
Article:
Smilovitskiy, M., Olmez, S. orcid.org/0000-0002-8802-4028, Richmond, P. orcid.org/0000-0002-4657-5518 et al. (5 more authors) (2025) FLAME-GPU for traffic systems: a scalable agent-based simulation framework. Systems, 13 (5). 376. ISSN 2079-8954

https://doi.org/10.3390/systems13050376

---

systems

MDPI

Technical Note

FLAME-GPU for Traffic Systems: A Scalable Agent-Based Simulation Framework †

Maxim Smilovitskiy [1], Sedar Olmez [1,*], Paul Richmond [2], Robert Chisholm [2], Peter Heywood [2], Alvaro Cabrejas [1], Sven van den Berghe [1] and Sachio Kobayashi [1]

[1] Fujitsu Research of Europe, The Urban Building, 3-9 Albert Street, Slough SL1 2BE, UK; maxim.smilovitskiy@fujitsu.com (M.S.); alvaro.cabrejasegea@fujitsu.com (A.C.); sven.vandenberghe@fujitsu.com (S.v.d.B.); satio.kobayashi@fujitsu.com (S.K.)
[2] Department of Computer Science (DCS), University of Sheffield, Regent Court, 211 Portobello, Sheffield S1 4DP, UK; p.richmond@sheffield.ac.uk (P.R.); robert.chisholm@sheffield.ac.uk (R.C.); p.heywood@sheffield.ac.uk (P.H.)
* Correspondence: sedar.olmez@fujitsu.com
† This paper is an extended version of our published paper: A Scalable Agent-Based Model of Traffic Activity Using FLAME-GPU, which was presented at the 22nd International Conference on Practical Applications of Agents and Multi-Agent Systems, PAAMS 2024, held in Salamanca, Spain, during 26–28 June 2024.

Abstract: Agent-based modelling (ABM) has revolutionised the simulation of complex systems, finding applications in diverse fields such as economic markets and traffic management. By modelling individuals as autonomous agents within a dynamic environment, ABM enables the exploration of system behaviours and the evaluation of interventions at various spatiotemporal resolutions. However, the computational intensity of ABM, particularly in large-scale simulations, remains a significant hurdle. This paper presents a novel approach to addressing these challenges through the development of a GPU-accelerated transport model, specifically applied to a road network. Utilising the FLAME-GPU framework, the proposed model demonstrates enhanced scalability and efficiency compared with traditional CPU-based simulations, such as Simulation of Urban MObility (SUMO). Through rigorous comparative analysis, this study highlights significant improvements in simulation speed and the capacity to manage larger vehicle populations. The research underscores the transformative potential of GPU acceleration in mitigating computational constraints within ABM, offering a practical framework for simulating transport systems with greater precision and depth. Extensive experimentation validates the model's ability to realistically simulate the vehicle population of the Isle of Wight, achieving a balance between computational efficiency and the accurate representation of complex traffic dynamics.

Keywords: agent-based model; FLAME-GPU; traffic simulation; individual-based model; data analytics

Academic Editors: Philippe Mathieu, Fernando De la Prieta Pintado and Alfonso González-Briones

Received: 3 March 2025
Revised: 20 March 2025
Accepted: 12 May 2025
Published: 14 May 2025

Citation: Smilovitskiy, M.; Olmez, S.; Richmond, P.; Chisholm, R.; Heywood, P.; Cabrejas, A.; van den Berghe, S.; Kobayashi, S. FLAME-GPU for Traffic Systems: A Scalable Agent-Based Simulation Framework. *Systems* **2025**, *13*, 376. https://doi.org/10.3390/systems13050376

1. Introduction

This paper is an extended version of the conference paper [1] presented at the International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS).

Agent-based modelling (ABM) has emerged as a pivotal tool for simulating complex systems across various domains, enabling researchers to explore the dynamics of financial markets [2–4], traffic systems [5–7], ecological systems [8–10], crime patterns [11–13], and more. ABM encapsulates interactions between agents and their environments, facilitating the examination of individual and collective behaviours within a controlled computational

setting, allowing researchers to document and observe the complex interactions of a system at various spatiotemporal resolutions.

Despite ABM's versatility, its application, particularly in simulating large-scale or emergent phenomena, is often constrained by a lack of computational resources. Simplifications or reductions in agent characteristics or populations are common workarounds, though they may compromise the model's fidelity and outcomes [14–17]. This article addresses these computational challenges, using the transportation domain as an example. The proposed method leverages the computational resources of graphics processing units (GPUs) to simulate traffic behaviour and interactions on the Isle of Wight in the United Kingdom, showcasing the potential for GPU-accelerated ABM in capturing complex transport dynamics.

The Isle of Wight, located off the south coast of England, was selected as the geographical focus for this study due to several factors. Firstly, its status as an island with a self-contained and isolated street network makes it an ideal testbed for spatial modelling, minimising external influences and enhancing the reliability of experimental outcomes. Secondly, the scale of its transportation infrastructure—comprising over 400 miles of roadway and approximately 90,000 registered vehicles—offers a balance between complexity and manageability, ensuring that computational demands remain feasible while still providing a robust environment for model validation.

This research article presents the following:

- A scalable traffic simulation using the Flexible Large-scale Agent Modelling Environment (FLAME-GPU) framework [18], demonstrating the capability to model complex vehicle behaviours at a large scale;
- Several simulation experiments utilising the Isle of Wight's transport network, integrating real-world geographic and traffic data to validate the model's efficacy, where these experiments compare the proposed method with a well-known traditional traffic simulator, namely Simulation of Urban MObility (SUMO).

While limited, there are several studies in the past which have exploited FLAME-GPU's capabilities in modelling transport [19,20] and biological systems [21], evidencing the architecture's reach and accessbility. However, these studies often employed simplified models or, in the case of transport systems, did not address the complexities of real-world street networks. This gap highlights the necessity for a scalable ABM transport model that harnesses the empirical characteristics of the complexities of road networks.

The experiments utilise the Isle of Wight's road network, which ensures that the FLAME-GPU can be benchmarked on a real-world street network, evaluating the proposed model's processing performance and scalability. Key performance metrics include the real-time factor, simulation runtime, interaction rates, and vehicle insertion counts, with comparisons drawn against SUMO.

The remainder of this article is structured as follows. Section 2 reviews existing transport models, comparing and contrasting them against our contributions. Section 3 outlines the FLAME-GPU framework, the proposed model's architecture, and the experiment set-up. Section 4 presents the experimental findings, and Section 5 concludes with an assessment of the proposed approach, limitations, and future research directions.

2. Literature Review

2.1. Agent-Based Modelling of Transport Systems

Agent-based modelling (ABM) has been instrumental in simulating complex systems at various spatiotemporal resolutions. One prominent example is that of the complex interactions of vehicle driver behaviours within street networks, leading to novel insights into individual-level decision making and its ramifications on traffic flow and conges-

tion [22–24]. Despite the wealth of contributions, these models encounter hurdles regarding high-level computing, scalability, and in some cases a lack of empirical geographic data, such as real-world street networks. Notable efforts have integrated empirical city data to refine traffic signal controls [25], explored multi-modal transportation navigation [24,26,27], and sought to estimate and optimise emissions produced from vehicle activity [28].

There are several agent-based models that attempt to model the behaviours of individual vehicle drivers in complex street networks, such as those in [7,23,24]. While these models often lead to useful insights, such as the relationship between speeding and collisions and the impact of demographics on vehicle usage, they experience computational complexity issues, and thus simplified characteristics are selected. While some articles explicitly highlight this to be the issue, others may also suggest other reasons for simplifying their models and sometimes opt for a toy world representation [29,30].

Agent-based models (ABMs) are widely adopted for studying complex systems due to their ability to capture intricate interactions at an individual level. However, researchers often face the challenge of making their models sufficiently detailed while keeping computational demands manageable [29]. To address computational complexity, one approach involves partitioning the simulation environment into smaller, manageable sections. While this method can enhance computational efficiency, it necessitates a deeper understanding of parallelisation and introduces challenges such as spatial and temporal desynchronisation of the results, which may arise due to bottlenecks in the execution of partitioned simulations. Additionally, the exchange of information between partitioned components must be robust to prevent biases or inaccuracies in the outcomes. Thus, achieving a balance between empirical validity and computational feasibility is essential for the development of realistic ABMs. Our proposed method aims to address this balance effectively, contributing to the advancement of the field.

In the following subsection, a popular road transport model is described, this method will be benchmarked against by comparing the proposed methodology to other models to demonstrate how the gaps in the current literature can be overcome.

2.2. Road Transportation and the Role of SUMO

Simulation of Urban MObility (SUMO) [31] is a prominent open-source modelling framework for microscopic traffic simulation. Introduced in the early 2000s, SUMO has become a staple in transportation research, offering versatile modelling capabilities, visualisation tools, and auxiliary features like emission calculation and route planning. By utilising microscopic car-following models, SUMO simulates individual vehicle movements and interactions based on dynamic road conditions and traffic regulations [32], employing a modified version of Krauss's car-following model [33] by default. SUMO benefits from a large developer community that maintains it by updating its features. This makes it a robust method for simulating comprehensive characteristics of traffic systems, thus making it suitable to benchmark against.

One of the key reasons for selecting SUMO as a benchmark tool is its extensive use in the research community, as evidenced by the large number of publications that deployed it. For instance, it has been utilised in the development of traffic management systems (TMSs) to test tree search algorithms for identifying time-saving routes [34]. However, the authors emphasised the importance of applying such systems to real-world use cases by integrating real-time data. Similarly, SUMO has been employed to evaluate traffic flow and test algorithms in real-world locations. For example, the authors of [35] used the city of Wuhan, China as a case study, integrating OpenStreetMap (OSM) data, which are also utilised in this research. The authors focused on a subregion within Wuhan, Jianghan, due to its high population density and significant traffic volume compared with other areas.

In contrast, our proposed method leverages the entire street network of the Isle of Wight in the United Kingdom, simulating tens of thousands of vehicles. Additionally, SUMO has been adopted as a foundational resource for building driving simulators. For instance, the authors of [36] demonstrated the challenges and feasibility of coupling SUMO with SILAB to synchronise data between the two tools. Their findings underscore the need for faster processing architectures to address the challenges of maintaining synchronicity in such simulations.

Although SUMO's open-source nature and detailed simulation capabilities offer significant advantages, several limitations hinder its broader applicability. These include a steep learning curve, restricted flexibility in modifying agent behaviours, a lack of diverse transportation profiles such as cars and buses [37], and most notably, its reliance on a serial central processing unit (CPU) processing model, which limits scalability and performance. This article focuses on the processing capabilities of computational models, particularly the potential of graphics processing unit (GPU) architectures, compared with CPU-based systems. GPUs are widely recognised as superior to CPUs in numerical tasks due to their parallel processing architecture [38]. However, transitioning from serial to parallel processing models to leverage the GPU's capabilities requires significant software engineering effort, presenting integration challenges with established frameworks like SUMO.

In response to these limitations, custom simulation frameworks such as CityFlow [39] have been developed, claiming processing speeds up to 25 times faster than SUMO for large-scale simulations. Despite these advancements, the development of GPU-accelerated solutions capable of handling even larger and more complex network simulations remains an open challenge. This gap highlights a critical area of research that warrants further academic inquiry to achieve the next level of simulation performance and scalability. This research article aims to bridge this gap by leveraging GPU-enhanced ABM within a micro-simulation traffic model, proposing a scalable and complex traffic behaviour simulation framework. By integrating ABM with micro-simulation on the FLAME-GPU platform, we capitalise on GPU parallelism, assigning individual agent state updates to separate GPU cores. This approach heralds a significant leap towards realising a computationally efficient, parallelisable traffic simulator that surpasses the current state of the art.

2.3. GPU Frameworks for Traffic Simulations

As mentioned earlier, GPU-enhanced traffic simulation frameworks have emerged as a pivotal approach to address the computational demands of large-scale simulations [20,40–48]. Several studies have explored this approach, leveraging frameworks such as CUDA [20,47] and OpenCL [45] to achieve significant performance gains. Strippgen and Nagel [47] and Heywood et al. [20] both utilized CUDA, with the former achieving a speedup of over 60 times by employing dynamic queues and ring buffers. Their approach demonstrated the potential of CUDA for efficient memory management and parallel execution but faced scalability challenges with increased agent counts. Heywood et al. [20] implemented fine-grained data parallelism using CUDA, reporting a roughly 43× speedup in agent-based microscopic simulations. This method adopted a graph-based traversal technique. Some articles opted to use OpenCL, including studies by Xiao et al. [48] and Xu et al. [45]. Xiao et al. [48] explored both partial offloading and fully GPU-based execution schemes, achieving up to a 28.7× speedup. Their study highlighted the trade-offs between maintainability and performance. In contrast, Xu et al. [45] combined CUDA and OpenCL in a mesoscopic simulation, focusing on supply simulation with a boundary processing method. They reported an 11.2× speedup but noted limitations in memory access latency and generalisability.

Our proposed approach leverages FLAME-GPU, which offers a flexible and scalable environment for agent-based modelling. By utilizing the Isle of Wight's real-world street network, we demonstrate the capability to simulate complex vehicle behaviours at a large scale. The key contributions of our approach include the following:

- Real-world data integration: Unlike many existing studies, our method integrates real-world geographic and traffic data, ensuring empirical validity and providing a robust environment for model validation.
- Scalability and performance: Our experiments showcase the ability to handle large-scale simulations, with key performance metrics such as the real-time factor, simulation runtime, interaction rates, and vehicle insertion counts.
- Comparison with SUMO: We benchmark our approach against SUMO, a well-established traffic simulator, highlighting the advantages of GPU-accelerated ABM in capturing complex transport dynamics.

While existing studies have predominantly used CUDA and OpenCL, our use of FLAME-GPU provides a more accessible and flexible platform for developing large-scale agent-based simulations. Our focus on real-world street networks and empirical data addresses a gap in the literature, where many studies rely on simplified or synthetic networks. Our approach also achieves competitive speedups compared with existing studies, demonstrating the potential of GPU-accelerated ABM for real-time or faster-than-real-time simulations.

In conclusion, our approach builds on the foundations laid by existing GPU-enhanced traffic simulation frameworks, offering a scalable and empirically validated approach to capturing complex transport dynamics. By leveraging the FLAME-GPU framework and integrating real-world data, we address key limitations in the current literature and contribute to the advancement of GPU-accelerated agent-based modelling in traffic simulations.

3. Methodology

3.1. FLAME-GPU

FLAME-GPU represents a high-performance agent-based simulation framework that harnesses the parallel computing capabilities of modern GPUs to enhance the efficiency and scalability of system simulations. The framework abstracts the complexities of GPU programming, allowing researchers to concentrate on the conceptual design of their models rather than the intricacies of algorithm implementation. This separation of concerns ensures a clear distinction between model representation and execution, enabling the development and simulation of large-scale models within practical timeframes. FLAME-GPU demonstrates versatility across various domains, supporting applications ranging from pedestrian dynamics [49] and road network simulations [19] to cellular biological systems [50]. Using FLAME-GPU requires mapping the system under study to an agent-based paradigm, where agents represent entities with defined states. Messaging mechanisms enable indirect interactions among agents via a global messaging pool, and the environment serves as a repository for globally accessible data.

3.2. Traffic Simulation Model Overview

This section describes the proposed model's approach to simulating vehicular traffic on a microscopic scale within a real-world network, capturing individual vehicle dynamics such as speed and position and incorporating road characteristics like lanes, intersections, and traffic signals.

This article proposes a vehicle behaviour model which modifies the Krauss car-following paradigm, which is utilised in SUMO's simulation mechanics [51,52]. This adaptation is encapsulated by

$$
\begin{aligned}
v_{safe}(t) &= v_l(t) + \frac{g(t) - g_{des}(t)}{\tau_b + \tau}, \\
v_{des}(t) &= \min[v_{max}, v(t) + a(v)\Delta t, v_{safe}], \\
v(t + \Delta t) &= \max[0, v_{des}(t) - \eta], \\
x(t + \Delta t) &= x(t) + v\Delta t,
\end{aligned} \tag{1}
$$

where $g_{des} = \tau v_l(t)$ denotes the desired vehicle gap, with τ as the driver reaction time and $\tau_b = \frac{\overline{v}}{b(\overline{v})}$ representing the braking time, influenced by the average velocity $\overline{v}$ and random perturbation η of model deviations from ideal driving, while $v_{safe}(t)$ is the speed that guarantees the model upholds its non-collision assumption at time t, and $v_{des}(t)$ is the desired speed at time t.

Lane changing and intersection navigation are guided by the principles outlined in [53,54], respectively, with minor technical alterations in implementation that do not affect the principle mechanics and only serve the purpose of better adapting to the FLAME-GPU architecture.

The system of Equation (1) presented above incorporates lane-changing and intersection-crossing rules, alongside additional rules required to update the state of the system. These are expressed as a series of agent functions. The agent variables necessary to execute the functions are stored and exchanged through a global messaging pool, a feature facilitated by FLAME-GPU. The request to update the system's state is fulfilled by launching function kernels for all agents in the agent-state population. Each agent is represented by a thread, and when a kernel is launched, a grid of threads is created. This grid is divided into blocks which are then assigned for concurrent execution to available streaming multiprocessors (SMs) within the GPU. If the total number of threads n exceeds the GPU's maximum thread capacity N_{max}, then the kernel execution is divided into approximately $c = n \; mod \; N_{max}$ steps. While this scheme does not fundamentally change the time complexity of the algorithm, it ensures a smooth and efficient computational scaling process, effectively leveraging the GPU's parallel processing capabilities. Figure 1 describes the aforementioned simulation process.

3.3. Mapping Traffic Model to FLAME-GPU

To map the transportation model onto the FLAME-GPU framework, a network-based messaging approach was employed for agent communication. This approach enables agents to operate within a static multi-lane network, leveraging the network structure to query agents located on the same or adjacent edges and lanes. Both a compressed sparse row (CSR) and compressed sparse column (CSC) representation of the network were stored within the agent environment, facilitating efficient querying of upstream and downstream edges. These network lookups are essential for calculating leader and follower relationships, which are critical for vehicle traversal and junction entry. Additionally, network communication supports key behaviours such as vehicle following, lane changing, and vehicle insertion into the network. Within the FLAME-GPU implementation, vehicles dynamically select lanes while relying on precomputed routes to reach their destinations. This behaviour mirrors that observed in SUMO, where equivalent route files are generated using SUMO's routing tools.

In the FLAME-GPU model, agents are used to represent vehicles. Parameters and distributions from the SUMO model are mapped directly to FLAME-GPU agent variables. Vehicle agents transition between three distinct states. The default state, "driving", involves performing car following, lane changing, and junction traversal. The "pre-insertion" state ensures safe entry onto road edges through gap acceptance, while the "pre-removal" state

allows vehicles to persist within the simulation for data collection before being removed from the network. Additional agents are utilized to represent sections of the road network, enabling efficient data aggregation and collection. Traffic lights are also represented as agents within the FLAME-GPU model, operating with fixed signal patterns.

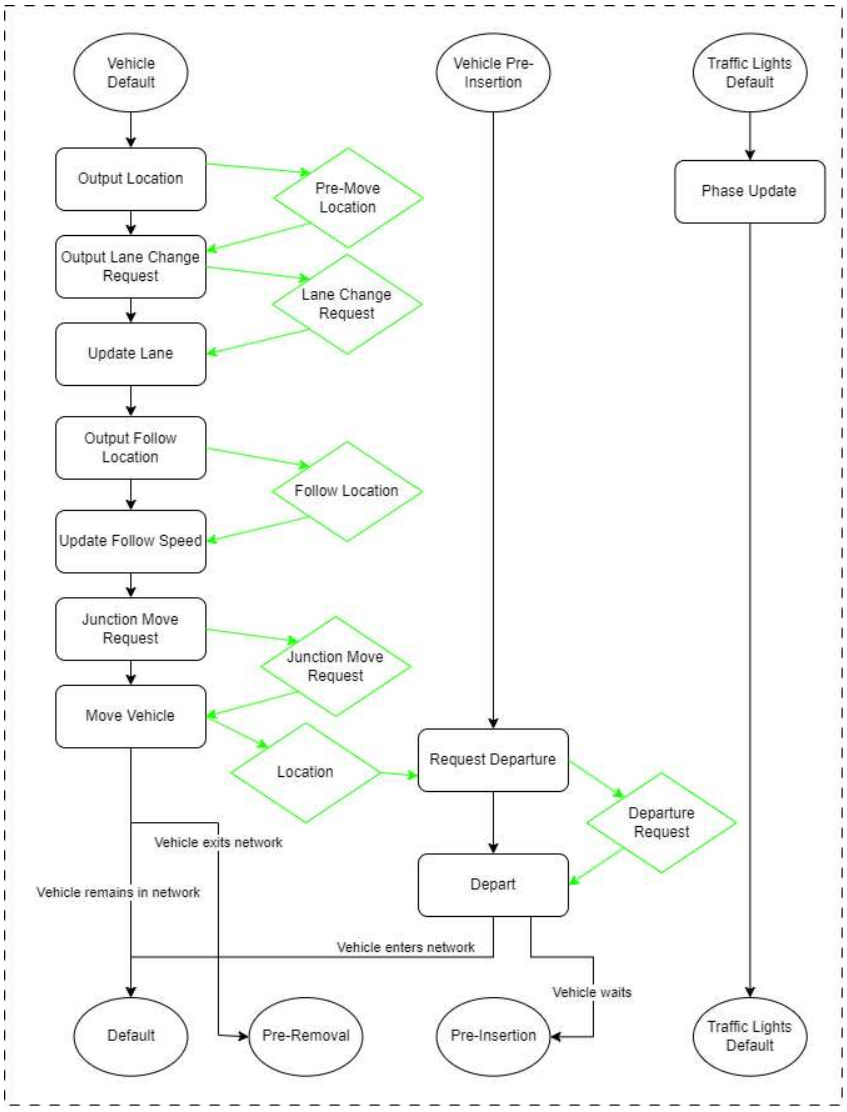


Figure 1. Simplified state diagram of a single iteration of the FLAME-GPU model. Circular nodes represent agent states, boxes represent agent functions responsible for behaviour, and green diagonal boxes represent messages which create execution order dependencies between agents and functions.

While the proposed method, FLAME-GPU, successfully incorporates many of the features found in SUMO, certain functionalities remain unsupported. These include vehicle type restrictions, teleporting mechanisms for resolving deadlocks, and comprehensive speed limit foresight. To ensure a balanced comparison between the two frameworks, these features were deliberately disabled in SUMO during the evaluation process. By aligning the functionalities of both models, this approach provides a fair basis for performance assessment while highlighting the strengths and limitations of FLAME-GPU in handling large-scale traffic simulations.

3.4. Experiment Motivation and Set-Up

The primary objective of this research is to evaluate the computational efficiency of the proposed solution in simulating large-scale, agent-based models and demonstrate its

applicability within the domain of traffic behaviour modelling. To facilitate a fair and consistent comparison across methodologies, the real-time factor (RTF) was employed as the evaluation metric. The RTF measures the ratio of the simulation time to the computation time, where an RTF greater than one indicates a simulation running faster than real time and an RTF less than one signifies a slower pace [55,56]. The experiments were designed to benchmark the performance of the proposed model against the SUMO platform, a widely adopted tool in traffic simulation research, with approximately 11,956 references in the literature attesting to its utility [56].

The analysis was conducted using three experimental configurations: one based on a real-world transportation network and two utilising synthetically generated grid networks. Each configuration provides distinct advantages for evaluating the performance of the proposed method. The real-world transportation network offers a complex and challenging environment, showcasing the model's ability to handle scalability and versatility under realistic conditions. In contrast, the synthetically generated grid networks facilitate controlled experimentation with seamless load scalability, allowing for a systematic assessment of the method's computational efficiency across varying network sizes.

3.4.1. Experiment Configuration: Real-World Network (Isle of Wight, UK)

The first experimental configuration leverages the Isle of Wight's transportation network, depicted in Figure 2a. The map was derived from OpenStreetMap (OSM) data and processed as a left-hand drive network. To prepare the map, the Java OpenStreetMap Editor (JOSM) was used for data cleaning, followed by SUMO's textttnetconvert utility to convert the map into the required `net.xml` format. Traffic flow simulation was generated using SUMO's `randomTrips.py` and `duarouter` utilities.

(**a**) (**b**)

Figure 2. The Isle of Wight road network as modelled in the proposed simulation configuration, alongside a depiction of simulated traffic within the Newport area. (**a**) The Isle of Wight's street network. (**b**) Traffic simulation at various junctions.

To scale the simulation, the vehicle insertion density parameter `-insertion-density` in `randomTrips.py` was varied, starting at 60 vehicles per hour per kilometer of road network length and increasing up to 600. The simulation duration parameter `-e` was incremented in steps of 60 s, starting at 60 s and extending up to 360 s. Additionally, the `-validate` parameter was enabled to account for potential network connectivity issues. Using these configurations, the `duarouter` tool was employed to generate vehicle routes for the simulation.

3.4.2. Experiment Configuration: SUMO-Generated Bidirectional Grid Network

The second experimental configuration used a synthetic network generated with SUMO's `netgenerate` tool. This set-up created a two-dimensional (2D) grid of roads consisting of $N \times N$ rows and columns. Each road segment was bidirectional, and U-turns were disabled. The grid size was scaled incrementally from 5×5 to 100×100, covering

networks ranging from 40 bidirectional road sections with 9 junctions to 19,800 bidirectional roads with 10,000 junctions.

Demand data for the network were generated using SUMO's `randomTrips.py` and `duarouter` utilities to create random, preplanned vehicle routes through the network. The vehicle insertion density was fixed at 300 vehicles per hour per kilometer, with demand generated during the first 360 s of the simulation. This resulted in an average of 30 vehicles per kilometer being introduced into the network. The grid was configured to be multilane using the `netgenerate` suite, specifying a maximum of three lanes per road segment. This produced a network where each road segment had between 1 and 3 lanes.

3.4.3. Experiment Configuration: Fixed-Size Grid Network

The third experimental configuration involved performance measurements conducted on a fixed-size grid network with varying vehicle insertion densities. The grid size was fixed at 100×100, consisting of single-lane roads throughout. The initial vehicle density was varied from 60 vehicles per hour per kilometer to 600, in increments of 60. This configuration allowed for an evaluation of performance under varying levels of demand within a network of a fixed size.

To ensure consistency across all configurations, the simulations were executed three times for one hour. Furthermore, all three configurations were implemented on a workstation equipped with an Intel Core i7-5930K CPU, an NVIDIA GeForce RTX 3090 (24 GiB) GPU, and 64 GB of system memory.

4. Results Analysis

In this section, the results from the three experiment configurations described earlier are presented. Each experiment's outcomes are illustrated in the corresponding figures. To ensure a fair comparison, the same configurations were executed in SUMO for benchmarking purposes.

Figure 3 presents the results of the first experiment configuration, which employed a real-world, complex street network: the Isle of Wight in the United Kingdom. Next, Figure 4 depicts the outcomes from the second experiment configuration, utilising a synthetically generated bidirectional grid network that incrementally increased in size. Finally, Figure 5 showcases the results of the third experiment configuration, which used a fixed-size grid network with varying initial vehicle densities.

In Figure 3a, the real-time factor (RTF) is plotted on a logarithmic scale (y axis) to compare the performance of FLAME-GPU with SUMO. As the number of vehicles in the simulation increased, SUMO showed a significant drop in performance, whereas FLAME-GPU maintained a consistently high RTF. By the end of the simulation, FLAME-GPU achieved an average RTF of around 99, compared with SUMO's 1.45. This demonstrates that FLAME-GPU can handle a larger number of vehicles much faster than SUMO. Similar results can be seen in Figures 4a and 5a.

Figure 3b provides further insight by showing the average simulation time (in seconds) as the number of vehicles increased. For approximately 95,000 vehicles, SUMO took over 2481 s (around 41.35 min), whereas FLAME-GPU completed the simulation in roughly 36 s, achieving a speedup of roughly 68 times. Additionally, SUMO's simulation time increased linearly with the number of vehicles, while FLAME-GPU remained almost constant, highlighting its ability to scale effectively.

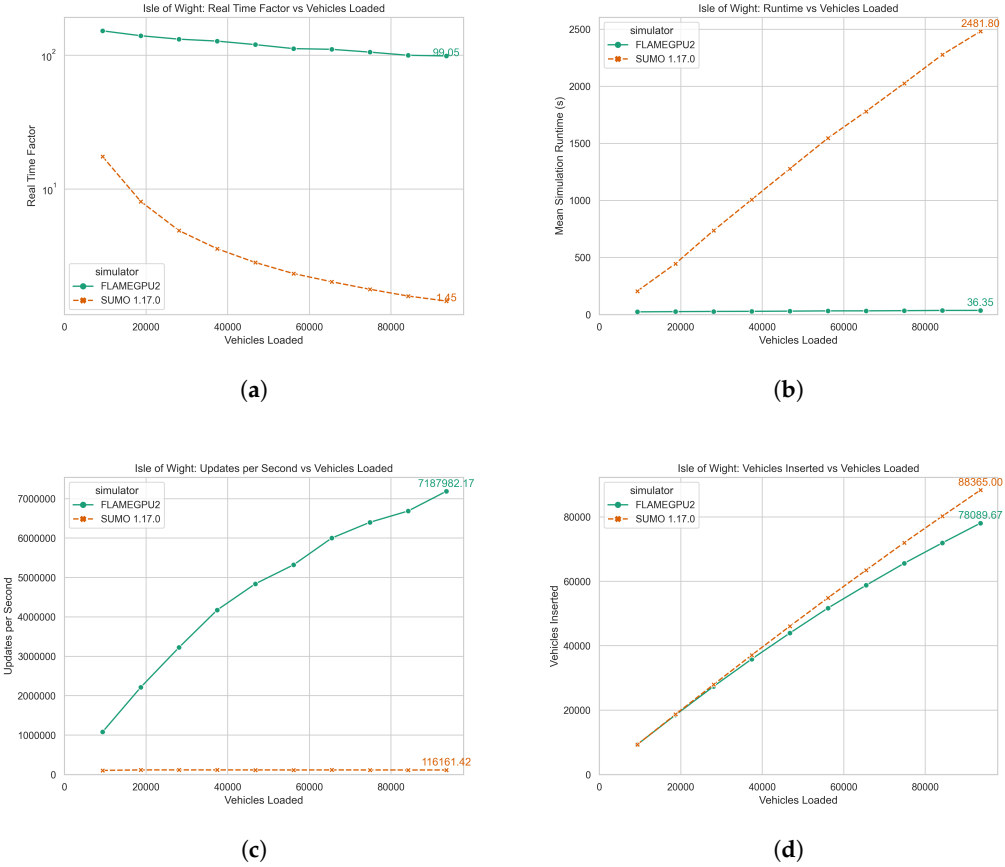


Figure 3. Benchmark results illustrating performance metrics for both the proposed traffic simulation and SUMO conducted on the Isle of Wight. (**a**) Real-time factor compared with vehicles loaded into simulation. (**b**) Average simulation runtime in seconds compared with vehicles loaded. (**c**) Simulation state change per second compared with vehicles loaded. (**d**) Number of vehicles inserted into the model compared with vehicles loaded.

Figure 3c examines the updates per second (UPS)——the number of state changes across all agents per second——as the number of vehicles increased. In FLAME-GPU, the number of updates grew linearly with the vehicle count, while SUMO maintained an average of approximately 116,000 updates per second. At the largest scale (95,000 vehicles), FLAME-GPU performed updates about 62 times faster than SUMO. This trend is also evident in Figures 4c and 5c.

The final benchmark, shown in Figure 3d, evaluated how quickly the simulation could calculate available spots on the street network for inserting new vehicles. In this test, SUMO performed better, demonstrating higher efficiency in loading vehicles into the network. The results also show that FLAME-GPU's insertion rate decreased as the number of vehicles increased. This reduction was likely caused by small differences in how congestion was cleared over time. If vehicles join queues faster than they leave, then these small differences can accumulate, reducing the number of available spots for inserting new vehicles. A similar pattern can be observed in Figures 4d and 5d.

Overall, the results highlight the strengths and limitations of both approaches. SUMO is more efficient at inserting new vehicles into the environment, but FLAME-GPU offers significant improvements in overall performance, particularly for simulations involving large vehicle populations. While SUMO's performance is constrained by its reliance on the CPU, FLAME-GPU benefits from GPU parallelism, allowing it to handle larger, more complex simulations with consistent performance. This approach provides an opportunity

to improve the accuracy and scale of large-scale simulations, without the need to simplify or reduce agent representations.

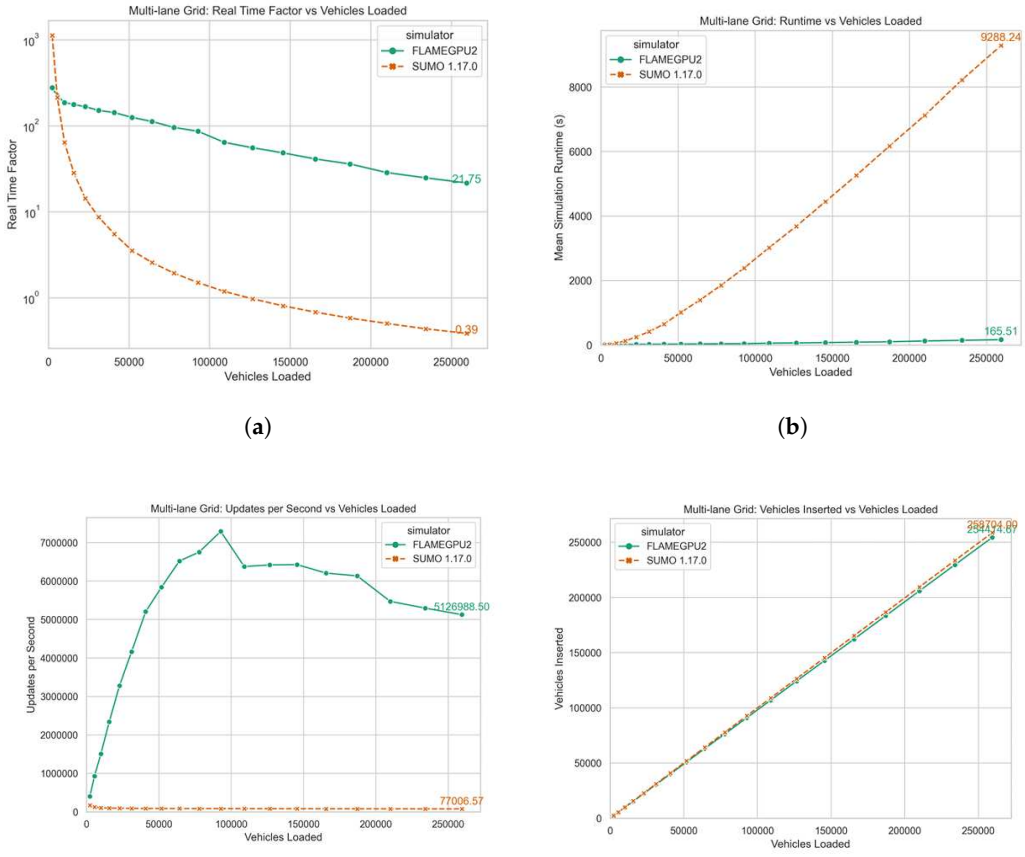


Figure 4. Benchmark results illustrating performance metrics for both the proposed traffic simulation and SUMO, conducted over a synthetic multilane grid. (**a**) Real-time factor compared with vehicles loaded into simulation. (**b**) Average simulation runtime in seconds compared with vehicles loaded. (**c**) Simulation state-change per second compared with vehicles loaded. (**d**) Number of vehicles inserted into the model compared with vehicles loaded.

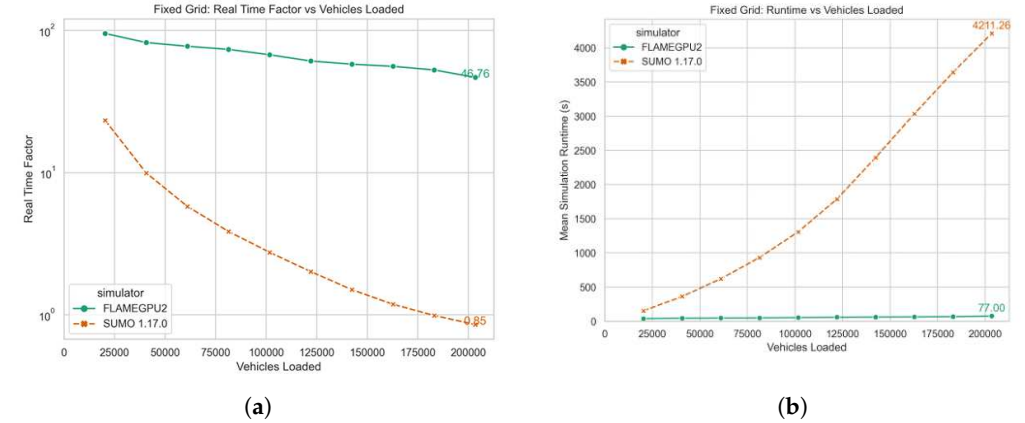
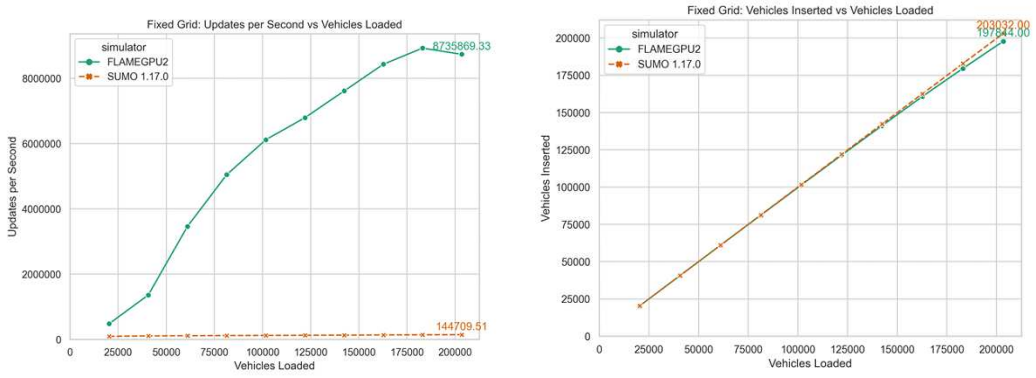


Figure 5. *Cont.*



(**c**) (**d**)

Figure 5. Benchmark results illustrating performance metrics for both the proposed traffic simulation and SUMO, conducted over a synthetic fixed grid. (**a**) Real-time factor compared with vehicles loaded into simulation. (**b**) Factor compared with vehicles loaded into simulation. (**c**) Simulation state-change per second compared with vehicles loaded. (**d**) Number of vehicles inserted into the model compared with vehicles loaded.

5. Conclusions

This article presents a GPU-accelerated framework for agent-based modelling designed to address the computational challenges of simulating large-scale vehicle populations in transport networks. Our contributions are twofold: (1) the utilisation of a scalable modelling framework and (2) its application to simulate vehicle dynamics across different network structures, ranging from real-world street layouts to synthetic grid networks. The results demonstrate significant performance improvements over traditional methods. Comparative experiments with the SUMO transport simulator [57,58] show that FLAMEGPU [18,59] achieved up to 68 times faster simulation speeds for equivalent scenarios. Notably, our model simulated over 95,000 vehicles in just 36 s, compared with SUMO, which required 2481 s. This efficiency enables the simulation of complex, large-scale transport systems, such as the Isle of Wight, within practical timeframes.

The experiments demonstrate that the proposed approach remained highly efficient even as the scale of the environment and vehicle population increased. Despite this growth, our model remained 62 times faster than SUMO, highlighting its adaptability to larger and more complex simulations.

However, certain features available in SUMO, such as vehicle teleportation and adaptive traffic lights, are not currently incorporated into our framework. This may limit the realism and flexibility of the simulation in specific scenarios. Additionally, improvements in handling roundabouts and speed limit transitions could enhance the accuracy of vehicle behaviour representation.

Future work will focus on addressing these limitations by refining the model's fidelity and expanding its capabilities to incorporate emerging transport technologies, such as electric vehicles. These developments will not only improve computational scalability but also provide deeper insights into traffic dynamics and energy consumption, aligning with broader environmental and policy objectives.

Author Contributions: Conceptualization, M.S. and S.v.d.B.; methodology, P.R., R.C., P.H. and M.S.; software, P.R., R.C. and P.H.; validation, S.O., M.S., A.C. and S.v.d.B.; formal analysis, S.O. and M.S.; investigation, S.O. and M.S.; resources, S.K.; data curation, M.S.; writing—original draft preparation, S.O. and M.S.; writing—review and editing, S.O., M.S. and S.v.d.B.; visualization, M.S.; supervision, S.v.d.B. and S.K.; project administration, S.v.d.B.; funding acquisition, S.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: Maxim Smilovitskiy, Sedar Olmez, Alvaro Cabrejas, Sven van den Berghe, and Sachio Kobayashi were employed by the company Fujitsu Research of Europe. Maxim Smilovitskiy, Sedar Olmez, and Sven van den Berghe are applicants for pending patent EP24162384.2, which covers the technical process of application of a described framework to a commercial use case scenario, with emphasis on the practical aspects and specific implementations of the general principles described in this paper. Paul Richmond, Robert Chisholm, and Peter Heywood are employed by the University of Sheffield. Paul Richmond, Robert Chisholm, and Peter Heywood have received research grants to work on FLAME GPU, which are available at https://public.tableau.com/app/profile/epsrcdatateam/viz/EPSRCFundingApplicationOutcomes/MeetingList (accessed on 13 May 2025). via Orchid on the web.

References

1. Smilovitskiy, M.; Olmez, S.; Richmond, P.; Chisholm, R.; Heywood, P.; Cabrejas, A.; van den Berghe, S.; Kobayashi, S. Overcoming Computational Complexity: A Scalable Agent-Based Model of Traffic Activity Using FLAME-GPU. In *Advances in Practical Applications of Agents, Multi-Agent Systems, and Digital Twins: The PAAMS Collection*; Mathieu, P., De la Prieta, F., Eds.; Springer Nature: Cham, Switzerland, 2025; pp. 240–251.
2. Ge, J. Endogenous rise and collapse of housing price: An agent-based model of the housing market. *Comput. Environ. Urban Syst.* **2017**, *62*, 182–198. [CrossRef]
3. Axtell, R.; Farmer, D.; Geanakoplos, J.; Howitt, P.; Carrella, E.; Conlee, B.; Goldstein, J.; Hendrey, M.; Kalikman, P.; Masad, D.; et al. An agent-based model of the housing market bubble in metropolitan Washington, DC. In Proceedings of the Housing Markets and the Macroeconomy: Challenges for Monetary Policy and Financial Stability, Deutsche Bundesbank, Frankfurt am Main, Germany, 6 May–6 June 2014; pp. 5–6.
4. Geanakoplos, J.; Axtell, R.; Farmer, J.D.; Howitt, P.; Conlee, B.; Goldstein, J.; Hendrey, M.; Palmer, N.M.; Yang, C.Y. Getting at Systemic Risk via an Agent-Based Model of the Housing Market. *Am. Econ. Rev.* **2012**, *102*, 53–58. [CrossRef]
5. Manley, E.; Cheng, T.; Penn, A.; Emmonds, A. A framework for simulating large-scale complex urban traffic dynamics through hybrid agent-based modelling. *Comput. Environ. Urban Syst.* **2014**, *44*, 27–36. [CrossRef]
6. Haman, I.T.; Kamla, V.C.; Galland, S.; Kamgang, J.C. Towards an multilevel agent-based model for traffic simulation. *Procedia Comput. Sci.* **2017**, *109*, 887–892. [CrossRef]
7. Hager, K.; Rauh, J.; Rid, W. Agent-based modeling of traffic behavior in growing metropolitan areas. *Transp. Res. Procedia* **2015**, *10*, 306–315. [CrossRef]
8. Zhang, B.; DeAngelis, D.L. An overview of agent-based models in plant biology and ecology. *Ann. Bot.* **2020**, *126*, 539–557. [CrossRef]
9. Filatova, T.; Verburg, P.H.; Parker, D.C.; Stannard, C.A. Spatial agent-based models for socio-ecological systems: Challenges and prospects. *Environ. Model. Softw.* **2013**, *45*, 1–7. [CrossRef]
10. McLane, A.J.; Semeniuk, C.; McDermid, G.J.; Marceau, D.J. The role of agent-based models in wildlife ecology and management. *Ecol. Model.* **2011**, *222*, 1544–1556. [CrossRef]
11. Cornelius, C.V.; Lynch, C.J.; Gore, R. Aging out of crime: Exploring the relationship between age and crime with agent based modeling. In Proceedings of the Agent-Directed Simulation Symposium, Virginia Beach, VA, USA, 23–26 April 2017; pp. 1–12.
12. Olmez, S.; Birks, D.; Heppenstall, A.; Ge, J. Learning the rational choice perspective: A reinforcement learning approach to simulating offender behaviours in criminological agent-based models. *Comput. Environ. Urban Syst.* **2024**, *112*, 102141. [CrossRef]
13. Malleson, N.; Heppenstall, A.; See, L. Crime reduction through simulation: An agent-based model of burglary. *Comput. Environ. Urban Syst.* **2010**, *34*, 236–250. [CrossRef]
14. Gilbert, N.; Troitzsch, K. *Simulation for the Social Scientist*; McGraw-Hill Education: Luton, UK, 2005.
15. Railsback, S.F.; Grimm, V. *Agent-Based and Individual-Based Modeling: A Practical Introduction*; Princeton University Press: Princeton, NJ, USA, 2019.
16. An, L. Modeling human decisions in coupled human and natural systems: Review of agent-based models. *Ecol. Model.* **2012**, *229*, 25–36. [CrossRef]
17. Heppenstall, A.J.; Crooks, A.T.; See, L.M.; Batty, M. *Agent-Based Models of Geographical Systems*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2011.

18. Richmond, P.; Chimeh, M.K. FLAME GPU: Complex system simulation framework. In Proceedings of the 2017 International Conference on High Performance Computing & Simulation (HPCS), Genoa, Italy, 17–21 July 2017; pp. 11–17.
19. Heywood, P.; Richmond, P.; Maddock, S. Road network simulation using FLAME GPU. In Proceedings of the Euro-Par 2015: Parallel Processing Workshops: Euro-Par 2015 International Workshops, Vienna, Austria, 24–25 August 2015; Revised Selected Papers 21; Springer: Berlin/Heidelberg, Germany, 2015; pp. 430–441.
20. Heywood, P.; Maddock, S.; Casas, J.; Garcia, D.; Brackstone, M.; Richmond, P. Data-parallel agent-based microscopic road network simulation using graphics processing units. *Simul. Model. Pract. Theory* **2018**, *83*, 188–200. [CrossRef]
21. de Paiva Oliveira, A.; Richmond, P. Feasibility study of multi-agent simulation at the cellular level with flame gpu. In Proceedings of the Twenty-Ninth International Flairs Conference, Key Largo, FL, USA, 16–18 May 2016.
22. Alqurashi, R.; Altman, T. Hierarchical agent-based modeling for improved traffic routing. *Appl. Sci.* **2019**, *9*, 4376. [CrossRef]
23. Zhao, B.; Kumar, K.; Casey, G.; Soga, K. Agent-based model (ABM) for city-scale traffic simulation: A case study on San Francisco. In *International Conference on Smart Infrastructure and Construction 2019 (ICSIC) Driving Data-Informed Decision-Making*; ICE Publishing: Lancashire, UK, 2019; pp. 203–212.
24. Olmez, S.; Douglas-Mann, L.; Manley, E.; Suchak, K.; Heppenstall, A.; Birks, D.; Whipp, A. Exploring the Impact of Driver Adherence to Speed Limits and the Interdependence of Roadside Collisions in an Urban Environment: An Agent-Based Modelling Approach. *Appl. Sci.* **2021**, *11*, 5336. [CrossRef]
25. Bieker, L.; Krajzewicz, D.; Morra, A.; Michelacci, C.; Cartolano, F. Traffic simulation for all: A real world traffic scenario from the city of Bologna. In Proceedings of the Modeling Mobility with Open Data: 2nd SUMO Conference 2014, Berlin, Germany, 15–16 May 2014; pp. 47–60.
26. Wallentin, G.; Loidl, M. Agent-based bicycle traffic model for Salzburg city. *GI_Forum J. Geogr. Inf. Sci.* **2015**, *2015*, 558–566. [CrossRef]
27. Casas, J.; Ferrer, J.L.; Garcia, D.; Perarnau, J.; Torday, A. Traffic simulation with aimsun. *Fundam. Traffic Simul.* **2010**, *145*, 173–232.
28. Quaassdorff, C.; Borge, R.; Pérez, J.; Lumbreras, J.; de la Paz, D.; de Andrés, J.M. Microscale traffic simulation and emission estimation in a heavily trafficked roundabout in Madrid (Spain). *Sci. Total. Environ.* **2016**, *566*, 416–427. [CrossRef]
29. Sun, Z.; Lorscheid, I.; Millington, J.D.; Lauf, S.; Magliocca, N.R.; Groeneveld, J.; Balbi, S.; Nolzen, H.; Müller, B.; Schulze, J.; et al. Simple or complicated agent-based models? A complicated issue. *Environ. Model. Softw.* **2016**, *86*, 56–67. [CrossRef]
30. Rhodes, D.M.; Holcombe, M.; Qwarnstrom, E.E. Reducing complexity in an agent based reaction model—benefits and limitations of simplifications in relation to run time and system level output. *Biosystems* **2016**, *147*, 21–27. [CrossRef]
31. Lopez, P.A.; Behrisch, M.; Bieker-Walz, L.; Erdmann, J.; Flötteröd, Y.P.; Hilbrich, R.; Lücken, L.; Rummel, J.; Wagner, P.; Wießner, E. Microscopic traffic simulation using sumo. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 2575–2582.
32. Mecheva, T.; Furnadzhiev, R.; Kakanakov, N. Modeling driver behavior in road traffic simulation. *Sensors* **2022**, *22*, 9801. [CrossRef]
33. Krauß, S.; Wagner, P.; Gawron, C. Metastable states in a microscopic model of traffic flow. *Phys. Rev. E* **1997**, *55*, 5597. [CrossRef]
34. Shamim Akhter, M.; Quaderi, S.J.S.; Al Forhad, M.A.; Sumit, S.H.; Rahman, M.R. A SUMO based simulation framework for intelligent traffic management system. *J. Traffic Logist. Eng.* **2020**. [CrossRef]
35. Ma, X.; Hu, X.; Weber, T.; Schramm, D. Evaluation of accuracy of traffic flow generation in SUMO. *Appl. Sci.* **2021**, *11*, 2584. [CrossRef]
36. Barthauer, M.; Hafner, A. Coupling traffic and driving simulation: Taking advantage of SUMO and SILAB together. *EPiC Ser. Eng.* **2018**, *2*, 56–66.
37. Alghamdi, T.; Mostafi, S.; Abdelkader, G.; Elgazzar, K. A comparative study on traffic modeling techniques for predicting and simulating traffic behavior. *Future Internet* **2022**, *14*, 294. [CrossRef]
38. BUBER, E.; DIRI, B. Performance Analysis and CPU vs GPU Comparison for Deep Learning. In Proceedings of the 2018 6th International Conference on Control Engineering & Information Technology (CEIT), Istanbul, Turkey, 25–27 October 2018; pp. 1–6. [CrossRef]
39. Zhang, H.; Feng, S.; Liu, C.; Ding, Y.; Zhu, Y.; Zhou, Z.; Zhang, W.; Yu, Y.; Jin, H.; Li, Z. Cityflow: A multi-agent reinforcement learning environment for large scale city traffic scenario. In Proceedings of the World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; pp. 3620–3624.
40. Vu, V.A.; Tan, G. A Framework for Mesoscopic Traffic Simulation in GPU. *IEEE Trans. Parallel Distrib. Syst.* **2019**, *30*, 1691–1703. [CrossRef]
41. Vu, V.A.; Tan, G. High-performance mesoscopic traffic simulation with GPU for large scale networks. In Proceedings of the 2017 IEEE/ACM 21st International Symposium on Distributed Simulation and Real Time Applications (DS-RT), Rome, Italy, 18–20 October 2017; pp. 1–9. [CrossRef]
42. Hirabayashi, M.; Kato, S.; Edahiro, M.; Sugiyama, Y. Toward GPU-accelerated traffic simulation and its real-time challenge. *Reaction* **2012**.

43. Rajf, D.; Potuzak, T. Comparison of Road Traffic Simulation Speed on CPU and GPU. In Proceedings of the 2019 IEEE/ACM 23rd International Symposium on Distributed Simulation and Real Time Applications (DS-RT), Cosenza, Italy, 7–9 October 2019; pp. 1–8. [CrossRef]
44. Wang, K.; Shen, Z. A GPU based trafficparallel simulation module of artificial transportation systems. In Proceedings of the 2012 IEEE International Conference on Service Operations and Logistics, and Informatics, Suzhou, China, 8–10 July 2012; pp. 160–165. [CrossRef]
45. Xu, Y.; Tan, G.; Li, X.; Song, X. Mesoscopic traffic simulation on CPU/GPU. In Proceedings of the 2nd ACM SIGSIM Conference on Principles of Advanced Discrete Simulation, Atlanta, GA, USA, 24–26 June 2014; pp. 39–50. [CrossRef]
46. Shen, Z.; Wang, K.; Zhu, F. Agent-based traffic simulation and traffic signal timing optimization with GPU. In Proceedings of the 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), Washington, DC, USA, 5–7 October 2011; pp. 145–150. [CrossRef]
47. Strippgen, D.; Nagel, K. Multi-agent traffic simulation with CUDA. In Proceedings of the 2009 International Conference on High Performance Computing & Simulation, Leipzig, Germany, 21–24 June 2009; pp. 106–114. [CrossRef]
48. Xiao, J.; Andelfinger, P.; Eckhoff, D.; Cai, W.; Knoll, A. Exploring Execution Schemes for Agent-Based Traffic Simulation on Heterogeneous Hardware. In Proceedings of the 2018 IEEE/ACM 22nd International Symposium on Distributed Simulation and Real Time Applications (DS-RT), Madrid, Spain, 15–17 October 2018; pp. 1–10. [CrossRef]
49. Shirvani, M.; Kesserwani, G.; Richmond, P. Agent-based simulator of dynamic flood-people interactions. *J. Flood Risk Manag.* **2021**, *14*, e12695. [CrossRef]
50. Richmond, P.; Walker, D.; Coakley, S.; Romano, D. High performance cellular level agent-based simulation with FLAME for the GPU. *Briefings Bioinform.* **2010**, *11*, 334–347. [CrossRef]
51. Krauss, S. Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics. Ph.D. Thesis, DLR Deutsches Zentrum fuer Luft- und Raumfahrt e.V., Koeln (Germany). Abt. Unternehmensorganisation und -information; Koeln Univ. (Germany). Mathematisch-Naturwissenschaftliche Fakultaet, Köln, Germany, 1999.
52. Tan, F.; Wei, D.; Zhu, J.; Xu, D.; Yin, K. An aggressive car-following model in the view of driving style. *Can. J. Civ. Eng.* **2017**, *44*, 775–782. [CrossRef]
53. Erdmann, J. SUMO's Lane-Changing Model. In *Modeling Mobility with Open Data, Proceedings of the 2nd SUMO Conference 2014 Berlin, Germany, 15–16 May 2014*; Behrisch, M., Weber, M., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 105–123.
54. Erdmann, J.; Krajzewicz, D. SUMO's Road Intersection Model. In *Simulation of Urban Mobility*; Behrisch, M., Krajzewicz, D., Weber, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2014; pp. 3–17.
55. Pratap, V.; Xu, Q.; Kahn, J.; Avidov, G.; Likhomanenko, T.; Hannun, A.; Liptchinsky, V.; Synnaeve, G.; Collobert, R. Scaling up online speech recognition using convnets. *arXiv* **2020**, arXiv:2001.09727.
56. Krajzewicz, D. Traffic simulation with SUMO–simulation of urban mobility. *Fundam. Traffic Simul.* **2010**, *145*, 269–293.
57. Behrisch, M.; Bieker, L.; Erdmann, J.; Krajzewicz, D. SUMO–simulation of urban mobility: An overview. In Proceedings of the SIMUL 2011, The Third International Conference on Advances in System Simulation, Barcelona, Spain, 23–29 October 2011.
58. Krajzewicz, D.; Hertkorn, G.; Rössel, C.; Wagner, P. SUMO (Simulation of Urban MObility)-an open-source traffic simulation. In Proceedings of the 4th Middle East Symposium on Simulation and Modelling (MESM20002), Sharjah, United Arab Emirates, 28–30 September 2002; pp. 183–187.
59. Richmond, P.; Chisholm, R.; Heywood, P.; Chimeh, M.K.; Leach, M. FLAME GPU 2: A framework for flexible and performant agent based simulation on GPUs. *Softw. Pract. Exp.* **2023**, *53*, 1659–1680. [CrossRef]