

LimSim: A Long-term Interactive Multi-scenario Traffic Simulator

Licheng Wen^{1,†}, Daocheng Fu^{1,†}, Song Mao¹, Pinlong Cai^{1,*}, Min Dou¹, Yikang Li¹ and Yu Qiao¹

Abstract—With the growing popularity of digital twin and autonomous driving in transportation, the demand for simulation systems capable of generating high-fidelity and reliable scenarios is increasing. Existing simulation systems suffer from a lack of support for different types of scenarios, and the vehicle models used in these systems are too simplistic. Thus, such systems fail to represent driving styles and multi-vehicle interactions, and struggle to handle corner cases in the dataset. In this paper, we propose LimSim, the Long-term Interactive Multi-scenario traffic Simulator, which aims to provide a long-term continuous simulation capability under the urban road network. LimSim can simulate fine-grained dynamic scenarios and focus on the diverse interactions between multiple vehicles in the traffic flow. This paper provides a detailed introduction to the framework and features of the LimSim, and demonstrates its performance through case studies and experiments. The source code of LimSim is available on GitHub: github.com/PJLab-ADG/LimSim.

I. INTRODUCTION

The urban traffic system is complex and diverse, encompassing numerous types of roads, such as intersections, ramps, and roundabouts, as well as a wide variety of traffic participants. Vehicles are the principal component of road scenarios, and drivers with different driving styles exhibit heterogeneous behavior while driving [1], [2]. By providing a comprehensive view of how urban transport systems operate, traffic simulators offer policymakers valuable insights into the potential real-world impacts of their decisions [3]–[5]. Simulators also allow potential problems to be identified and addressed before implementation in the physical environment, resulting in improved transportation while saving time and resources. With the wide use of advanced technologies like virtual reality and data mining, simulation focusing on driving scenarios has gained significant attention in recent years and is becoming a crucial element in the construction of the digital twins in cities [6], [7]. Scenario-oriented simulation also plays a vital role in the development and testing of autonomous driving. It enables testing and validation of multiple algorithms and systems by conducting simulations in different scenarios, environments, system configurations, and driving characteristics [8].

Generally, simulators can be classified into three kinds, including flow-based simulators (e.g., PARAMICS [9], Vissim [10], Aimsun [11], SUMO [12]), vehicle-based simulators (e.g., AirSim [13], LGSVL [14], CARLA [15], MetaDrive

¹ All authors are with the Shanghai Artificial Intelligence Laboratory, Shanghai, China, 200032, Email: {wenlicheng, fudaocheng, maosong, caipinlong, doumin, liyikang, qiaoyu}@pjlab.org.cn.

[†] Equal contribution; Order determined by a coin toss.

* Corresponding author.

[16]), and data-based simulators (e.g., SimNet [17], InterSim [18], TrafficGen [19]). While flow-based simulators can reflect the characteristics of the real-world transportation system and analyze various traffic flow conditions, they tend to simplify vehicle motion behaviors and fail to consider multi-vehicle interactions and kinematics constraints. The vehicle-based simulators aim to construct virtual road scenarios to verify the performances of developed algorithms. However, the generation of background traffic flow relies on manual editing scenarios or using collected road data, which fails to accurately reproduce the characteristics of actual scenarios. In addition, the data-based simulators rely on real-world traffic data. By extracting vehicle motion features from the datasets, these simulators allow ego vehicle to interact with the background traffic flow. However, the datasets typically provide fragmented, small-scale scenarios, making these simulators unable to conduct long-term continuous simulations.

To overcome these limitations, we propose LimSim, the Long-term Interactive Multi-scenario Traffic Simulator. LimSim simulates dynamic traffic scenarios in urban road networks at a fine-grained level. It includes a multi-scenario road network construction module, a multi-source traffic flow generation module, a multi-vehicle joint decision planning module, and a multi-dimensional scenario analysis module. The core features of LimSim are listed below:

- **Long-term:** Traffic flow can be generated over long periods under the guidance of demand construction and route planning. Both geometric information and topological structure are essential inputs to represent a city-level road network, as shown in Fig. 1(a).
- **Diversity:** The built-in behavioral models take heterogeneous social value orientations and driving styles of vehicles into account, supporting various driving behaviors like car-following, lane-changing, and merging, as shown in Fig. 1(c.1) and 1(c.2).
- **Interactivity:** Vehicles in the scenario area are controlled by a joint decision-making and planning framework, offering sophisticated interactions among vehicles, as shown in Fig. 1(b) and 1(d).
- **Multi-scenario:** The universal road components are proposed to support a variety of road structures in the real world. These components integrate macroscopic and microscopic scenarios and thus ensure a consistent simulation system, as shown in Fig. 1(e).

II. RELATED WORK

A. Flow-based simulators

The development of the flow-based simulating systems has been going on for decades, intending to provide tech-

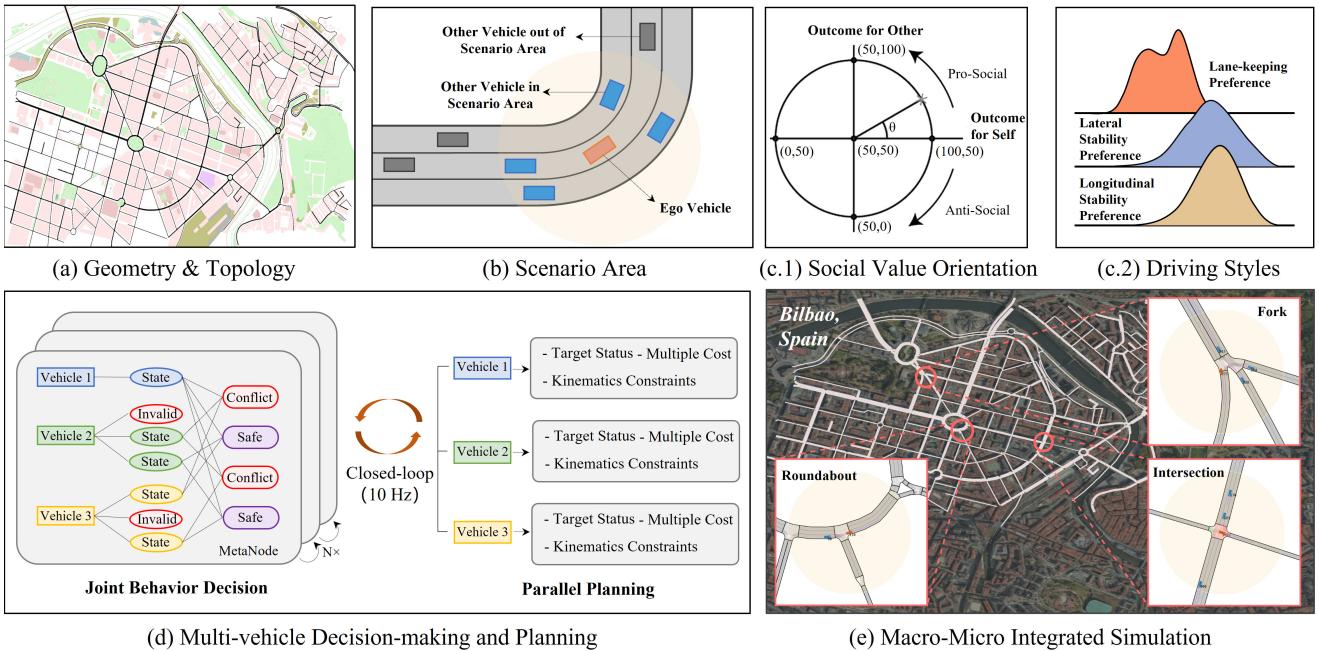


Fig. 1. (a) LimSim supports the representation of both geometric information and topological structure of road networks derived from vector maps. (b) The range of microscopic simulation is defined by the scenario area. Vehicles outside this area are controlled by classic traffic flow models, while those within the range undergo precise decision-making and planning processes. (c.1) The social value orientations quantify the balance of interests in the process of multi-vehicle interaction. (c.2) Heterogeneous driving behaviors are taken into account during trajectory planning. (d) Multi-vehicle joint behavior decision-making allows all vehicles with potential interactions to display mature driving maneuvers. High-frequency parallel trajectory planning considers vehicle's kinematic constraints and ensures runtime efficiency. (e) LimSim combines macroscopic traffic flow simulations for the entire road network with microscopic vehicle motion simulations in specific scenarios, providing a comprehensive representation of traffic behaviour.

nical support for urban planners and managers. PARAMICS (Parallel Microscopic Simulator) was released as commercial software by Quadstone in 1998, which integrates simulation, visualization, interactive road network drawing, adaptive signal control, online simulation data statistical analysis, and traffic control strategy evaluation, among other features [9]. Vissim is a commercial microscopic traffic simulation software developed by PTV GmbH, providing high-level visualization of complex traffic situations with realistic traffic models [10]. CORSIM (CORridor SIMulation) is a traffic engineering analysis and modeling software developed with support from the Federal Highway Administration (FHWA), which can perform complex road geometry modeling, large-scale traffic condition simulation, traffic control, and management simulation, road network topology and vehicle interaction [20]. Aimsun is widely used by traffic engineers for traffic planning, microscopic traffic simulation, traffic demand, and related data analysis [11]. SUMO (Simulation of Urban MObility) is a free and open-source traffic flow simulation software developed by the German Aerospace Center (DLR) for modeling intermodal traffic systems, including various modes of transportation, and can perform traffic network construction, route planning, emission calculations, and other functions, with multiple APIs to control the simulation remotely [12].

While flow-based simulators are powerful in facilitating large-scale traffic simulations to demonstrate the overall traffic conditions of the road network, they generally rely on simplistic car-following and lane-changing models to

control vehicles, resulting in limited vehicle behavior simulations. Moreover, due to the lack of strict vehicle kinematic constraints, flow-based simulators cannot accurately capture the microscopic movements of vehicles, thus reducing their simulation fidelity.

B. Vehicle-based simulators

Vehicle-based simulation can also provide realistic dynamic simulations [21]. Early autonomous driving software used modified commercial game engines for physical simulation and visualization, providing fundamental physical assets and interaction between objects [22], including USARSim [23], Webots [24], SimRobot [25], Unity [26], etc. These autonomous driving simulation engine requires compatible open interfaces, obtainable object, and sensor parameters, realistic graphics rendering and physical engine, and deployment of cost efficiency [27]. The following simulators are very popular nowadays.

Gazebo is an open-source, scalable, and flexible multi-robot 3D simulator, which is usually used along with ROS to simulate the physical characteristics of mock objects and realize dynamic 3D rendering and interactive communication [28]. AirSim is an Unreal Engine-based simulator that provides physically and visually realistic simulations for a variety of purposes, which includes vehicle simulations, city scenarios, APIs to simplify programming, and plug-and-play code to quickly create rich scenarios [13]. LGSVL is an open-source autonomous driving simulator developed by LG's Silicon Valley laboratory based on the Unity engine, which permits users to label and export for a high-precision

TABLE I
COMPARISONS OF EXISTING SIMULATORS

Simulator	Long Term	Vehicle Dynamic	Diverse Interaction	Custom Trajectory	Cost Efficiency	Transferability
Flow-based	+	—	○	+	○	+
Vehicle-based	○	+	—	○	—	+
Data-based	○	○	○	—	○	—
LimSim	+	+	+	+	+	+
— Not Available	○ Basic Performance	—	+ Good Performance	—	—	—

map format and also provides support for multi-type sensor simulation including LiDAR, millimeter wave radar, GPS, IMU, and camera [14]. CARLA is an open-source simulator for autonomous driving research, which provides open digital assets (including urban layouts, buildings, vehicles, etc.) and supports flexible specifications of sensor suites and environmental conditions [15]. MetaDrive is an easy-to-install and powerful driving simulation platform that is used to explore universal reinforcement learning algorithms to support machine autonomy. It generates an infinite number of driving scenarios from generating programs and importing real data [16].

The vehicle-based simulators have more precise kinematic constraints for trajectory planning, providing a more realistic driving environment for testing autonomous driving decision and planning algorithms. However, this type of simulations lacks interactive modeling of vehicles, and cannot provide background traffic flow with real scenario characteristics.

C. Data-based simulators

Simulators that use traffic flow data appeals to many researchers. During the simulation process, the ego vehicles will take different behaviors from the dataset, making it difficult to consider the dynamic interaction between ego vehicles and other vehicles in the road environments, since other vehicles need to respond the behavior of ego vehicle. This type of work is called data-based simulators, and some of the latest work are summarized as follows.

SimNet is the first to present a machine learning-based simulation system for ego vehicles that generates realistic driving episodes based on historical driving data [17]. The system's performance improves with the amount of data used for training and can help identify issues in planning systems. The approach is more scalable than hand-coding actors and can achieve high realism by directly learning behavior from observed data. InterSim is an interactive traffic simulator used to test autonomous driving planners, which infers the interaction relationships between vehicles in the scenarios and generates realistic trajectories consistent with these relationships for environmental vehicles [18]. TrafficGen is also a data-driven method for generating realistic traffic scenarios. It uses an autoregressive generative model with an encoder-decoder architecture to synthesize complete traffic scenarios. Specifically, TrafficGen places vehicles on a given map and uses a motion forecasting model to generate multi-mode diverse samples for vehicles' trajectories [19].

Data-based simulators can implicitly learn multi-vehicle interactions from natural driving environments, and simulate how the vehicles interact with each other. However, these

simulations rely heavily on the collected dataset, making it difficult to edit and expand the scenarios, resulting in the problem of scenario fragmentation.

Table I provides a comparison of LimSim and other types of simulation systems in terms of functional coverage. Compared with three existing kinds of simulators, LimSim is able to simulate scenarios finely and efficiently, with a balance between real-time and fidelity. It supports the generation of dynamic traffic scenario data and system testing for autonomous driving.

III. SYSTEM OVERVIEW

A. System framework

As shown in Fig. 2, LimSim includes four modules to simulate long-term traffic scenarios, and the detailed descriptions of these modules are given below.

a) *Multi-scenario road network construction:* LimSim defines a standardized road network representation based on vector maps, which encompasses geometric and topological information. LimSim can cover multiple types of road scenarios such as intersections, roundabouts, and ramps. Users can construct the road network using tools such as SUMO or generate it directly from sensor data and expert knowledge. For long-term simulation, the route planning is carried out based on the standardized representation of the road network to construct road-level paths and determine feasible lanes for vehicles on each road section of the route. The scenario area of LimSim supports both fixed scene mode and hero mode (moving scenes following the ego vehicles). Vehicles in the scenario area have finer motion control to create a more realistic traffic environment.

b) *Multi-source traffic flow generation:* The multi-source traffic flow generation module in LimSim supports several baseline microscopic traffic models, such as car-following, lane-changing, and merging motions. It also supports customized scenarios using naturalistic driving datasets and relevant standards. LimSim takes the continuity of traffic flow into account in both macroscopic and microscopic views, demonstrating compatibility and diversity while simulating real-world scenarios.

c) *Multi-vehicle joint decision-making and planning:* For controlled vehicles within the scenario area, LimSim adopts a hierarchical design that we call the *PDP process*, including three steps: Prediction, Decision-making, and Planning. Such a design not only enables control for sophisticated vehicle behaviors but also matches the prevailing autonomous driving technology stack. LimSim also adopts microscopic traffic models for predicting the vehicles outside

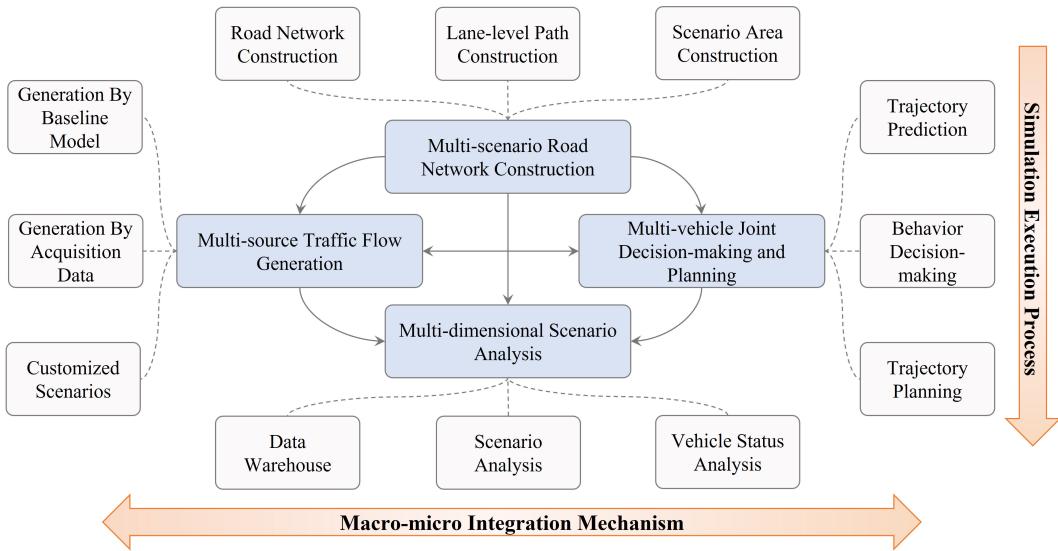


Fig. 2. Framework of LimSim. LimSim comprises a multi-scenario road network construction module, a multi-source traffic flow generation module, a multi-vehicle joint decision-making and planning module, and a multi-dimensional scenario analysis module.

the scenario area, acting as inputs for multi-vehicle joint decision-making and planning.

d) Multi-dimensional scenario analysis: LimSim captures information such as road network, vehicle properties, vehicle motion parameters, etc., throughout the simulation for scenario replay and data generation. It also provides online or offline analysis of the vehicle status during the simulation process, including indicators such as safety, comfort, energy consumption, and trajectory quality. The module supports a comprehensive evaluation of the simulation process including scenario complexity and risk.

The four modules of LimSim form an efficient and effective scenario-oriented simulation system that is used to test autonomous driving algorithms and facilitate transportation planning and policy-making. LimSim supports various driving behaviors and road scenarios while maintaining a realistic traffic environment and high data compatibility.

B. User preference

Users need to adapt the inputs of LimSim to complete traffic flow simulation tasks. According to the description of system modules, the input of LimSim must include the geometric dimensions of roads and the topological connections among them. Besides, traffic signal schemes or traffic sign information are optional inputs. LimSim supports vehicle control at signalized intersections and adjusts vehicle control strategies based on traffic sign information, such as adjusting the maximum vehicle speed based on road speed limits, or limiting whether vehicles are allowed to turn left or right based on traffic sign information. If only the road network information are provided, then LimSim can randomly generate demands and routes of vehicles for randomized simulation. We recommend the joint simulation of LimSim and SUMO, where necessary scenario information can be extracted from SUMO and relevant implementation scripts are provided. If users provide an instantaneous state of the

scenario, including the moving vehicles in the road network, LimSim can take over and generate the subsequent motion paths of these vehicles. More generally, users can offer a collected scenario sequence, and LimSim can generate many simulation scenarios based on the original sequence. We have implemented this functionality based on the Waymo motion dataset.

The output of LimSim includes real-time records of the simulation, which are available to users via the database, including road information, traffic control information, and vehicle trajectory information. Additionally, a visual user interface developed based on Dear PyGUi¹ (a fast and powerful graphical user interface toolkit for Python) provides simulation visualization from the bird's eye view.

LimSim's interface allows users to replace algorithms used in PDP process with their own. Specifically, ego and other vehicles in the scenario are distinguished, and users can use self-developed algorithms to control the motion of the ego vehicle and then test the solutions' performance in the simulations. The scenario analysis module provides evaluation information for each simulation. Users can also extract the trajectories of all vehicles from the database to calculate custom performance metrics. For other vehicles in the scenario area, users can generate diverse scenarios by modifying the strategy parameters of the other vehicles. The customized parameters include the weights of the cost functions in trajectory planning, the social value orientation of vehicles, etc.

IV. FEATURES

A. Long-term simulation and evaluation

One important features of LimSim is extracting key scenarios through long-term simulation and evaluation. Here we refer to key scenario as scenarios that provide a good

¹<https://github.com/hoffstadt/DearPyGui>

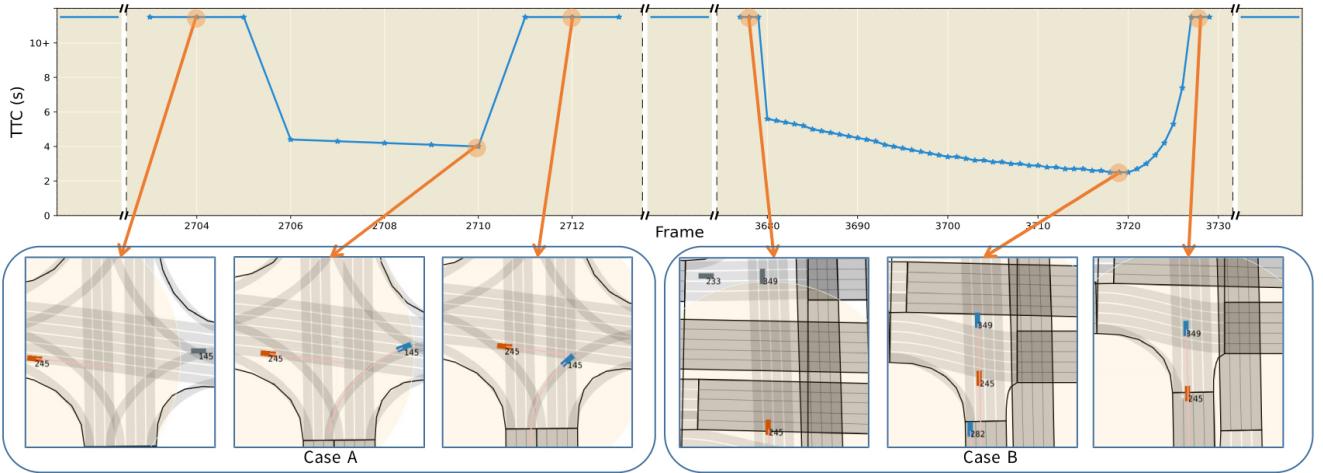


Fig. 3. Extracting scenarios from long-term evaluation results. Case A describes a potential collision when a vehicle takes an unprotected left turn; Case B describes a potential rear-end collision where the following vehicle finally brakes to prevent crash.

specification for autonomous driving systems under design. Classically, a large number of tests are required to estimate the performance of automated vehicles reliably due to rarity of key scenarios, which leads to high economic and time costs [29]. To improve the efficiency of testing and validation, extracting key scenarios from driving data is critical. Our strategy for extracting key scenarios is “One for all”, that is, running one episode of simulation with all kinds of scenarios contained in it. In LimSim, the city level map includes a range of complex scenarios such as roundabouts, intersections, motorways, etc. Different scenarios can be easily made available in testing phase. After setting up the environment, customized algorithms can be sufficiently tested during one simulation episode. When one episode of simulation is done, the key scenarios are extracted via evaluation results.

Most existing datasets lack context of driving scenarios due to simplicity of scenario extraction methods. For example, Lyft dataset is partitioned by a fixed time frequency [30], which leads to discontinuity or incompleteness of a scenario. LimSim fills the gap by performing long-term scenario evaluation. We adopt the Time-to-collision (TTC), which is defined as “The time required for two vehicles to collide if they continue at their present speed and on the same path” [31], as a measure of the severity of conflicts. Fig. 3 depicts extracting scenarios from the TTC curve. In this simulation, the TTCs are above 10 seconds most of the time, indicating that those scenarios are ordinary. However, there are two segments with relatively short TTCs, both occurring in more critical scenes: a vehicle making an unprotected left turn and an emergency stop to avoid a rear-end collision.

Normally, only a few frames will be extracted to a key scenario, which may cause incompleteness. Benefit from the continuity of long-term simulation, LimSim improves this strategy by looking backward and forward. On the one hand, by adding frames before scenario begins, the states of tested and other vehicles are clearer and easier to simulate; on the other hand, adding frames after the ending of key scenario makes it possible to find a feasible trajectory. Overall, the

extra frames are added to the corner cases to serve the improvement and deployment of automated vehicles better.

B. Multi-vehicle motion control

While popular microscopic traffic flow models partially simulate human driving habits, the social interaction behaviour between vehicles is still very homogeneous [32]. Nowhere is this more noticeable than when a vehicle is changing lanes and merging. These microscopic models do not take into account the vehicle’s kinematic constraints and therefore fail to ensure smoothness and continuity of vehicle motion. Thus, a more sophisticated control strategy is required for vehicles that have a direct impact on the ego vehicle, i.e. those in the scenario area.

We adopt the hierarchical PDP process to perform motion control for vehicles within the scenario area. LimSim comes with an implementation of the PDP process modified from our previous work [33] and outperforms traffic flow models in various metrics. LimSim also supports users’ own implementation of part or all of the algorithm in the PDP process, allowing them to test their algorithm performance, as shown in Fig. 4.

1) Decision-making: Joint decision-making for multiple vehicle using Monte-Carlo Tree Search, or MCTS, has proven to allow for social vehicle interaction and generate traffic flows with diversity [34]. However, when simulating a busy intersection or a congested motorway, the large number of vehicles in the scenario area causes the search tree to grow exponentially. This results in a dramatic decrease of the computation efficiency and the success rate of decision-making algorithm. Actually it’s neither efficient nor realistic to require each vehicle to pay attention to every other vehicle’s movements. In LimSim, we employ a group-based MCTS decision-making algorithm for multiple vehicles based on the idea of dividing the global vehicle interaction problem into a set of sub-problems with in each group.

Specifically, the decision-making procedure first calculates potential interactions between each two vehicles. For example, a cut-in vehicle and the vehicle behind it, and

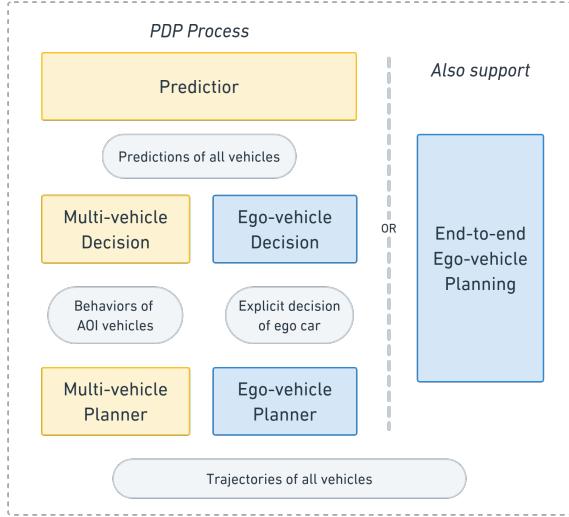


Fig. 4. The PDP process in LimSim. The yellow boxes indicate the implemented procedures of LimSim, including multi-vehicle prediction, decision making and planning. The blue boxes indicate user-implemented procedures, focusing on ego-vehicle control.

vehicles entering an intersection at the same time, both have some potential interactions. We divide all vehicles in the scenario area into several groups and make sure vehicles with potential interaction are assigned to the same group. An intra-group MCTS decision-making algorithm is then performed for each group in turn and combined into a joint decision result for all vehicles in scenario area. The decision results of the previous groups are known and taken into account by the subsequent groups.

LimSim allows vehicles with different Social Value Orientations (SVOs) during the decision-making procedure [35]. During MCTS, a vehicle considers not only the effect of the intended action on itself, i.e. reward to self, but also the risk of that action to other surrounding vehicles, i.e. penalty to others. Vehicles with different SVOs balance between these two factors in varying ways, resulting in radically divergent decision outcomes.

2) Planning: The planning algorithm with a parallel architecture fits well as a downstream procedure of the decision-making described above. The planner first attempts to generate a smooth and continuous path based on each vehicle's decision results. It also checks whether the path satisfies vehicle's kinematic constraints, including turning radius, speed/acceleration limits, etc. If the above attempts fail and a vehicle is unable to follow its decision, the planner switches to a larger target state space to generate a valid trajectory, such as aborting a lane change or emergency braking. Vehicles with different SVOs are also given different cost weights during the planning procedure, which further increases the diversity of vehicle trajectories.

3) Case study: We demonstrate the performance of our PDP process in an on-ramp merging scenario from CitySim dataset [36]. As shown in Fig. 5, the vehicle with ID 257 on the ramp acts as the ego vehicle, and the vehicle with ID 133 on the mainline competes with it for the right of way.

In case 1, the decision-making procedure is disabled. Due

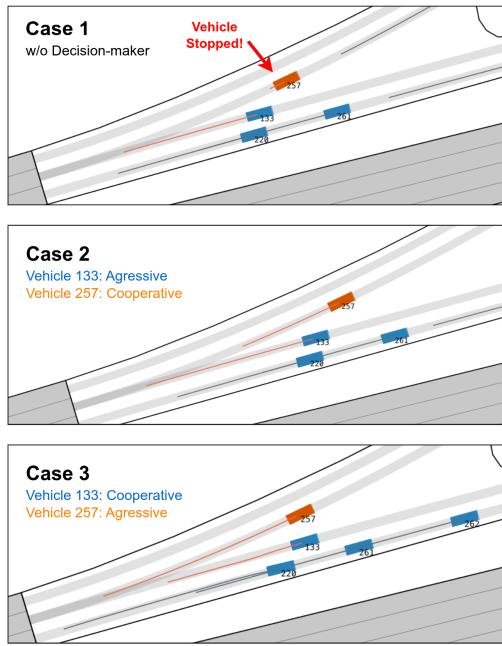


Fig. 5. A merging scenario in highway. The line in the front of each vehicle indicates its trajectory for the next 3 seconds. In case 1 the decision-maker is disabled, causing ego vehicle performing an emergency stop. And in case 2 and 3, vehicles possess different SVOs.

to a lack of foresighted long-term decisions, the ego vehicle missed the best time to merge onto mainline and thus had to brake urgently to avoid vehicle with ID 133. Instead, in cases 2 and 3, the decision procedure is activated so that the dangerous situation never occurs. In these two cases, vehicles possess different SVOs, where the aggressive style means that the vehicle is more selfish, and the cooperative style gives more consideration to other vehicles. As a result, the order of passage are changed.

C. Interactive scenario reconstruction

In a road environment, it would be very valuable for simulation if a large number of real scenarios could be collected and reconstructed. Currently, there are two approaches to using real-world scenarios: scenario replay and scenario generation [37]. In the approach of scenario replay, the simulation system provides decision-making and planning methods for the ego vehicle, while other vehicles in the scenario drive exactly as logged. Although easy to implement, the ego vehicle controlled by this method cannot interact with other vehicles and is limited to testing its responsiveness in very few scenarios. With regard to the scenario generation method, designed models are applied to plan trajectories for all vehicles in the log information from a certain moment until the end. Although it can achieve interaction between the ego vehicle and other vehicles, the subsequent vehicle movements may deviate completely from real scenarios due to insufficient model authenticity.

LimSim offers the interactive scenario reconstruction as one of its core features, making the simulation scenarios as close to the real scenarios as possible. In most cases, other vehicles in the simulation scenario will move according to

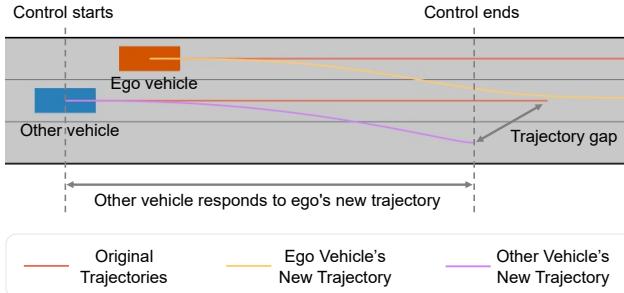


Fig. 6. Illustration of interactive scenario reconstruction: When the ego vehicle adopts a new trajectory that affects another vehicle, the other vehicle must respond. At this point, the other vehicle is controlled by the LimSim to make an interactive action. When the other vehicle moves away from the ego vehicle, the control ends and the trajectory gap is considered for the other vehicle to return to its original trajectory.

their original trajectory in the log data. However, if other vehicles have to adjust their trajectories due to the impact of the ego vehicle, they need to be controlled. To achieve this goal, three problems have to be solved. Firstly, how to identify which vehicles need to be controlled by the simulation system? The original trajectories will be judged to see whether there is any conflict with other controlled vehicles by checking the multiple trajectories in the next few seconds. Secondly, how to determine the control period of the controlled vehicles? Based on the time length of the trajectory planning process, the controlled vehicles will obtain trajectories for next few seconds. Once these vehicles reach the end points of the planned trajectories, they will be released from the control of the simulation system. Finally, after the vehicles are released from control, how to handle them? At the current time step, they will be destroyed if the vehicles do not have original trajectories in the log data. If the vehicles still have their original trajectories, it is necessary to detect whether there is any conflict between the original trajectories and trajectories of other vehicles in the scenario for the next few seconds. If there is no conflict, the vehicles will return to their original trajectories; otherwise, vehicles will not be placed in the simulation scenario temporarily until all conflicts are resolved.

D. Performance

The generation of near-realistic traffic flows matters for a traffic simulator. LimSim can simulate human driving behavior and produce diverse traffic flows by adjusting the SVO parameters of the PDP process. To verify the performance of the simulated traffic flow generated by LimSim, we conduct a simulation based on the Freeway B dataset from CitySim [36] and analyze the speed distribution and car-following distance distribution of the simulation results. The Freeway B dataset is a six-lane highway in both directions, containing two roads, a westbound road and an eastbound road. The westbound road is relatively congested with a higher traffic volume than the eastbound road. Simulation results are shown in Fig. 7.

The velocity distributions of the LimSim-generated traffic flows and the real data closely resemble the normal distribution with similar means and standard deviations. The

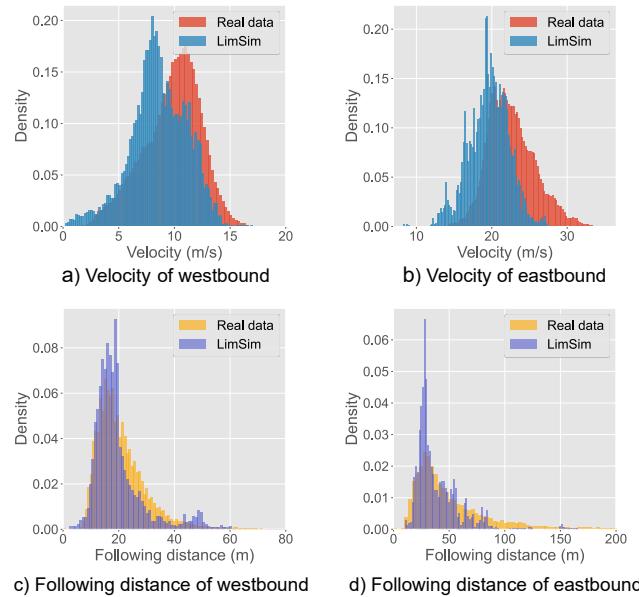


Fig. 7. Comparison of velocity and following distance between LimSim and real data: In both the LimSim simulation and the real dataset, the speed distributions are close to the normal distribution, with similar mean and standard deviation values. The distributions of the following distance are also close to the Poisson distribution, with similar parameters between the simulated and real data. These results show that LimSim can effectively simulate human driving behavior and generate realistic traffic flow.

mean speed of the LimSim-generated traffic flow on the westbound road is 8.57 m/s with a standard deviation of 2.57 m/s, while the mean speed of the real data is 9.81 m/s with a standard deviation of 2.58 m/s. The mean speed of the LimSim-generated traffic flow on the eastbound road is 19.57 m/s with a standard deviation of 2.67 m/s, while the mean speed of the real data is 22.62 m/s with a standard deviation of 3.11 m/s. The average speed on the westbound road is lower than that on the eastbound road, which is consistent with the real dataset. Two distributions of the car-following distance closely resemble the Poisson distribution, and the average distance in the eastbound direction is greater than that in the westbound direction, which is also consistent with the dataset. Overall, LimSim can simulate human driving behavior and provide traffic scenarios that are consistent with the dataset.

The experiment and demo videos of LimSim can be found on YouTube: <https://youtu.be/YR2A25v0hj4>.

V. CONCLUSION

In this paper, we introduce LimSim, the long-term interactive traffic simulator, which ensures the continuity of macroscopic and microscopic traffic flows and provides multi-type fine-grained dynamic traffic scenarios. LimSim adopts a closed-loop framework that supports a variety of driving behaviors in different scenarios while maintaining a realistic traffic environment and supporting multiple data formats. LimSim performs a multi-dimensional real-time evaluation of the long-term simulation process to extract rare and challenging scenarios for further analysis. Besides, LimSim provides easy-to-use APIs to help users evaluate the

performance of their own PDP solutions.

We will open source our code and encourage community users to extend the current framework to support more personalised tasks, subject to license compliance. We will continue to maintain LimSim and continuously add useful features. We also plan to make LimSim compatible with various datasets to provide more realistic traffic scenarios for users involved in autonomous driving research.

ACKNOWLEDGMENTS

This work was supported by the Science and Technology Commission of Shanghai Municipality (Nos. 22YF1461400 and 22DZ1100102).

REFERENCES

- [1] F. Sagberg, Selpi, G. F. Bianchi Piccinini, and J. Engström, “A review of research on driving styles and road safety,” *Human factors*, vol. 57, no. 7, pp. 1248–1275, 2015.
- [2] Y. Zhang, W. Jin, Z. Xiong, Z. Li, Y. Liu, and X. Peng, “Demystifying interactions between driving behaviors and styles through self-clustering algorithms,” in *HCI in Mobility, Transport, and Automotive Systems*, 2021, pp. 335–350.
- [3] H. Paul ’t, “Preparing policy makers for crisis management: The role of simulations,” *Journal of Contingencies and Crisis Management*, vol. 5, no. 4, pp. 207–215, 1997.
- [4] J. S. Teo, E. Taniguchi, and A. G. Qureshi, “Evaluating city logistics measure in e-commerce with multiagent systems,” *Procedia-Social and Behavioral Sciences*, vol. 39, pp. 349–359, 2012.
- [5] M. Daniel, R. Dostál, S. Kozhevnikov, A. Matysková, K. Moudrá, A. M. Pereira, and O. Přibyl, “City simulation software: Perspective of mobility modelling,” in *Smart City Symposium Prague (SCSP)*, 2021, pp. 1–7.
- [6] S.-H. Wang, C.-H. Tu, and J.-C. Juang, “Automatic traffic modelling for creating digital twins to facilitate autonomous vehicle development,” *Connection Science*, vol. 34, no. 1, pp. 1018–1037, 2022.
- [7] J. Argota Sánchez-Vaquerizo, “Getting real: The challenge of building and validating a large-scale digital twin of Barcelona’s traffic with empirical data,” *ISPRS International Journal of Geo-Information*, vol. 11, no. 1, p. 24, 2022.
- [8] H.-P. Schöner, “Simulation in development and testing of autonomous vehicles,” in *Internationales Stuttgarter Symposium: Automobil- und Motorenrechnik*, 2018, pp. 1083–1095.
- [9] G. D. Cameron and G. I. Duncan, “PARAMICS—parallel microscopic simulation of road traffic,” *The Journal of Supercomputing*, vol. 10, pp. 25–53, 1996.
- [10] M. Fellendorf and P. Vortisch, “Microscopic traffic flow simulator VISSIM,” *Fundamentals of Traffic Simulation*, pp. 63–93, 2010.
- [11] J. Casas, J. L. Ferrer, D. Garcia, J. Perarnau, and A. Torday, “Traffic simulation with Aimsun,” *Fundamentals of Traffic Simulation*, pp. 173–232, 2010.
- [12] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücke, J. Rummel, P. Wagner, and E. Wießner, “Microscopic traffic simulation using sumo,” in *International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 2575–2582.
- [13] S. Shah, D. Dey, C. Lovett, and A. Kapoor, “AirSim: High-fidelity visual and physical simulation for autonomous vehicles,” in *Field and Service Robotics: Results of the 11th International Conference*, 2018, pp. 621–635.
- [14] G. Rong, B. H. Shin, H. Tabatabaei, Q. Lu, S. Lemke, M. Možek, E. Boise, G. Uhm, M. Gerow, S. Mehta et al., “LGSVL simulator: A high fidelity simulator for autonomous driving,” in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2020, pp. 1–6.
- [15] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, “CARLA: An open urban driving simulator,” in *Conference on Robot Learning (CoRL)*, 2017, pp. 1–16.
- [16] Q. Li, Z. Peng, L. Feng, Q. Zhang, Z. Xue, and B. Zhou, “MetaDrive: Composing diverse driving scenarios for generalizable reinforcement learning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 3, pp. 3461–3475, 2022.
- [17] L. Bergamini, Y. Ye, O. Scheel, L. Chen, C. Hu, L. Del Pero, B. Osiński, H. Grimmett, and P. Ondruska, “SimNet: Learning reactive self-driving simulations from real-world observations,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 5119–5125.
- [18] Q. Sun, X. Huang, B. C. Williams, and H. Zhao, “InterSim: Interactive traffic simulation via explicit relation modeling,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 11416–11423.
- [19] L. Feng, Q. Li, Z. Peng, S. Tan, and B. Zhou, “TrafficGen: Learning to generate diverse and realistic traffic scenarios,” *ArXiv preprint arXiv:2210.06069*, 2022.
- [20] L. E. Owen, Y. Zhang, L. Rao, and G. McHale, “Traffic flow simulation using CORSIM,” in *Winter Simulation Conference Proceedings*, vol. 2, 2000, pp. 1143–1147.
- [21] J. Xu, Q. Luo, K. Xu, X. Xiao, S. Yu, J. Hu, J. Miao, and J. Wang, “An automated learning-based procedure for large-scale vehicle dynamics modeling on Baidu Apollo platform,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 5049–5056.
- [22] J. Craighead, R. Murphy, J. Burke, and B. Goldzie, “A survey of commercial & open source unmanned vehicle simulators,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2007, pp. 852–857.
- [23] S. Carpin, M. Lewis, J. Wang, S. Balakirsky, and C. Scrapper, “USARSim: a robot simulator for research and education,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2007, pp. 1400–1405.
- [24] O. Michel, “Cyberbotics ltd. webots™: professional mobile robot simulation,” *International Journal of Advanced Robotic Systems*, vol. 1, no. 1, p. 5, 2004.
- [25] T. Laue, K. Spiess, and T. Röfer, “SimRobot—a general physical robot simulator and its application in robocup,” in *RoboCup 2005: Robot Soccer World Cup IX*, 2006, pp. 173–183.
- [26] A. Juliani, V.-P. Berges, E. Teng, A. Cohen, J. Harper, C. Elion, C. Goy, Y. Gao, H. Henry, M. Mattar et al., “Unity: A general platform for intelligent agents,” *ArXiv preprint arXiv:1809.02627*, 2018.
- [27] C. Pilz, G. Steinbauer, M. Schratter, and D. Watzenig, “Development of a scenario simulation platform to support autonomous driving verification,” in *IEEE International Conference on Connected Vehicles and Expo (ICCVE)*, 2019, pp. 1–7.
- [28] N. Koenig and A. Howard, “Design and use paradigms for gazebo, an open-source multi-robot simulator,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3, 2004, pp. 2149–2154.
- [29] S. Feng, H. Sun, X. Yan, H. Zhu, Z. Zou, S. Shen, and H. X. Liu, “Dense reinforcement learning for safety validation of autonomous vehicles,” *Nature*, vol. 615, no. 7953, pp. 620–627, 2023.
- [30] J. Houston, G. Zuidhof, L. Bergamini, Y. Ye, L. Chen, A. Jain, S. Omari, V. Iglovikov, and P. Ondruska, “One thousand and one hours: Self-driving motion prediction dataset,” in *Conference on Robot Learning*. PMLR, 2021, pp. 409–418.
- [31] J. C. Hayward, “Near-miss determination through use of a scale of danger,” *Highway Research Record*, 1972.
- [32] M. Treiber, A. Hennecke, and D. Helbing, “Congested traffic states in empirical observations and microscopic simulations,” *Physical review E*, vol. 62, no. 2, p. 1805, 2000.
- [33] L. Wen, P. Cai, D. Fu, S. Mao, and Y. Li, “Bringing diversity to autonomous vehicles: An interpretable multi-vehicle decision-making and planning framework,” *ArXiv preprint arXiv:2302.06803*, 2023.
- [34] C. Li, T. Trinh, L. Wang, C. Liu, M. Tomizuka, and W. Zhan, “Efficient game-theoretic planning with prediction heuristic for socially-compliant autonomous driving,” *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10248–10255, 2022.
- [35] W. Schwarting, A. Pierson, J. Alonso-Mora, S. Karaman, and D. Rus, “Social behavior for autonomous vehicles,” *Proceedings of the National Academy of Sciences*, vol. 116, no. 50, pp. 24972–24978, 2019.
- [36] O. Zheng, M. Abdel-Aty, L. Yue, A. Abdelraouf, Z. Wang, and N. Mahmoud, “CitySim: A drone-based vehicle trajectory dataset for safety oriented research and digital twins,” *ArXiv preprint arXiv:2208.11036*, 2022.
- [37] D. Rempe, J. Philion, L. J. Guibas, S. Fidler, and O. Litany, “Generating useful accident-prone driving scenarios via a learned traffic prior,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 17305–17315.