Proceedings Paper:
Smilovitskiy, M., Olmez, S., Richmond, P. orcid.org/0000-0002-4657-5518 et al. (5 more authors) (2024) Overcoming computational complexity: a scalable agent-based model of traffic activity using FLAME-GPU. In: Mathieu, P. and De la Prieta, F., (eds.) Advances in Practical Applications of Agents, Multi-Agent Systems, and Digital Twins: The PAAMS Collection. 22nd International Conference, PAAMS 2024, 26-28 Jun 2024, Salamanca, Spain. Lecture Notes in Computer Science, 15157 . Springer Nature Switzerland , pp. 240-251. ISBN 9783031704147

https://doi.org/10.1007/978-3-031-70415-4_21

Overcoming Computational Complexity: A Scalable Agent-Based Model of Traffic Activity using FLAME-GPU

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, 2nd Floor (West), The Urban Building, 3-9 Albert Street, Slough, SL1 2BE, England, UK
[2] Department of Computer Science, University of Sheffield, Regent Court (DCS), 211 Portobello, Sheffield, S1 4DP

Abstract. Agent-based modelling (ABM) has significantly advanced the simulation of complex systems across various disciplines, including economic markets, traffic systems, and ecological research. By representing individuals as autonomous agents operating within a defined environment, ABMs facilitate the exploration of system behaviours and the testing of interventions in a controlled setting. However, the computational demands of ABMs, particularly in simulating large-scale systems, pose significant challenges. This article addresses these challenges by introducing a GPU-accelerated transport model for the Isle of Wight, leveraging the FLAME-GPU framework to enhance scalability and efficiency in traffic simulation. By comparing the proposed model with traditional, non-GPU accelerated simulations, such as SUMO, the study demonstrates improved performance metrics, including simulation speed and the ability to handle larger vehicle populations effectively. This contribution not only showcases the potential of GPU acceleration in overcoming computational constraints of ABMs but also provides a practical framework for simulating transport systems at a more granular and extensive scale than previously possible. Through detailed experimentation, the article illustrates the model's capability to realistically simulate the vehicle population of the Isle of Wight, aiming for a balance between computational tractability and the accurate representation of complex traffic interactions.

Keywords: Agent-Based Model · FLAME GPU · Traffic Simulation · Individual-Based Model · Data Analysis

1 Introduction

Agent-based modelling (ABM) has emerged as a pivotal tool for simulating complex systems across various domains, enabling researchers to explore the dynamics of markets [10, 2, 11], traffic systems [23, 14, 13], ecological systems [35, 9, 24], and more. ABM encapsulates interactions between autonomous agents and their environments, facilitating the examination of individual and collective behaviours within a controlled computational setting.

Despite ABM's versatility, its application, particularly in simulating large-scale phenomena, is often hampered by computational constraints. Simplifications or reductions in agent populations are common workarounds, though they may compromise the model's fidelity and outcomes [12, 29, 1, 15]. This article addresses these computational challenges within the transportation domain by leveraging the computational resources of graphics processing units (GPUs) to simulate traffic on the Isle of Wight, showcasing the potential for GPU-accelerated ABM in capturing complex transport dynamics.

Our contributions to the field include:

- A scalable traffic simulation using the FLAME-GPU (Flexible Large-scale Agent Modelling Environment) framework [30], demonstrating the capability to model large volumes of vehicular traffic.
- A detailed simulation of the Isle of Wight's transport system, integrating real-world geographic and traffic data to validate the model's efficacy.

Previous studies have explored FLAME-GPU's application in transport [17, 16] and biological systems [26], underscoring the architecture's scalability and computational efficiency. However, these works often employ simplified models or do not address the complexities of real-world networks such as the Isle of Wight's. This gap highlights the necessity for a general-purpose, scalable ABM transport model, which we address by implementing a detailed simulation informed by established vehicle behaviour models and empirical data.

Our experiments focus on the Isle of Wight's road 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 to benchmark our model's performance enhancements in a GPU-accelerated environment.

The remainder of this article is structured as follows: Section 2 reviews existing transport models and identifies gaps our work aims to fill. Section 3 outlines the FLAME-GPU framework and our model's architecture. Section 4 presents our experimental findings, and Section 5 concludes with an assessment of our 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 vehicular dynamics within street networks, offering profound insights into individual-level decision-making and its ramifications on traffic flow and congestion. Despite the contributions, the approach encounters hurdles regarding computational demands, scalability, and the depiction of geographic realities. Notable efforts have integrated empirical city data to refine traffic signal controls [4], explored multimodal transportation navigation [34, 6, 25], and sought to estimate and optimise emissions [28].

Addressing computational complexity often involves partitioning extensive simulation areas into manageable sections, facilitating higher real-time factor simulations. The calibration and validation of ABMs pose significant challenges, impacting the models' accuracy. Balancing empirical validity and simplification is crucial for realistic yet computationally feasible ABMs.

2.2 Road transportation and the role of SUMO

SUMO (Simulation of Urban MObility) [22] is a prominent open-source software 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. Utilising microscopic car-following models, SUMO simulates individual vehicle movements and interactions based on dynamic road conditions and traffic regulations, employing a modified version of Krauss's car-following model [21] by default.

While SUMO's open-source nature and detailed simulation capabilities are advantageous, the steep learning curve, limited agent behaviour modification, and notably, its serial CPU-based processing model restrict scalability and performance. High-performance computation comparisons highlight GPUs' superiority over CPUs in numerical tasks due to their parallel processing architecture [5]. Transitioning from serial to parallel processing models to exploit GPU capabilities entails substantial software engineering, posing integration challenges with established frameworks like SUMO.

The quest for enhanced simulation performance has led to the development of custom frameworks like CityFlow [36], which claims up to 25 times faster simulations than SUMO for extensive agent-based networks. Despite such improvements, the need for GPU-accelerated solutions for even larger and more complex network simulations remains unmet, underscoring the potential benefits of parallel computing in traffic simulation.

Our research aims to bridge this gap by leveraging GPU-enhanced ABM within a microsimulation traffic model, proposing a scalable and complex traffic behaviour simulation framework. By integrating ABM with microsimulation 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 capabilities of SUMO and similar tools.

3 Methodology

3.1 FLAME-GPU

FLAME-GPU epitomises a high-performance agent-based simulation framework, leveraging the parallel computing capabilities of modern GPUs to enhance the simulation of intricate systems. This framework abstracts GPU complexities, enabling modellers to focus on the conceptual design rather than the computational intricacies. Such separation ensures the distinct representation of models from their execution, facilitating the construction and simulation of expansive models in feasible timeframes. FLAME-GPU's versatility extends across various domains, supporting simulations ranging from pedestrian dynamics [33], through road networks [17], to cellular biological systems [32]. Employing FLAME-GPU requires mapping the system under study to an agent-centric paradigm, where agents embody entities with inherent states, messages facilitate indirect interactions among agents through a global pool, and the environment encapsulates globally accessible data.

3.2 Traffic simulation model overview

This section delineates our 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 authentic road attributes like lanes, intersections, and traffic signals.

The core of our vehicle behaviour model modifies the Krauss car-following paradigm, also foundational to SUMO's simulation mechanics [20]. 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 driver reaction time and $\tau_b = \frac{\overline{v}}{b(\overline{v})}$ representing braking time, influenced by average velocity $\overline{v}$ and random perturbation η to model deviations from ideal driving.

Lane-changing and intersection navigation are guided by principles outlined in [7] and [8], with minor technical alterations in implementation that do not affect principle mechanics, but only serve the purpose of better fitting into FLAME-GPU architecture.

A system of equations from (1), lane-changing and intersection-crossing rules, as well as other rules necessary to update the state of the system, are expressed as a series of agent functions. Agent variables necessary to execute the functions are stored and exchanged through a global messaging pool - a feature facilitated by

FLAME-GPU's design. The request for a state update of the system is fulfilled by launching function kernels for all agents in the agent-state population. Each agent is represented by a thread, and as a kernel is launched, a grid of threads is formed and is subsequently combined into blocks which are then assigned for concurrent execution to available Streaming Multiprocessors (SMs) within a GPU. In case the number of threads n exceeds a GPU's maximum thread capacity N_{max}, it then takes roughly $c = n\%N_{max}$ steps to complete each kernel run. While this scheme does not fundamentally change the time complexity of the algorithm, it allows for a smooth and efficient computational scaling process.

Figure 1 visualises this simulation pipeline. Note that the current implementation of the traffic simulator accepts the same input files as SUMO, which makes for straightforward testing and validation.

3.3 Mapping traffic model to FLAME-GPU

In order to translate the transportation model in to the FLAME-GPU framework, a network based messaging approach was adopted for the bulk of agent communication. This approach allows agents to exist within a static multi-lane network, using the network structure to query agents within the same or adjacent edges or lanes. Both a compressed spare row (CSR) and compressed spare column (CSC) representation of the network were stored within the agent environment allowing for efficient querying of both upstream and downstream edges. Network lookups are used for calculations to find followers and leaders, essential components of traversing the network and entering junctions. Elsewhere network communication is used within vehicle following, lane changing and vehicle insertion (into the network). Within the FLAME-GPU implementation, vehicles make dynamic choices of lanes but use pre-computed routes to reach their destinations. The equivalent behaviour is observed from within SUMO by using equivalent route files generated from SUMOs routing tool.

The FLAME-GPU model uses agents to represent vehicles. The SUMO model parameters and distributions are mapped to FLAME-GPU agent variables. Vehicle agents have three distinct states, the default state is "driving" in which agents will perform call following, lane changing and traversal through junctions. Vehicles also utilise states for pre-insertion into the network and pre-removal. The former uses gap acceptance to ensure vehicles can safely enter a road edge, the latter allows agents to persist within the model for data collection. Agents are also used to represent sections of the road network. This facilitates data aggregation and collection. Additionally, agents are used to represent traffic lights which use a fixed pattern within the FLAME-GPU model.

While our FLAME-GPU model incorporates many of SUMO's advanced features, certain functionalities like vehicle type restrictions, teleporting for deadlock resolution, and comprehensive speed limit foresight are not currently supported. These limitations are addressed by disabling corresponding features in SUMO for a balanced comparison.

3.4 Experiment motivation and setup

The objective of our research is to elucidate the computational efficacy of our proposed solution in simulating extensive, agent-based models with a focus on vehicular traffic. We utilise the real-time factor (RTF) for evaluation, a metric denoting the ratio of simulated time to computation time, where an RTF above one suggests a simulation pace surpassing real-time [27, 18]. Our experiments set out to benchmark our model's performance against the SUMO platform, selected for its prevalent application in traffic simulation studies, with roughly 11,956 references in the literature [18].

Our empirical analysis utilised the Isle of Wight's transport network, shown in Figure 3a. Contrary to the use of synthetic networks, which are characterised by their uniformity, real-world networks offer a heterogeneous and challenging environment, highlighting our model's scalability and versatility.

The map was generated from Open Street Map data, as a left hand drive network. JOSM was used to clean the map, and SUMO's netconvert utility was used to convert the map to required net.xml format. For simulating traffic flow, we employed SUMO's randomTrips.py and duarouter utilities. To scale the simulation in randomTrips.py we varied the vehicle insertion density parameter –insertion-density starting from 60 and going up to 600 vehicles per hour per kilometer of road network's length, with parameter -e being varied in increments of 60, starting from 60 and going up to 360 seconds. We also used –validate parameter as network may be not fully connected. Following this procedure, duarouter was used to produce routes. To ensure data reliability, each simulation was executed three times, spanning from timestep 0 to 3600 seconds, equating to one hour. Please note, however, that upper end density settings produce thousands of teleportation events due to congestion and multiple collisions. As a caveat, it requires mentioning, that currently SUMO has a more elaborate algorithm of inserting teleported (gridlocked) cars into downstream traffic. This may account for some additional performance gains that FLAME traffic simulator may have in test results.

The computational experiments were conducted 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 Figure 2, we present four sub-figures comparing our model to SUMO. In the following paragraph, we break down each graph and describe the results.

In Figure 2a, the RTF is plotted on a log-scale (y-axis) and compares our model (FLAME-GPU) with SUMO. As the number of vehicles increases in the simulation, SUMO exhibits a decline in performance, while FLAME maintains a high RTF, achieving an average of approximately 99 compared to SUMO's 1.45 at the end of the simulation. These results indicate that our solution can simulate a larger population of vehicles compared to SUMO in a significantly shorter time.

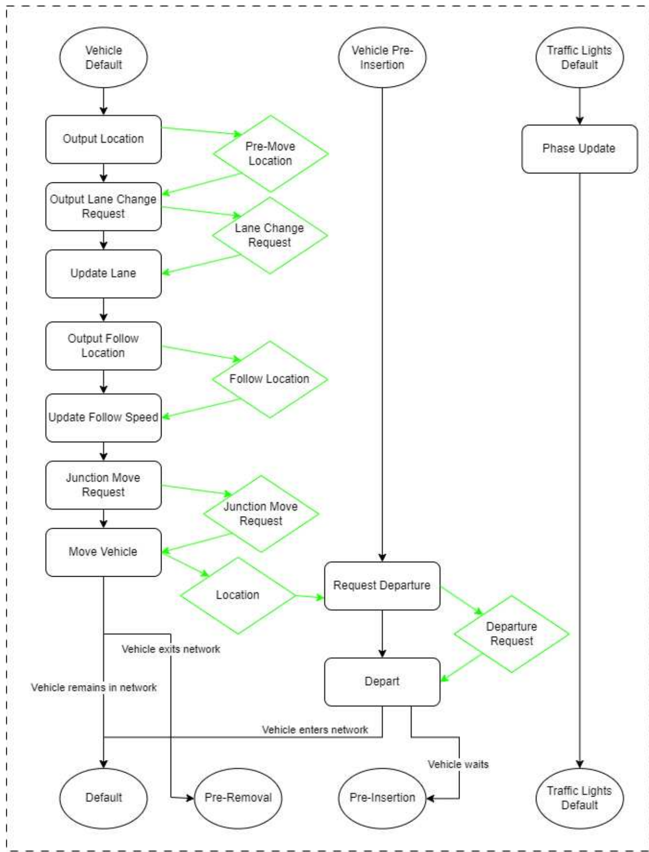
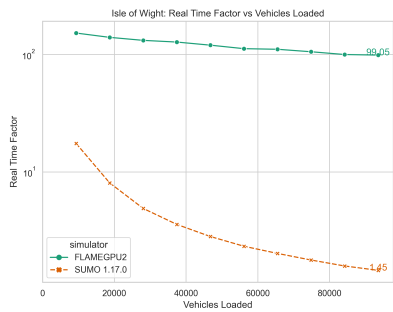
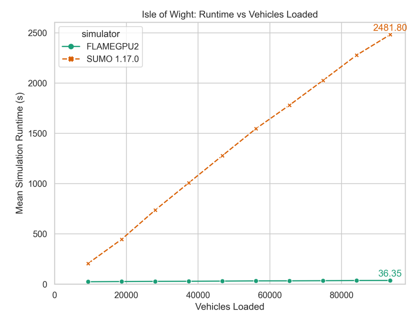


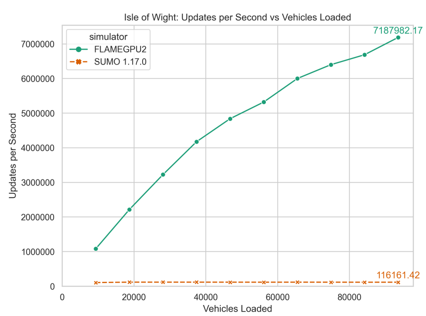
Fig. 1: Simplified state diagram of a single iteration of the FLAME-GPU model. Circular nodes represent agent states, boxes represent agent functions responsible for behaviour and green diagonal boxes represent messages which create execution order dependencies between agents and functions.



(a) Real-time factor compared to the number of vehicles loaded.



(b) Average simulation run-time in seconds compared to the number of vehicles loaded.



(c) Simulation state-change per second compared to the number of vehicles loaded.

(d) Number of vehicles inserted into the model compared to the number of vehicles loaded.

Fig. 2: Benchmark results illustrating performance metrics for both the proposed traffic simulation and SUMO, conducted on the Isle of Wight.

To supplement these findings, Figure 2b displays the average simulation time in seconds as the number of vehicles increases. While SUMO takes over 2,481 seconds (41.35 minutes) to simulate approximately 95,000 vehicles, our proposal completes the simulation in just around 36 seconds, representing a speedup of approximately 68 times. Furthermore, SUMO's simulation time increases linearly with the number of vehicles, while our model remains relatively consistent.

The interactions between agents and their environment can be computationally intensive as the number of agents increases. In Figure 2c, we compare the updates per second (the number of state-changes across all agents every second) with the number of vehicles in the model. The interactions grow linearly with the number of vehicles in our solution, while SUMO maintains an average of around 116,000 updates per second. The final data point, with 95,000 vehicles loaded, shows FLAME-GPU performing vehicle updates approximately 62 times faster than SUMO.

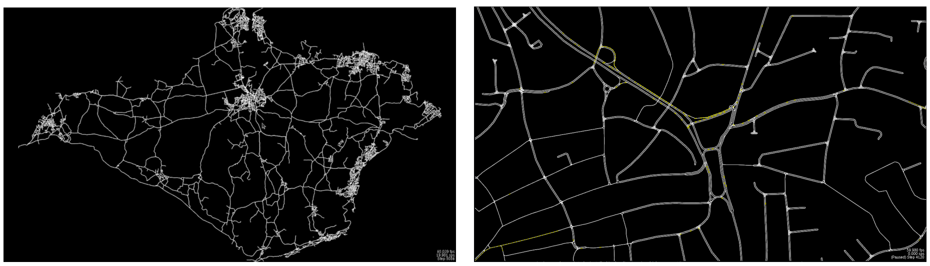
The last benchmark test in Figure 2d evaluates how quickly the model can compute available spots on the street network to insert a new vehicle. SUMO appears to outperform our solution, indicating that SUMO is more efficient in loading new vehicles into street networks. The graph reveals that FLAME-GPU's insertion rate decreases as the number of vehicles increases, partly due to small differences in congestion clearance compounding over time. If vehicles join a queue faster than they leave, even minor differences in clearance rate can quickly accumulate, leading to fewer available slots for new vehicles to insert in the model with the slower clearance rate.

These results demonstrate the performance advantages of our solution over SUMO and vice versa. Algorithmic efficiency in loading new agents into an environment is an area where SUMO excels. However, in terms of raw performance (the time taken to execute a model from timestep 0 to N), simulations executed using GPU can be significantly faster than those primarily utilising the Central Processing Unit (CPU). It is worth noting that some modellers may reduce the granularity of agent representation as computational complexity increases, making the model less representative of the complex system being simulated. This is a challenge that most agent-based modellers and simulation experts must contend with. Hopefully, our contribution to the literature can empower researchers to leverage GPU-based modelling frameworks to develop more accurate and reflective models of the real world. In the following section, we revisit our findings and discuss future avenues for further exploration while assessing the strengths and weaknesses of our solution.

5 Conclusion

This article introduces a GPU-enhanced framework for agent-based modelling, specifically tailored to address the computational challenges associated with simulating extensive agent populations in the transport domain. Our contributions are twofold: (1) the development of a scalable modelling framework, and (2) the application of this framework to simulate vehicle dynamics on the Isle of

Wight, demonstrating significant performance advantages over traditional methods. Comparative experiments against the SUMO transport simulator [3, 19] underscore our framework's efficiency, with FLAME-GPU [31, 30] achieving a 68x faster simulation rate for equivalent scenarios. Most notably, our model facilitated the simulation of over 95,000 vehicles within 36 seconds, a stark contrast to SUMO's 2,481 seconds, enabling the representation of realistic agent populations and complex environments.



(a) The Isle of Wight's street network (b) Traffic simulation at various junctions

Fig. 3: The Isle of Wight road network as modelled in our simulation alongside a depiction of simulated traffic within the Newport area.

Despite its successes, our model does not incorporate certain features present in SUMO, such as vehicle teleportation and adaptive traffic lights, which may limit the realism and adaptability of the simulation. Additionally, the model's handling of roundabouts and speed limit transitions could be improved to better mirror real-world driving behaviours.

Future work will focus on refining these aspects, enhancing the simulation's fidelity, and expanding its capabilities to model emerging transport technologies, such as electric vehicles. Such developments will not only address the computational scalability challenges but also contribute to a more nuanced understanding of traffic dynamics and energy demands, aligning with broader environmental and policy objectives.

References

1. An, L.: Modeling human decisions in coupled human and natural systems: Review of agent-based models. Ecological modelling **229**, 25–36 (2012)
2. Axtell, R., Farmer, D., Geanakoplos, J., Howitt, P., Carrella, E., Conlee, B., Goldstein, J., Hendrey, M., Kalikman, P., Masad, D., others: An agent-based model of the housing market bubble in metropolitan Washington, DC. In: Deutsche Bundesbank's Spring Conference on "Housing markets and the macroeconomy: Challenges for monetary policy and financial stability. pp. 5–6 (2014)
3. Behrisch, M., Bieker, L., Erdmann, J., Krajzewicz, D.: SUMO–simulation of urban mobility: an overview. In: Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation. ThinkMind (2011)

4. 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: Modeling Mobility with Open Data: 2nd SUMO Conference 2014 Berlin, Germany, May 15-16, 2014. pp. 47–60. Springer (2015)
5. BUBER, E., DIRI, B.: Performance analysis and cpu vs gpu comparison for deep learning. In: 2018 6th International Conference on Control Engineering & Information Technology (CEIT). pp. 1–6 (2018). https://doi.org/10.1109/CEIT.2018.8751930
6. Casas, J., Ferrer, J.L., Garcia, D., Perarnau, J., Torday, A.: Traffic simulation with aimsun. Fundamentals of traffic simulation pp. 173–232 (2010)
7. Erdmann, J.: Sumo's lane changing model (2014), https://elib.dlr.de/102254/1/Springer-SUMOs_Lane_changing_model.pdf
8. Erdmann, J., D.Krajzewicz.: Sumo's road intersection model (2011), https://core.ac.uk/download/pdf/31015083.pdf
9. Filatova, T., Verburg, P.H., Parker, D.C., Stannard, C.A.: Spatial agent-based models for socio-ecological systems: Challenges and prospects. Environmental modelling & software **45**, 1–7 (2013)
10. Ge, J.: Endogenous rise and collapse of housing price: An agent-based model of the housing market. Computers, Environment and Urban Systems **62**, 182–198 (3 2017). https://doi.org/10.1016/J.COMPENVURBSYS.2016.11.005
11. 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. American Economic Review **102**(3), 53–58 (2012). https://doi.org/10.1257/AER.102.3.53
12. Gilbert, N., Troitzsch, K.: Simulation for the social scientist. McGraw-Hill Education (UK) (2005)
13. Hager, K., Rauh, J., Rid, W.: Agent-based modeling of traffic behavior in growing metropolitan areas. Transportation Research Procedia **10**, 306–315 (2015)
14. Haman, I.T., Kamla, V.C., Galland, S., Kamgang, J.C.: Towards an multilevel agent-based model for traffic simulation. Procedia Computer Science **109**, 887–892 (2017)
15. Heppenstall, A.J., Crooks, A.T., See, L.M., Batty, M.: Agent-based models of geographical systems. Springer Science & Business Media (2011)
16. Heywood, P., Maddock, S., Casas, J., Garcia, D., Brackstone, M., Richmond, P.: Data-parallel agent-based microscopic road network simulation using graphics processing units. Simulation Modelling Practice and Theory **83**, 188–200 (2018). https://doi.org/https://doi.org/10.1016/j.simpat.2017.11.002, https://www.sciencedirect.com/science/article/pii/S1569190X17301545, agent-based Modelling and Simulation
17. Heywood, P., Richmond, P., Maddock, S.: Road network simulation using FLAME GPU. In: Euro-Par 2015: Parallel Processing Workshops: Euro-Par 2015 International Workshops, Vienna, Austria, August 24-25, 2015, Revised Selected Papers 21. pp. 430–441. Springer (2015)
18. Krajzewicz, D.: Traffic simulation with sumo–simulation of urban mobility. Fundamentals of traffic simulation pp. 269–293 (2010)
19. 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). pp. 183–187 (2002)
20. Krauss., S.: Microscopic Modeling of Traffic Flow: Investigation of Collision Free Vehicle Dynamics. Ph.D. thesis (1999), https://sumo.dlr.de/pdf/KraussDiss.pdf

21. Krauß, S., Wagner, P., Gawron, C.: Metastable states in a microscopic model of traffic flow. Physical Review E **55**(5), 5597 (1997)
22. 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: 2018 21st international conference on intelligent transportation systems (ITSC). pp. 2575–2582. IEEE (2018)
23. Manley, E., Cheng, T., Penn, A., Emmonds, A.: A framework for simulating large-scale complex urban traffic dynamics through hybrid agent-based modelling. Computers, Environment and Urban Systems **44**, 27–36 (2014)
24. McLane, A.J., Semeniuk, C., McDermid, G.J., Marceau, D.J.: The role of agent-based models in wildlife ecology and management. Ecological modelling **222**(8), 1544–1556 (2011)
25. 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. Applied Sciences **11**(12) (2021). https://doi.org/10.3390/app11125336, https://www.mdpi.com/2076-3417/11/12/5336
26. de Paiva Oliveira, A., Richmond, P.: Feasibility study of multi-agent simulation at the cellular level with flame gpu. In: The Twenty-Ninth International Flairs Conference (2016)
27. 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 preprint arXiv:2001.09727 (2020)
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). Science of the Total Environment **566**, 416–427 (2016)
29. Railsback, S.F., Grimm, V.: Agent-based and individual-based modeling: a practical introduction. Princeton university press (2019)
30. Richmond, P., Chimeh, M.K.: FLAME GPU: Complex system simulation framework. In: 2017 International Conference on High Performance Computing & Simulation (HPCS). pp. 11–17. IEEE (2017)
31. 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. Software: Practice and Experience (2023)
32. Richmond, P., Walker, D., Coakley, S., Romano, D.: High performance cellular level agent-based simulation with FLAME for the GPU. Briefings in Bioinformatics **11**(3), 334–347 (02 2010). https://doi.org/10.1093/bib/bbp073, https://doi.org/10.1093/bib/bbp073
33. Shirvani, M., Kesserwani, G., Richmond, P.: Agent-based simulator of dynamic flood-people interactions. Journal of Flood Risk Management **14**(2), e12695 (2021). https://doi.org/https://doi.org/10.1111/jfr3.12695, https://onlinelibrary.wiley.com/doi/abs/10.1111/jfr3.12695
34. Wallentin, G., Loidl, M.: Agent-based bicycle traffic model for salzburg city. GI_Forum J. Geogr. Inf. Sci **2015**, 558–566 (2015)
35. Zhang, B., DeAngelis, D.L.: An overview of agent-based models in plant biology and ecology. Annals of Botany **126**(4), 539–557 (2020)
36. 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: The world wide web conference. pp. 3620–3624 (2019)