Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 280 (2026) 993–1000

Procedia Computer Science

www.elsevier.com/locate/procedia

The 15th International Workshop on Agent-based Mobility, Traffic and Transportation Models, Methodologies and Applications (ABMTrans 2026)
April 14-16, 2026, Istanbul, Türkiye

Synchronizing Demand and Supply in Agent-Based Ride-Pooling Simulations: The mobiTopp–MATSim Live Coupling

Gabriel Wilkes[a,*], Robin Andre[a], Nico Kuehnel[b], Martin Kagerbauer[a], Peter Vortisch[a]

[a]*Karlsruhe Institute of Technology (KIT), Institute for Transport Studies, 76131 Karlsruhe, Germany*
[b]*MOIA GmbH, Alexanderufer 5, 10117 Berlin, Germany*

Abstract

This paper presents a live coupling between the agent-based travel demand and simulation modeling frameworks mobiTopp and MATSim to better capture feedbacks between on-demand ride-pooling services and travel demand. The approach embeds MATSim's demand-responsive transit module into mobiTopp's short-term simulation, so that each mode choice decision can respond to the contemporaneous fleet state, including vehicle availability, waiting times, detours and system-side rejections. This enables a consistent distinction between supply-side rejections due to limited capacity and passenger-side rejections of unattractive offers. We implement the coupling for a ride-pooling service in a weekly model of Hamburg, Germany, with about 187,000 agents and 3.8 million trips, and compare multiple scenarios that vary fleet size and vehicle capacity, as well as an uncoupled benchmark in which ride-pooling is always available with static service attributes. The results show that larger fleets increase the number of served trips, reduce waiting times and raise pooling rates, but also exhibit diminishing returns and lower vehicle utilisation at very high fleet sizes. We conclude that live coupling of demand and assignment models substantially improves the behavioural realism of ride-pooling simulations and supports more robust policy analysis, while highlighting the possibilities of using this approach for other applications. Furthermore, for applications with very large fleets there is a need for more selective request generation and computational enhancements.

Keywords: agent-based transport model; activity-based travel demand; ride-pooling; on-demand transportation

1. Introduction

Cities are increasingly experimenting with on-demand mobility services such as ride-hailing and ride-pooling, both as stand-alone offerings and as complements to public transport. The performance of these services is strongly shaped by how travel demand and fleet operations interact over time: vehicle availability, waiting times, and detours

* Corresponding author. Tel.: +49-721-608-52243.
E-mail address: gabriel.wilkes@kit.edu

feed back into user choices, which in turn affect fleet utilisation and service quality. Travelers not only decide whether to request an on-demand service, but may also reject offers if realised service quality (e.g., waiting time or detour) is too low and switch to alternative modes. Capturing these feedbacks and accept/reject decisions in a realistic yet computationally feasible way remains a central challenge for transport modeling and for the design and evaluation of on-demand services.

Agent-based travel demand models are well suited to address this challenge because they represent individual travelers and vehicles, their interactions, and the networks and facilities they use. This allows the dynamic interplay between demand and supply to emerge endogenously and is particularly valuable for systems with strong and rapidly changing feedbacks, just like on-demand services are. In contrast, macroscopic models typically struggle to represent such dynamics unless they are extended by specialized components [10, 3]. Even in agent-based simulations, however, on-demand transport remains demanding, as demand and supply react in real time while vehicles are shared among many travelers and users face uncertainty about whether a ride can be served.

Within the class of agent-based frameworks, MATSim is a widely used transport simulation tool that represents a network and a synthetic population with activity schedules. Multiple choice dimensions (route, mode, time-of-day) are explored by a co-evolutionary algorithm. Mode choice takes place between iterations during replanning. Live fleet status of supply-restricted on-demand systems is therefore not available at the moment of choice but has to be approximated via expectations about fleet availability. This can lead to pathologies such as repeated attempts to use an oversubscribed system, followed by a drop in demand and sudden oversupply in later iterations [9]. Lu et al. [6] mitigate this behaviour by temporarily relaxing explicit fleet simulation and estimating expected service quality during co-evolution, but the underlying mismatch between the time scale of learning (iterations) and operations (within-day simulation) remains. MATSim has been combined with external demand models. Ziemke et al. [13] integrate the aspatial activity scheduling model actiTopp [4]. Ziemke et al. [15] embedded MATSim in the FABILUT land use–transport suite together with SILO. Further examples include couplings with FEATHERS and CEMDAP [14, 16]. These studies demonstrate the flexibility of combining MATSim with external demand models but typically rely on offline iterations and provide only limited integration with the live dynamics of on-demand supply.

mobiTopp is another agent- and activity-based travel demand modeling framework written in Kotlin [7]. A long-term module generates a synthetic population, weekly activity plans, and mobility tools at the person level, while a short-term module simulates a full week of activities and trips in chronological order. During this simulation, destinations and modes are chosen using discrete choice models. mobiTopp has been successfully applied to regions with several million agents [12]. mobiTopp does not perform network assignment but depends on externally computed travel times. mobiTopp has been combined with MATSim in iterative workflows, where mobiTopp generates weekly plans and destinations and MATSim performs traffic assignment [2], or both mode choice and assignment [18]. Beyond MATSim, a live coupling of mobiTopp with the FleetPy simulation framework has been implemented, where FleetPy handles detailed on-demand fleet operations and mobiTopp simulates the remaining demand [11]. This provides a first example of live interaction between an agent-based demand model and a detailed fleet simulation, but it is limited on the scope of FleetPy (i.e., on-demand service only) and suffers from overhead because information is only exchanged through socket interfaces.

From the perspective of this study's goals, existing combinations of demand and assignment models either suffer from limited code interoperability [14], a trip- versus agent-based dichotomy at the assignment stage [8], or an insufficient representation of live fleet status for on-demand systems [9]. The temporal balance between demand and supply is crucial for services such as ride-pooling: the actual fleet status is a key determinant of whether a ride is available and under which conditions (e.g., waiting times, detours, occupancy), and whether travelers accept or reject the offered trip. A framework that combines (i) an activity-based weekly demand model, (ii) large-scale agent-based assignment, and (iii) representation of on-demand fleet status during mode choice and offer acceptance is still missing.

We address this gap by directly coupling mobiTopp with MATSim in a way that allows live feedback of current fleet status *during* the decision process itself. This paper makes three contributions. First, we propose an architecture for a

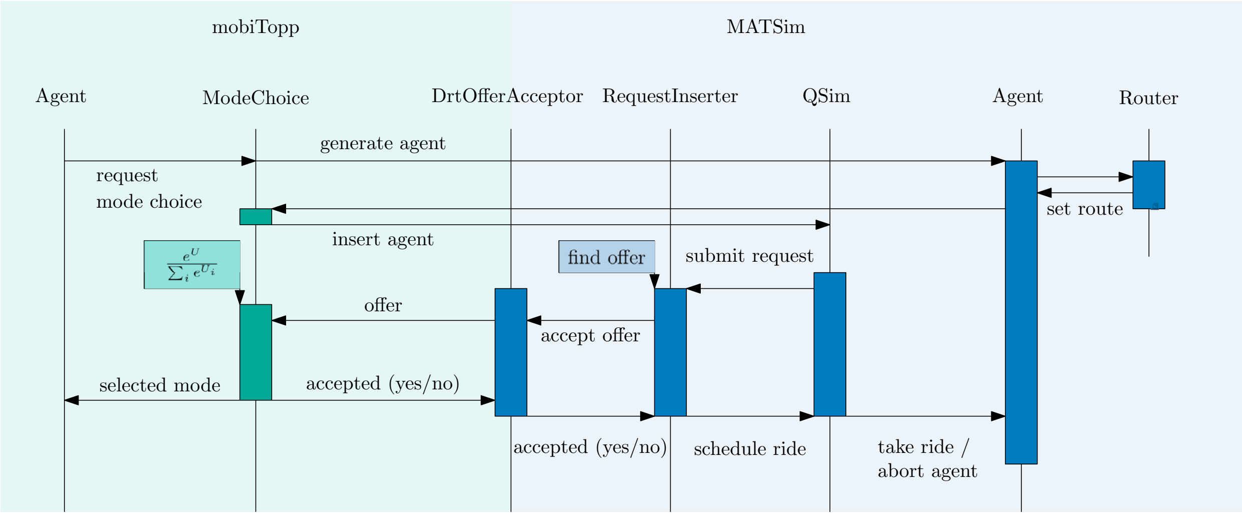

Fig. 1. The communication process between the mobiTopp and MATSim frameworks for a single mode choice request of a mobiTopp agent

live coupling in which MATSim's on-demand fleet simulation is embedded into mobiTopp's short-term agent-based demand model, enabling agents' mode choices and accept/reject decisions to respond to the contemporaneous state of an on-demand fleet. Second, we present a first implementation of this coupling, including the technical interface between the two frameworks and the integration with MATSim's on-demand transport module. Third, we apply the coupled model to an illustrative case study and compare it against setups without live fleet feedback. We furthermore demonstrate how the representation of ride-pooling demand, fleet utilisation, rejection patterns, and other indicators changes as a function of fleet characteristics.

2. Methodology

The goal of this software design is to create a fine granular coupling of mobiTopp and MATSim. The primary simulation state is held in mobiTopp, whereas MATSim provides services for individual requests passed by mobiTopp. In this particular case, we want to delegate the search of request insertions for an on-demand ride-pooling system to the DRT module in MATSim and play back these calculation results to the choice function in mobiTopp, so that the agent chooses their travel mode in accordance to the results of the DRT module. The general framework runs both, mobiTopp and MATSim, in parallel. Both models run in a temporal sequence. While MATSim typically runs in one-second time steps, mobiTopp runs on the scale of one-minute time steps but may skip time steps where nothing needs to be decided (event-based scheduling). As such, both models need to synchronize regularly. In essence, the MATSim simulation is halted whenever there is a new decision point within mobiTopp and it is resumed until the next decision point after the decision has been taken. In the case of our ride-pooling/DRT study, the synchronization primarily exchanges information about the fleet status for mode choice in mobiTopp. In addition, mobiTopp execution is halted until MATSim provides the necessary inputs for the decision problem to ensure temporal consistency within mobiTopp. As both mobiTopp and MATSim are Java or Java-compatible software, the coupling is implemented in Kotlin. It is published as open-source software [1].

Figure 1 visualizes the communication procedure to generate a coupled mode choice situation. Once an agent in mobiTopp needs to decide for a transport mode, the utility for the ride-pooling mode is calculated using live fleet information. In the current state, this is achieved by:

1. Cloning the mobiTopp agent for each trip in MATSim (trip-based assignment).
2. Creating on-demand trip route information incl. access and egress times as well as insertion constraints (i.e., max wait time, max detour/travel times) using MATSim's trip router.

3. Inserting the clone into MATSim's queue simulation. This will trigger an on-demand request submission, for which MATSim's request inserter searches a feasible insertion.
4. If an insertion is found, MATSim's offer acceptor has the possibility to accept or reject the offer. mobiTopp hooks onto that interface such that the offer, including all real time information such as offered waiting time as well as travel time can be fed back to the mode choice model. If no insertion is found, the mobiTopp agent performs mode choice without on-demand transportation as an available option.
5. In mobiTopp, the agent performs a mode choice based on all known mode utilities, considering the offer for on-demand transportation.
6. If the chosen mode is on-demand transport, the offer is accepted, otherwise it is rejected. This decision is fed back to MATSim. In case of acceptance, the request is scheduled into the respective vehicle schedule and will be served, otherwise the agent is set to stuck and abort in MATSim.

This means that we can now distinguish between two types of rejections: *system-side rejections* due to limited to supply and *passenger-side rejections* due to unattractive offers.

3. Scenario

This study was conducted in a model of the city of Hamburg (Germany) and its surrounding region. For the purpose of this work, the model was restricted to include only the inhabitants of the city and further reduced to a 10% sample, resulting in around 187,000 individuals. The persons are assigned to households and mobility-tool ownership models are used to ensure correct availability of cars and public transport passes. Each agent is equipped with a full 7-day activity schedule (Monday to Sunday). In total, around 3.8 million trips are undertaken. For this study, only the most common used modes were retained: walking, cycling, public transport, car as driver, car as passenger. Furthermore, a strongly simplified mode choice model (multinomial logit) is applied, including as influencing variables only travel time and travel costs, and (for ride-pooling) waiting time. Ride-pooling attractiveness is intentionally increased for the purpose of this study. Furthermore, it is assumed that ride-pooling passengers always ride alone.

The ride-pooling service area was set to encompass most parts of the city of Hamburg. Furthermore, a maximum detour of 15 minutes and a maximum waiting time of 10 min were set as constraints. No fleet rebalancing algorithm is used. Variation in the scenarios is performed concerning the number of vehicles and the number of seats per vehicle (i.e., maximum number of passengers). In addition to simulations coupled with MATSim, an uncoupled simulation was performed. In the uncoupled case, ride-pooling is not represented through explicit vehicle movements; instead, it is assumed that ride-pooling is always available and static travel and waiting times (3 min) are used. It therefore demonstrates a case of unconstrained ride-pooling. In total, the following nine scenarios were simulated: 1) without coupling; 2-9) with coupling, using the following variations: number of vehicles: [1, 10, 100, 1000], number of seats per vehicle: [6, 8].

4. Results

The results of the simulations regarding overall number of rides and average customer waiting time are depicted in Fig. 2, further results are given in Table 1. With a larger fleet, more rides can be served. However, while between the scenarios with 1, 10 and 100 vehicles the number of trips nearly grows linear with fleet size, in the 1000 vehicles scenario the number of rides only is around 4 times larger than in the 100 vehicles scenario. This indicates saturation effects and oversupply. Waiting times decrease with a larger fleet which indicates an increased service quality. As all trips in the service area result in requests at the DRT provider, system-side rejections are very high (nearly 1) with small fleet sizes and decrease with 1000 vehicles to 62 %. This is explained with a larger fleet being able to serve more requests.

Further indicators reveal improved system performance using a larger fleet. The pooling rate—defined as the trips that are with at least one other passenger for any period—increases from 77% in the 1 vehicle scenario to 90 % in 1000 vehicles scenario. The share of empty vehicle kilometers decreases from 42 % in 1 vehicle scenario to vs. 18% in the 100 vehicles scenario. The system efficiency, defined by Lehnert et al. [5], divides the total passenger kilometers

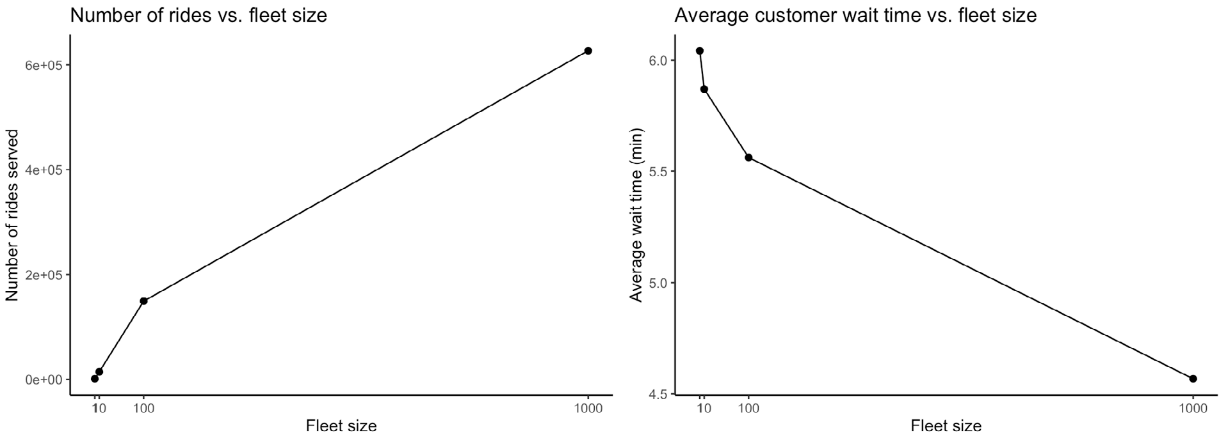

Fig. 2. Changing the fleet size directly affects number of served rides (left) and quality in terms of waiting time (right), allowing real-time impacts in mobiTopp's mode choice decisions.

booked (i.e., the demanded mileage) by the vehicle kilometers driven (i.e., the mileage required to serve the demand), with a higher value indicating a higher efficiency. In our results, the efficiency increases with the scale of the system (number of rides and vehicles). These positive effects of system performance are explained with an increase in demand density which increases probability of pooling [17]. However, further analysis reveals that the utilization of the fleet is smaller in the 1000 vehicles scenario: many vehicles are in idle status. This can be explained by the fact that without rebalancing, many vehicles are not in reach for some of the demand, leading to more rejections and underutilization of the fleet.

Interestingly, the different vehicle capacity of 6 or 8 passengers does not have a large impact, the results between these two values are nearly identical in all fleet size scenarios. However, as all results, this needs to be seen as a result in the specific conditions of the scenarios modeled here. In other settings, including other parameter values for the fleet configuration, results could be different.

Table 1. Simulation results: ride-pooling system and usage statistics in all scenarios

Scenario	Uncoupled	1v6s	1v8s	10v6s	10v8s	100v6s	100v8s	1000v6s	1000v8s
Fleet size [vehicles]	—	1	1	10	10	100	100	1000	1000
Capacity [seats]	—	6	8	6	8	6	8	6	8
Total number of rides	1,880,460	1,151	1,154	14,425	14,419	149,339	150,335	626,958	625,384
Mean total travel time [min]	—	13.1	13.2	12.2	12.2	12.1	12.1	12.1	12.1
Median waiting time [min]	—	6.6	6.6	6.3	6.2	5.8	5.8	4.2	4.3
Share system-side rejections	—	1.00	1.00	0.99	0.99	0.91	0.91	0.62	0.62
Pooling rate	—	0.77	0.78	0.80	0.80	0.84	0.84	0.90	0.90
Empty ratio	—	0.42	0.42	0.37	0.37	0.25	0.25	0.18	0.18
Mean occupancy [p-km/v-km]	—	0.83	0.83	0.96	0.96	1.2	1.2	1.37	1.37
System efficiency	—	0.69	0.69	0.78	0.79	0.93	0.93	1.00	1.00

The hourly demand profiles for ridesharing during the simulation week are given in Fig. 3. A regular demand pattern can be observed with two peaks Mondays to Fridays and one peak on the weekend. It can be observed that in the unconstrained service the demand is still higher than in the 1000 vehicles scenario, suggesting unserved demand. The larger hourly demand on Saturdays and Sundays compared to weekdays in the 100 and 10 vehicles scenarios can be explained by shorter trips, leading to more rides per hour per vehicle. This effect does not occur in the unconstrained and 1000 vehicles scenarios, as in those supply is sufficient to fulfill most of the trips at the weekend.

Using this coupling, direct interactions between fleet status and travel demand emerge endogenously. Figure 4 illustrates these dynamics by showing waiting times and the destination–origin trip ratio for three selected time in-

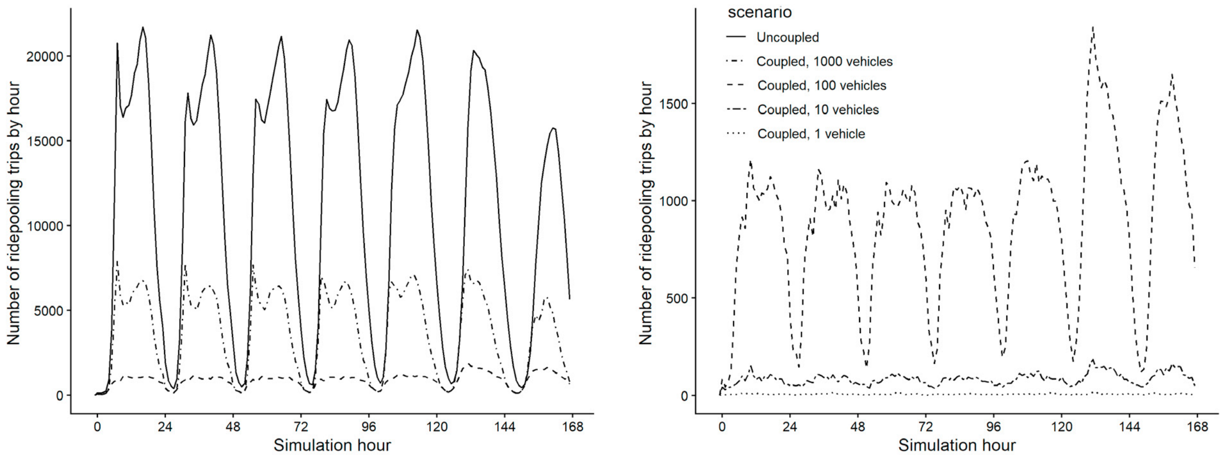

Fig. 3. Ride-pooling demand over the simulation week in all scenarios. Due to different magnitude of demand, scenarios are divided in two plots. The *coupled, 100 vehicles* scenario is included in both plots.

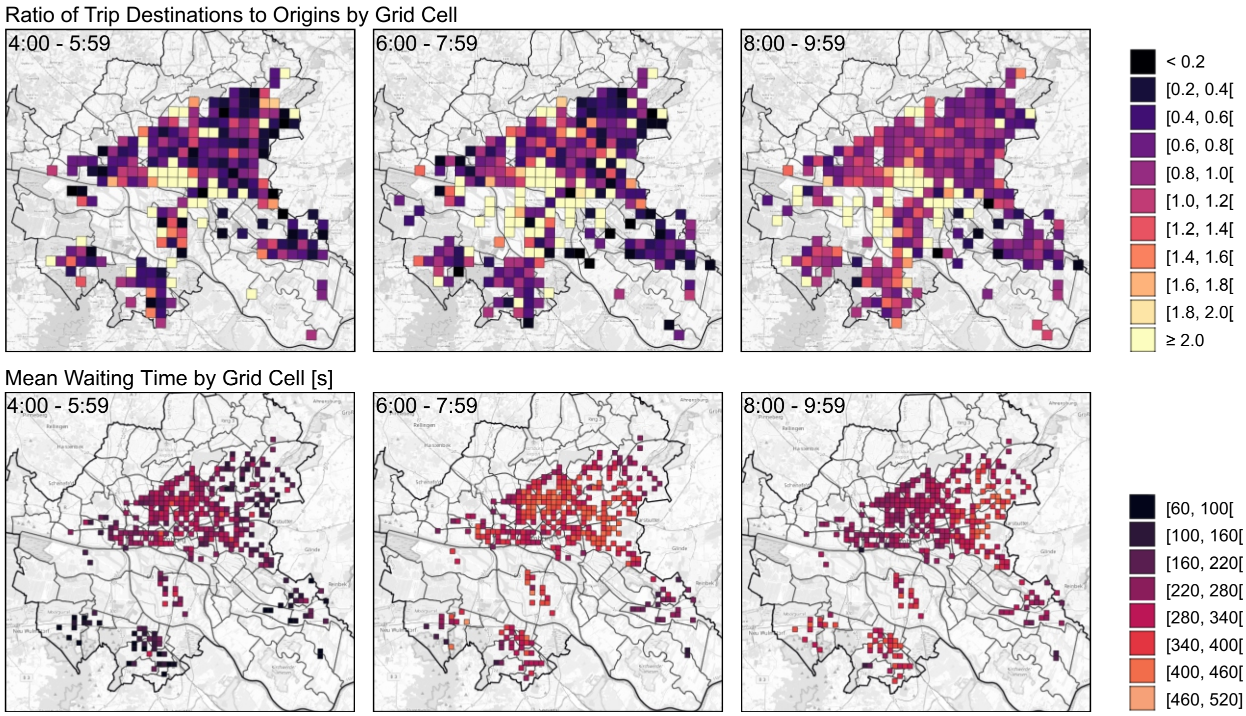

Fig. 4. Ratio of trip destinations to origins and waiting times by grid cell. Ratios above one indicate a net inflow of trips. Scenario used: *coupled, 1000 vehicles, 8 seats*. Background: OpenStreetMap.

tervals. In the early morning peak (before 6:00 AM), waiting times are generally even as vehicles are still relatively evenly distributed. As demand increases in the morning, waiting times increase (second time interval). After 8:00 AM, waiting times in the city center decrease markedly, while they increase in several peripheral areas. This shift coincides with a pronounced destination surplus in the center, as shown by destination–origin ratios above one, and is consistent with vehicles following preceding demand streams towards central destinations. As a result, supply accumulates in the center, reducing waiting times for trips originating there, while outer areas experience vehicle depletion and longer waits.

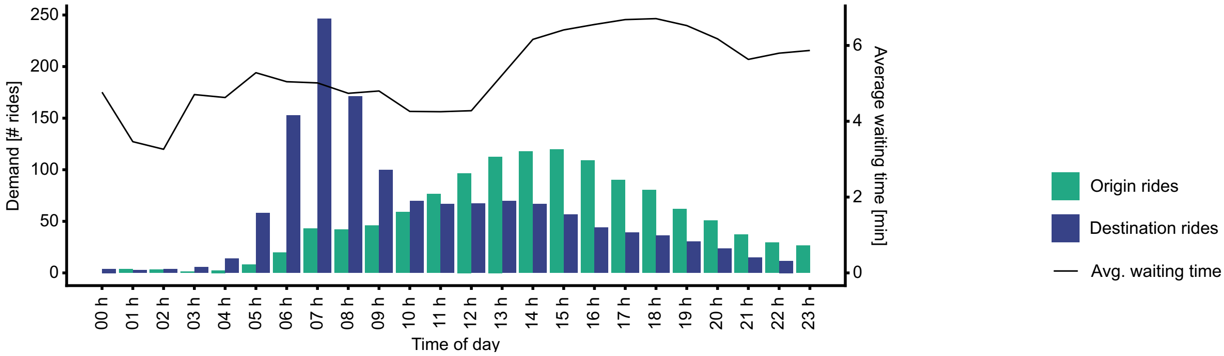

Fig. 5. Ride-pooling demand profile in the city center by demand of number of trips starting and ending, and waiting times. Means of Mondays to Fridays. Scenario used: *coupled, 1000 vehicles, 8 seats*.

These patterns can also be found in Figure 5, which shows the temporal evolution of demand and waiting times in the city center over 24 hours. Between 5:00 AM and 9:00 AM, the center exhibits a pronounced surplus of destinations over origins. As this balance shifts later in the day, waiting times increase. Importantly, these feedback effects arise only due to the explicit coupling of demand and fleet simulation. Waiting times result, among other factors, from past demand and influence subsequent ride-pooling mode choice; without this coupling—and in the absence of rebalancing—such spatially differentiated dynamics would remain hidden.

The run time of the simulation is 5 min in the uncoupled simulation and increases to up to 15 hours in the 1000 vehicles scenario—which still is feasible. Furthermore, the increased runtime due the coupling alone itself is only around 29 min. This means that major runtime increases are not due to the coupling but because of fleet control. Possibilities to improve this have been identified already.

5. Conclusion

The presented coupling combines a state-of-the-art activity-based travel demand model with a state-of-the-art agent-based transport simulation framework. While MATSim has capabilities of *updating* input demand along multiple dimensions, such as route and departure choice, mobiTopp allows us to generate the demand with live feedback from MATSim. A key difference is that MATSim usually looks at a single day only, whereas mobiTopp models a whole simulation week, demonstrating the capability of capturing within-week variability of demand. This is especially useful in DRT scenarios, as demand curves show strong fluctuations across weekdays and weekends, in particular.

In this paper we showed the integration for the case of ride-pooling, which has especially high interdependencies between demand and supply. We demonstrated some possibilities of using this approach in an example scenario with rather optimistic ride-pooling mode choice parameters. Based on this, many further details concerning interdependencies between choice of other modes and fleet configuration could be analyzed. For example, the explicit fleet simulation allows to model distinct demand-supply constraints such as for wheelchair users, who may require a specific subfleet of accessible vehicles.

Currently, an unfiltered amount of requests is submitted to the DRT system because every agent always submits a request for every trip. This results in foreseeable rejections. A more sophisticated selection strategy for the mode choice situation, such as a two-phase selection process to avoid generating DRT requests for infeasible trips, may prove beneficial in reducing the strain on the routing system, and in turn improve the simulation efficiency. This could be more in line with the real world, where passengers would first decide whether ride-pooling would be an option at all and only then actually submit a request, upon which the system either rejects or the passenger can decide based on the resulting offer.

Further research and development efforts will be taken to extend the coupling to larger scenarios, and to integrate intermodal trips. Similarly, this approach can be used to retrieve other demand-dependent travel times, such as for the mode car, or supply constraints such as live capacities of a crowded bus. Through this integration, the limitations of each individual travel demand modeling framework could be overcome.

Acknowledgements

This research was funded by the German Federal Ministry of Transport by decision of the German Bundestag within the project *Integration von autonomen Fahrzeugen in Mobilitätsanwendungen - ALIKE*, grant numbers: 45AVF3B021 (MOIA) and 45AVF3F061 (KIT).

Conflict of Interests

It is acknowledged that Nico Kuehnel is employed at the ride-pooling operator MOIA.

References

[1] Andre, R., Wilkes, G., Kuehnel, N., 2026. mobiTopp-MATSim-Coupling NG. URL: https://github.com/kit-ifv/mobitopp-matsim-coupling-ng.

[2] Briem, L., Mallig, N., Vortisch, P., 2019. Creating an integrated agent-based travel demand model by combining mobitopp and matsim. Procedia computer science 151, 776–781. doi:10.1016/j.procs.2019.04.105.

[3] Hartleb, J., Friedrich, M., Richter, E., 2022. Vehicle scheduling for on-demand vehicle fleets in macroscopic travel demand models. Transportation 49, 1133–1155.

[4] Hilgert, T., Heilig, M., Kagerbauer, M., Vortisch, P., 2017. Modeling week activity schedules for travel demand models. Transportation Research Record 2666, 69–77. URL: https://doi.org/10.3141/2666-08, doi:10.3141/2666-08.

[5] Lehnert, M., Liebchen, C., Schiefelbusch, M., Bruder, P., Harmann, D., 2026. Insights into ridepooling efficiency: A case study analysis using a novel operational performance measure framework. Journal of Urban Mobility 9, 100173. doi:https://doi.org/10.1016/j.urbmob.2025.100173.

[6] Lu, C., Rakow, C., Nagel, K., 2025. Developing an improved simulation procedure for demand-responsive transport services in agent-based transport simulation frameworks. Preprint available at SSRN doi:http://dx.doi.org/10.2139/ssrn.5132631.

[7] Mallig, N., Kagerbauer, M., Vortisch, P., 2013. mobitopp – a modular agent-based travel demand modelling framework. Procedia computer science 19, 854–859. doi:10.1016/j.procs.2013.06.114.

[8] Moeckel, R., Kuehnel, N., Llorca, C., Moreno, A.T., Rayaprolu, H., 2020. Agent-based simulation to improve policy sensitivity of trip-based models. Journal of Advanced Transportation 2020, 1902162. doi:https://doi.org/10.1155/2020/1902162, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1155/2020/1902162.

[9] Schlenther, T., Volotskiy, T., Leich, G., Zwick, F., Kuehnel, N., Smirnov, J., Nagel, K., 2025. Ridepooling and Public Transit: How Pricing Schemes Reveal the Tradeoff between Intermodality and On-Demand Efficiency. Transportation Research Record 2679, 207–220. doi:10.1177/03611981251346769, arXiv:https://doi.org/10.1177/03611981251346769.

[10] van der Tuin, M., Spruijtenburg, D., Zhou, H., 2023. Vehicle dispatching in macroscopic transport models: modelling demand responsive transit. Transportation Research Procedia 72, 2573–2580.

[11] Wilkes, G., Engelhardt, R., Briem, L., Dandl, F., Vortisch, P., Bogenberger, K., Kagerbauer, M., 2021. Self-regulating demand and supply equilibrium in joint simulation of travel demand and a ride-pooling service. Transportation research record 2675, 226–239. doi:10.1177/0361198121997140.

[12] Wilkes, G., Engelhardt, R., Kostorz, N., Dandl, F., Vortisch, P., Bogenberger, K., Kagerbauer, M., 2022. Autonomous ride-poooling in hamburg - results from an integrated travel demand and fleet simulation model, in: 101st Transportation Research Board Annual Meeting, Washington D.C., January 9-13, 2022.

[13] Ziemke, D., Charlton, B., Hörl, S., Nagel, K., 2021a. An efficient approach to create agent-based transport simulation scenarios based on ubiquitous Big Data and a new, aspatial activity-scheduling model. Transportation Research Procedia 52, 613–620. doi:https://doi.org/10.1016/j.trpro.2021.01.073.

[14] Ziemke, D., Knapen, L., Nagel, K., 2021b. Expanding the analysis scope of a matsim transport simulation by integrating the feathers activity-based demand model. Procedia Computer Science 184, 753–760. doi:https://doi.org/10.1016/j.procs.2021.04.022.

[15] Ziemke, D., Kuehnel, N., Llorca, C., Moeckel, R., Nagel, K., 2022. Fabilut: The flexible agent-based integrated land use/transport model. Journal of Transport and Land Use 15, pp. 497–526. URL: https://www.jstor.org/stable/48719783.

[16] Ziemke, D., Nagel, K., Bhat, C., 2015. Integrating cemdap and matsim to increase the transferability of transport demand models. Transportation Research Record 2493, 117–125. URL: https://doi.org/10.3141/2493-13, doi:10.3141/2493-13, arXiv:https://doi.org/10.3141/2493-13.

[17] Zwick, F., Kuehnel, N., Moeckel, R., Axhausen, K.W., 2021. Ride-pooling efficiency in large, medium-sized and small towns -simulation assessment in the munich metropolitan region. Procedia Computer Science 184, 662–667. URL: https://www.sciencedirect.com/science/article/pii/S1877050921007195, doi:https://doi.org/10.1016/j.procs.2021.03.083.

[18] Zwick, F., Wilkes, G., Engelhardt, R., Axer, S., Dandl, F., Rewald, H., Kostorz, N., Fraedrich, E., Kagerbauer, M., Axhausen, K.W., 2022. Mode choice and ride-pooling simulation: A comparison of mobitopp, fleetpy, and matsim. Procedia Computer Science 201, 608–613. doi:10.1016/j.procs.2022.03.079.