SUMO User Conference 2022
https://doi.org/10.52825/scp.v3i.110

Extending SUMO for Lane-Free Microscopic Simulation of Connected and Automated Vehicles

Dimitrios Troullinos[1][https://orcid.org/0000-0003-3228-3888],
Georgios Chalkiadakis[1][https://orcid.org/0000-0002-0716-2972],
Diamantis Manolis[1][https://orcid.org/0000-0003-4569-8732],
Ioannis Papamichail[1][https://orcid.org/0000-0002-6332-9327], and
Markos Papageorgiou[1][https://orcid.org/0000-0001-5821-4982]

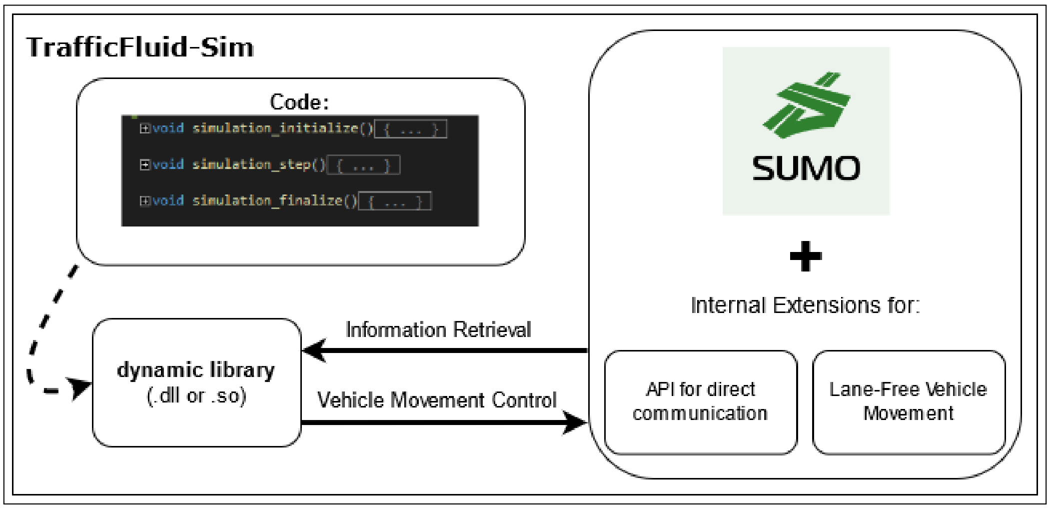
[1]Technical University of Crete, Chania, Greece
{dtroullinos, dmanolis, ipapa, markos}@dssl.tuc.gr
gehalk@intelligence.tuc.gr

Abstract: This paper presents some new developments related to TrafficFluid-Sim, a lane-free microscopic simulator that extends the SUMO simulation infrastructure to model lane-free traffic environments, allowing vehicles to be located at any lateral position, disregarding standard notions of car-following and lane-change maneuvers that are typically embedded within a (lane-based) simulator. A dynamic library has been designed for traffic monitoring and lane-free vehicle movement control, one that does not impose any inter-tool “communication” delays that standard practices with the TraCI module introduce; and enables the emulation of vehicle-to-vehicle and vehicle-to-infrastructure communication. We first summarize the various core components that constitute our simulator, and then discuss the new capability to utilize the bicycle kinematic model, additionally to the usual double-integrator model, as a more realistic model of vehicle movement dynamics, particularly for a lane-free traffic environment. Finally, we developed the necessary components so that the bicycle model can alternatively be combined with the use of global coordinates for more realistic simulation in road networks with curvature, such as roundabouts.
Keywords: lane-free traffic, microscopic modelling and simulation, connected and automated vehicles

Introduction

Technological advancements in the automotive industry reinforce the promise of Connected and Automated Vehicles (CAVs) [1] featuring superb perception and automated driving capabilities, fostered by vehicle-to-vehicle and vehicle-to-infrastructure communications. In consequence, novel traffic flow paradigms appear that consider current or projected capabilities of CAVs, such as the TrafficFluid paradigm [2] that investigates novel traffic environments featuring two novel vehicle characteristics, namely: (i) lane-free traffic, meaning that vehicles' lateral placement can be arbitrary within the road boundaries; and (ii) vehicles may use their automated driving and connectivity capabilities to apply “vehicle nudging” caused by other neighboring vehicles. Such a pushing force may lead vehicles to adjust their lateral position appropriately to accommodate faster vehicles upstream to pass. In addition, nudging is found to lead to improved characteristics of the emerging traffic flow, e.g., in terms of stability and capacity [2].

The use of traffic simulation software is central in the design and testing of vehicle movement strategies and related applications and can significantly facilitate research in emergent



```
⊞void simulation_initialize(){ ... }
⊞void simulation_step(){ ... }
⊞void simulation_finalize(){ ... }
```

Figure 1. A high-level overview of the different parts that constitute our lane-free microscopic simulation tool.

traffic paradigms. As such, it is important to have an appropriate simulator for lane-free traffic environments, one that allows for large-scale simulation scenarios, and its use can be easily generalized for different types of vehicles, connectivity schemes and road network structures.

To this end, we provide an extension of the SUMO simulator infrastructure, rendering it appropriate for the design and assessment of several vehicle movement strategies, including vehicle nudging, under the lane-free paradigm, considering the existence of CAVs. In what follows, we first summarize the most important aspects of our simulator, which are presented in more detail in [3]. Then, we present our newer development, the adoption of the bicycle model as an alternative to the existing double-integrator model, and the use of global coordinates for certain applications. Finally, we discuss imminent future work and conclusions.

Simulator Overview

SUMO [4] (Simulation of Urban MObility) is the prevalent platform to work with since it is an open-source project and therefore appropriate for the adjustments and extensions needed. While the use of TraCI for CAV related endeavors is quite popular in other existing tools, such as VEINS [5], “iTETRIS Control System” (iCS) [6] and MOSAIC (formerly known as VSimRTI) [7], TraCI was not considered due to limited efficiency and lack of customizability. Note that we are interested in designing and testing novel vehicle movement strategies in lane-free environments, and that our simulator should support large-scale experiments. Simulations with a large number of controllable vehicles would require a substantial amount of communication, when using TraCI, since each vehicle would request relevant information through the TraCI API; and also provide a custom control input through it. Therefore, in simulations containing, e.g., thousands of vehicles, these delays would impose a significant bottleneck. We note that, while the Libsumo tool of SUMO addresses the communication overhead of TraCI, its use is still quite limited and it does not operate with the GUI application of SUMO. Moreover, TraCI (and SUMO in general) is designed for lane-based traffic, meaning that it relies on notions tied to lane-based traffic, such as car-following and lane-changing behaviors. Only through internal modifications and extensions in the codebase we could provide a simulator that is appropriate for simulation in lane-free traffic environments, and one that allows for further customization and extensions for imminent and future requirements.

Therefore, we have opted to construct a new dynamic library and an API that are tied to lane-free traffic environments. In Figure 1 we show a high-level overview of our application, which

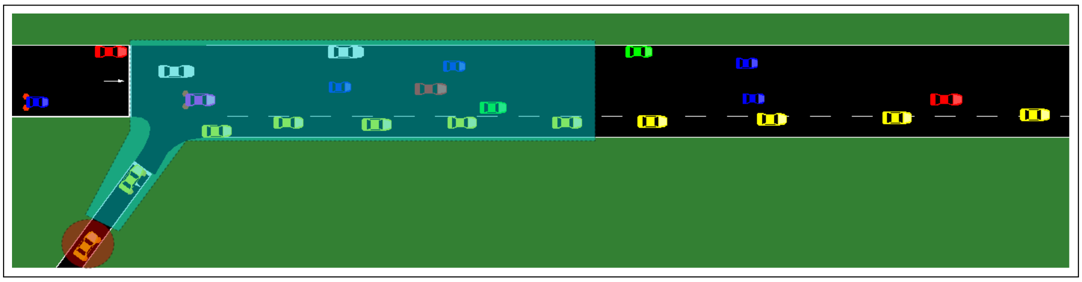


Figure 2. The ego vehicle on the on-ramp indicated with red, has direct access to the vehicles downstream located in the next segments.

contains the dynamic library that the user has access to and can develop code in C/C++ for any vehicle movement strategy, provided that the code yields the necessary control input of the vehicles in every discrete time-step. This is in contrast to the standard operation of SUMO, where the user has to select among a specific range of car-following and lane-changing models, and can only tune the associated set of parameters.

The code developed by the user is compiled into a .dll or .so file (depending on the operating system, windows or linux based), and is connected with our extension of the SUMO application, that contains the implementation of our API and the enabling of lane-free vehicle movement. With this API, the user has access to information regarding the traffic environment through the use of unique IDs (similar to TraCI). One can monitor online each vehicle's current status (position and speed), and request information regarding certain vehicle properties (e.g., length, width, followed route). Road Networks in SUMO have a graph structure, with road segments corresponding to the graph's edges. According to the unique ID of each segment, the user has access to relevant information, such as the IDs of the vehicles currently within that segment (ordered according to longitudinal position), and information regarding densities of vehicles within user-defined adjustable subregions of the segment. Dimensions of each road segment (length and width) are also available. Moreover, simple loop detectors placed through the road network can be monitored through the API as well.

We should emphasize that we mostly rely on the local coordinate system that vehicles employ through SUMO, meaning that each vehicle's position is w.r.t. the current road segment. This lifts the task of manually performing turning operations when needed and allows us to design vehicle movement strategies considering that each ego vehicle always operates on a straight highway. Essentially, a vehicle only needs to be placed appropriately laterally so that it follows the requested route (this is further discussed in future work, see Section 4). Each vehicle can observe downstream and upstream traffic, w.r.t. its own position and routing, and obtain an ordered (according to longitudinal distance) set of the neighbor IDs downstream and upstream. The important aspect of this feature is that the ego vehicle can automatically obtain information about the vehicle in the next (or previous, for upstream requests) road segments, depending on the specified range. Figure 2 showcases an example with an ego vehicle on an on-ramp that is about to enter the acceleration lane of the highway. It can request access to vehicles downstream by simply providing a longitudinal observation distance. The ego vehicle observes the highway as an unfolded straight road due to geometry of the road being handled by SUMO, and we performed the necessary developments so that information regarding longitudinal and lateral distances of neighboring vehicles (from the ego vehicle) is calculated accordingly.

Lane-Free Vehicle Movement with the Bicycle Model

In this section, we first briefly address how we consider the positioning of the vehicles in SUMO for our lane-free settings, and discuss the standard use of the double-integrator model for the movement dynamics. Then, we present the new option for movement dynamics with the bicycle kinematic model, along with the alternative to operate under global coordinates alongside this model.

Longitudinal and Lateral Positioning in SUMO

Regarding the positioning of the vehicles, the *local* coordinates (x, y) of the rectangular-shaped vehicles that we consider are as follows: The longitudinal position x of the vehicle is the distance of its center point from the starting point of the road segment. On the other hand, the lateral position y measures the distance from the right road boundary to the vehicle's center point. Hence, the vehicles observe a single lateral position, without knowledge about lateral placement w.r.t. to lane centers, as this information is not meaningful anymore for designing a vehicle movement control strategy under lane-free settings.

Double-Integrator Model

The 'Ballistic-Update' option of SUMO for the longitudinal movement, and the incorporation of lateral dynamics in a similar double-integrator manner, constitute a double-integrator model for the lane-free vehicle movement dynamics, in which longitudinal and lateral movements are disjointed and are controlled by respective independent (longitudinal and lateral) accelerations. For the double-integrator model, the orientation of the vehicle is essentially according to the ratio of longitudinal versus lateral speed, but it is always considered in parallel with the road boundaries for simplicity. This approximation is good enough for highways, where we typically have vehicles moving with high longitudinal speeds, while lateral speeds are much smaller. In the following subsection, we present an alternative approach, namely the bicycle kinematic model, which provides a more realistic depiction of movement dynamics, since it incorporates explicitly the orientation of the vehicle.

The associated controller for vehicles employing the double-integrator model should provide the control inputs, i.e., a longitudinal and a lateral acceleration (in m/s^2) value in every time-step. Also, under the lane-free paradigm, where vehicles can be placed anywhere laterally within the road boundaries, a collision between two vehicles is reported when their rectangular shapes overlap. This is a straightforward check, given the vehicles' information regarding positioning and their dimensions (length, width), since the vehicles' orientation is assumed parallel with the road boundaries.

Bicycle Kinematic Model

An important development, necessary to enable microscopic simulation of complex urban network applications such as [8], is the incorporation of the *bicycle kinematic model* [9] into the simulator. In Figure 3, we show a snapshot from a roundabout scenario containing vehicles that use the approach of [8] and utilize the bicycle kinematic model and global coordinates. In this model, the vehicle's front (and back) wheels are abstracted to a unique front (and back) wheel located at the respective axle middle points, whereby the front wheel is steerable, controlling the vehicle's orientation, and there is a unique forward acceleration (control input) and a unique forward speed, both in the direction of the current vehicle orientation. We refer the interested reader to the relevant paper [9] for more details regarding this kinematic model. In this model, longitudinal and lateral dynamics are interconnected and nonlinear, and, more generally, the bicycle model is more accurate in describing vehicle movement than the linear double-integrator model and is particularly interesting in a lane-free environment, when vehicles are driving in curves at relatively low speeds, such as in urban networks and roundabouts. In such cases, the user has the option to select an alternative method for vehicle movement dynamics in lane-free traffic. The bicycle model's state variables are, beyond the longitudinal and lateral vehicle

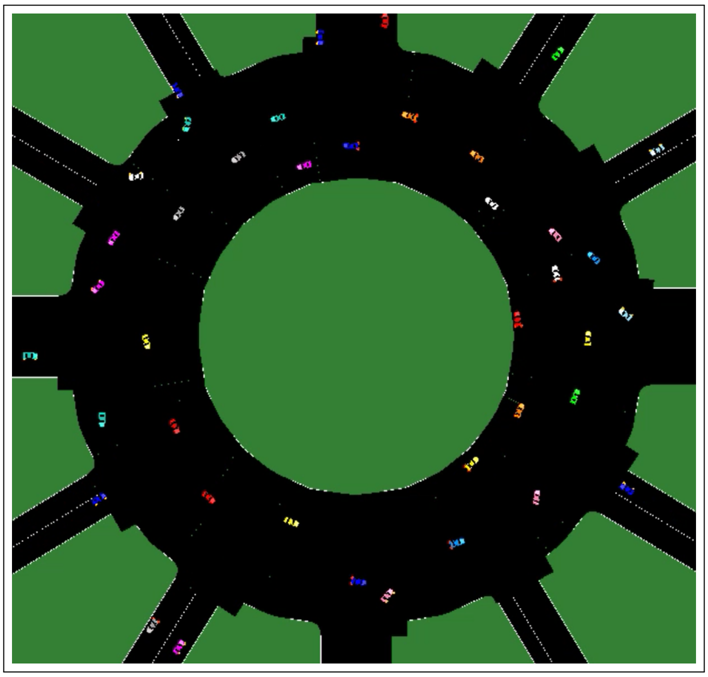


Figure 3. Vehicles employing the bicycle kinematic model with global coordinates in a roundabout scenario.

position (x, y), the vehicle orientation θ and the forward speed v; while the control variables are the forward acceleration and the steering angle. Figure 4 illustrates the state variables of the bicycle model, where we do not have two separate dynamics for longitudinal and lateral movements, but a unified and more realistic movement, involving the actual orientation resulting from steering the vehicle.

In terms of technical developments for the simulation, this involved an internal implementation regarding the position (x, y) and speed v update process of the vehicles, according to the bicycle kinematic model. Information regarding the orientation θ of the vehicle and the ability to directly change it is already available through SUMO, so we can adjust its value through internal extensions within the source code. The local coordinates (x, y) of the vehicle again refer to its center, as discussed earlier, and as depicted in Figure 4. However, the bicycle model utilizes the position corresponding to (x_b, y_b) in Figure 4, i.e., the vehicle's orientation changes with respect to this point. This is considered within our implementation, and this point is also directly available to the user who wishes to design a movement strategy using the bicycle model. User access to the orientation of the vehicles is granted through the API, providing either global (with respect to the global coordinate system of SUMO) or local (with respect to the longitudinal axis of the current road vehicles reside) coordinates. The associated lane-free controller should simply provide the value for the forward acceleration F (in m/s^2) and the steering angle δ (in rad) as control inputs for every time-step (instead of the longitudinal and lateral accelerations of the double-integrator model).

As of now, we have a simplified way that incorporates the orientation of the vehicles for the collision check. Essentially, we draw a rectangle that is in parallel to the road and contains the vehicle (red rectangular in Figure 4), and report a collision if these rectangular regions of two vehicles overlap. Of course, this procedure may report a collision even if two vehicles do not actually collide. However, it also guarantees that no collision will be disregarded. An exact collision identification method is in the course of implementation and will be reported when ready and tested.

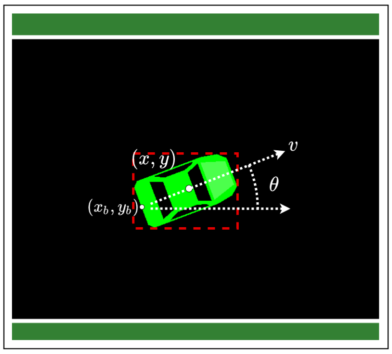


Figure 4. Illustration of a vehicle utilizing the bicycle model.

Bicycle Kinematic Model with Global Coordinates

The local coordinate system of SUMO is utilized for the bicycle model, and, as such, controllers do not need to take into account the geometry of the residing road. Hence, the operating orientation of the vehicles is local, i.e., with respect to the road segment the vehicle is currently located at, so turning operations (e.g., at junction points) are handled through SUMO. However, for certain applications, we may be interested in controlling turning operations instead of letting SUMO automatically handle such procedures, something that is also needed for more realistic depiction in road structures with continuous curvatures, e.g., roundabouts. We also provide the option to do so, in a way that does not affect or cause regression to existing functionalities. One can utilize the bicycle kinematics along with global coordinate control for more realistic behavior, that does not succumb to the road structure, but is rather based on the global Cartesian coordinates (x, y). This feature serves to facilitate an impending application that involves lane-free movement in a roundabout, where we wish to work with junctions for vehicles entering and exiting, and utilize polar coordinates instead of Cartesian ones for vehicles operating within the roundabout. This provides a more realistic and more convenient depiction of the actual vehicle and emerging traffic behavior compared to the use of the standard local coordinate system SUMO provides. Of course, in such a global environment, we need to properly design the above-mentioned behaviors.

For this feature, the use of internal mapping from global to local coordinates (and vice-versa) was crucial, given that the existing infrastructure and our extensions rely on the local coordinate system that SUMO provides. If we would completely neglect the local coordinates in favor of simplicity in the development process, then certain features would not function properly, e.g., the observation capabilities on surrounding vehicles, and as such, essential information for the design and real-time operation of vehicle movement strategies would not be available. Global coordinate control is tied only with the bicycle kinematics, since it would be quite restrictive for the double-integrator model, due to the absence of orientation control.

Future Extensions

A forthcoming extension is the incorporation of lateral boundaries based on the desired path for vehicles to follow, along with the capability to control them at execution time through the API for generalizing the vehicles' behavior in more complex road networks with vehicles following different routing schemes. In lane-based environments, vehicles can follow any (feasible) path by simply choosing lanes appropriately. For instance, a vehicle entering a highway from an on-ramp will typically need to perform a lane-change in order to merge on the highway. SUMO provides information regarding the availability of the road downstream through information on available lanes downstream. In our case, a vehicle adhering to the lane-free paradigm, wishing to enter a highway through an on-ramp, will again need to merge on the highway appropriately.

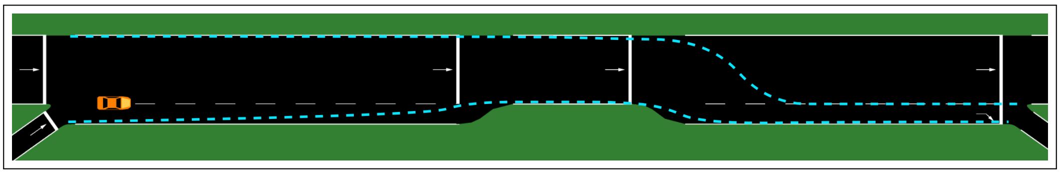


Figure 5. Left and right lateral boundaries for vehicles entering or scheduled to exit from the off-ramp.

Then, it is important to introduce an equivalent notion in a lane-free domain, since providing information on available lanes downstream is not appropriate in lane-free environments.

As of now, such maneuvers are performed with ad-hoc techniques, tied to specific traffic scenarios. Yet, there is a need to establish a formal way for vehicles to follow left and right boundaries on the road, guiding them to always operate within an admissible lateral region, so as to comply with their routing. This is illustrated in Figure 5, where we observe a vehicle with a route initiating from the on-ramp and leading to the off-ramp. The blue lines indicate the left and right lateral bounds, relevant to the path the vehicle needs to follow. As such, the vehicle will be able to request information regarding these two lateral bounds through the API, at any longitudinal distance corresponding to its position. Therefore, the vehicle will have the necessary time to adjust its behavior in order to be located within the bounds, and, as a consequence, to follow its desired routing scheme. Besides the application for on-ramps and off-ramps, this feature is crucial to enable microscopic simulation on emerging applications such as internal boundary control in lane-free traffic for two-way streams, as introduced in [10]. This will also involve an online update process for the boundaries' lateral position through the API.

Conclusions

In this work, we presented some advancements of TrafficFluid-Sim, an extension of SUMO appropriate for lane-free traffic environments, that is developed for the research project *TrafficFluid* [2]. Specifically, we discussed on the new capability to employ a more realistic depiction for movement dynamics with the bicycle model, which is more appropriate in certain applications that consider the lane-free paradigm. This tool is already being utilized for the design and evaluation of various vehicle movement strategies [2], [11]–[14].

Data Availability Statement

The associated code is developed for the Trafficfluid project [2] and cannot be shared as of now.

Underlying and related material

The interested reader may refer to `https://bit.ly/3tf53jD`, `https://bit.ly/3K4azwz`, `https://bit.ly/3AtBNJR`, and `https://bit.ly/3K3olzc` for work that utilizes TrafficFluid-Sim, and to `https://trafficfluid.tuc.gr` for more information regarding TrafficFluid.

Author contributions

The authors confirm contribution to the paper as follows: study conception and design: D. Troullinos, G. Chalkiadakis, I. Papamichail, M. Papageorgiou; data collection: D. Troullinos, D. Manolis; analysis and interpretation of results: D. Troullinos, I. Papamichail, M. Papageorgiou; draft manuscript preparation: D. Troullinos, G. Chalkiadakis, I. Papamichail, M. Papageorgiou. All authors reviewed the results and approved the final version of the manuscript.

Competing interests

The authors declare no competing interests.

Funding

The research leading to these results has received funding from the European Research Council under the European Union's Horizon 2020 Research and Innovation programme/ ERC Grant Agreement n. [833915], project TrafficFluid.

References

[1] D. Elliott, W. Keen, and L. Miao, "Recent advances in connected and automated vehicles," *Journal of Traffic and Transportation Engineering (English Edition)*, vol. 6, no. 2, pp. 109–131, 2019, ISSN: 2095-7564. DOI: `https://doi.org/10.1016/j.jtte.2018.09.005`.

[2] M. Papageorgiou, K.-S. Mountakis, I. Karafyllis, I. Papamichail, and Y. Wang, "Lane-free artificial-fluid concept for vehicular traffic," *Proceedings of the IEEE*, vol. 109, no. 2, pp. 114–121, 2021. DOI: `10.1109/JPROC.2020.3042681`.

[3] D. Troullinos, G. Chalkiadakis, D. Manolis, I. Papamichail, and M. Papageorgiou, "Lane-free microscopic simulation for connected and automated vehicles," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 3292–3299. DOI: `10.1109/ITSC48978.2021.9564637`.

[4] P. A. Lopez, M. Behrisch, L. Bieker-Walz, *et al.*, "Microscopic traffic simulation using sumo," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 2575–2582. DOI: `10.1109/ITSC.2018.8569938`.

[5] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing (TMC)*, vol. 10, no. 1, pp. 3–15, Jan. 2011. DOI: `10.1109/TMC.2010.133`.

[6] M. Rondinone, J. Maneros, D. Krajzewicz, *et al.*, "iTETRIS: A modular simulation platform for the large scale evaluation of cooperative its applications," *Simulation Modelling Practice and Theory*, vol. 34, pp. 99–125, 2013, ISSN: 1569-190X. DOI: `https://doi.org/10.1016/j.simpat.2013.01.007`.

[7] B. Schünemann, "V2X simulation runtime infrastructure VSimRTI: An assessment tool to design smart traffic management systems," *Computer Networks*, vol. 55, no. 14, pp. 3189–3198, 2011, ISSN: 1389-1286. DOI: `https://doi.org/10.1016/j.comnet.2011.05.005`.

[8] M. Naderi, M. Papageorgiou, I. Papamichail, and I. Karafyllis, "Automated vehicle driving on large lane-free roundabouts," in *2022 IEEE International Intelligent Transportation Systems Conference (ITSC), accepted*, 2022.

[9] P. Polack, F. Altché, B. d'Andréa-Novel, and A. de La Fortelle, "The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles?" In *2017 IEEE Intelligent Vehicles Symposium (IV)*, 2017, pp. 812–818. DOI: `10.1109/IVS.2017.7995816`.

[10] M. Malekzadeh, I. Papamichail, M. Papageorgiou, and K. Bogenberger, "Optimal internal boundary control of lane-free automated vehicle traffic," *Transportation Research Part C: Emerging Technologies*, vol. 126, p. 103 060, 2021, ISSN: 0968-090X. DOI: `https://doi.org/10.1016/j.trc.2021.103060`.

[11] D. Troullinos, G. Chalkiadakis, I. Papamichail, and M. Papageorgiou, “Collaborative multiagent decision making for lane-free autonomous driving,” in *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, Virtual Event, United Kingdom, 2021, pp. 1335–1343. [Online]. Available: `https://dl.acm.org/doi/10.5555/3463952.3464106`.

[12] V. K. Yanumula, P. Typaldos, D. Troullinos, M. Malekzadeh, I. Papamichail, and M. Papageorgiou, “Optimal path planning for connected and automated vehicles in lane-free traffic,” in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 3545–3552. DOI: `10.1109/ITSC48978.2021.9564698`.

[13] D. Troullinos, G. Chalkiadakis, V. Samoladas, and M. Papageorgiou, “Max-sum with quadtrees for decentralized coordination in continuous domains,” in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, International Joint Conferences on Artificial Intelligence Organization, 2022, pp. 518–526. DOI: `10.24963/ijcai.2022/74`.

[14] M. Malekzadeh, D. Manolis, I. Papamichail, and M. Papageorgiou, “Empirical investigation of properties of lane-free automated vehicle traffic,” in *2022 IEEE International Intelligent Transportation Systems Conference (ITSC), accepted*, 2022.