challenging scenarios (1-17 in the graph) which have a small number of safe paths. For these scenarios, MinEffort and AvgEffort is consistently high, indicating the effort required to navigate through the available safe paths is high. NarrowInv is also high for the most challenging scenarios (1-73), suggesting that in at least one node in the available safe paths, the AV has to perform a unique action to avoid the accident.

These results demonstrate that our characterization method is effective in identifying the few challenging driving scenarios from a large set of real or simulated scenarios, which opens the door to efficient, large scale AV testing.

IV. Related Work

There has been some progress in generating unsafe driving scenarios, fueled by recent advancements in the driving simulators and real-traffic data-sets. (1) In [5], a base distribution representing standard traffic behavior is learned from data. An adaptive importance sampling method is applied to learn alternative distributions from the base distribution, that can generate accidents more frequently. This method is limited to the road segments and types of scenarios present in the dataset, unlike our generation method. The scenarios are ranked based on their likelihood under the base distribution, without considering the avoidability of the accident, which is another key criteria for testing. (2) In [3], authors use Baysian Optimization to generate adversarial scenarios that increase the risk of collision with pedestrians and vehicles. This method scales poorly with the increasing number of actors. Our greedy and gradient-based approach to generating unsafe scenario performs much faster. (3) In [4], the authors model the problem of finding failure cases as a Markov decision process and use reinforcement learning to solve it. Methods based on reinforcement learning need long training time for each new configuration, which limits their applicability. We require no training to generate an unsafe scenario. None of these techniques characterize generated scenario based on accident avoidability. Our method can be applied to scenarios generated by any of the above techniques.

V. Conclusion

Recent years have witnessed great advances in autonomous vehicles (AVs). Systems used in autonomous vehicles (AVs) require rigorous testing to ensure safety. We develop a fast method to generate potentially unsafe scenarios that help us uncover shortcomings of an AV under test, resulting in the design a safer AV. We demonstrate that our method generates 240 potentially unsafe scenarios per hour with more than a third of the scenarios resulting in accidents on a state-of-the-art driving simulator. For the generated unsafe conditions, we propose a novel characterization method that quantifies how easily the unsafe condition can be avoided. We enumerate possible safe paths starting at different times before the accident and derive accident avoidability metrics. These metrics provide insights into how the AV could have avoided the accident, which is key to developing a safe AV.

References

[1] U.S. Federal Highway Administration. Next Generation Simulation (NGSIM). http://ops.fhwa.dot.gov/trafficanalysistools/ngsim.htm, 2006.

[2] NVIDIA DRIVE Sim. https://developer.nvidia.com/drive/drive-constellation, 2018.

[3] Yasasa Abeysirigoonawardena, Florian Shkurti, and Gregory Dudek. Generating adversarial driving scenarios in high-fidelity simulators. In *International Conference on Robotics and Automation (ICRA)*, 2019.

[4] Anthony Corso, Peter Du, Katherine Driggs-Campbell, and Mykel J Kochenderfer. Adaptive stress testing with reward augmentation for autonomous vehicle validatio. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 163–168. IEEE, 2019.

[5] Matthew O'Kelly, Aman Sinha, Hongseok Namkoong, Russ Tedrake, and John C Duchi. Scalable end-to-end autonomous vehicle testing via rare-event simulation. In *Advances in Neural Information Processing Systems*, 2018.