Online integration of mobiTopp and MATSim for agent-based simulations of on-demand services (and beyond)

Paper submitted for the MATSim user meeting 2025

Robin Andre[1], Nico Kuehnel[2], Gabriel Wilkes[1], Martin Kagerbauer[1], and Peter Vortisch[1]

[1]Karlsruhe Institute of Technology (KIT), Institute for Transport Studies, Karlsruhe, Germany
[2]MOIA GmbH, Berlin, Germany

Keywords: travel demand model; ride-pooling, fleet-simulation; MATSim; mobiTopp.

1. Introduction

Pooled ride-hailing (ride-pooling) is often regarded as a promising new transport mode, particularly in light of introducing autonomous vehicles. In the ALIKE research project, an autonomous ride-pooling service is implemented in the city of Hamburg. The project includes research on users' behavior and acceptance, including stated choice experiments. To be able to assess the impact of autonomous ride-pooling in different scenarios, a travel demand model is built. To capture the impact of different fleet-configurations, a coupling with a fleet-simulation is pursued. In the following, we present a working prototype of this integrated modeling approach, combining mobiTopp as base travel demand model and MATSim for simulating the ride-pooling fleet operations. The same coupling may be extended to tightly couple the demand model with a live assignment for further transport modes.

2. Background

As a minimal input, MATSim typically requires network representation(s) and a synthetic population including activity schedules. During the replanning phase, many choice dimensions (including route, mode and time of day choice, among others) may be explored by the co-evolutionary algorithm. However, it is also possible to treat the demand as a completely external input and make use of MATSim's highly performant simulation features only, i.e., using it as a pure assignment model.

In terms of mode choice, extensions to the original random mutation choice have been described by Hörl (2021) and Rakow and Nagel (2023). These extensions have in common that they are closer to typical random utility choice formulations such as the multinomial logit model. The overall design of the simulation, however, stays similar to MATSim's default behavior, in that the mode choice is performed during MATSim's replanning step between iterations. As such, the live fleet status cannot be accounted for in the decision making process, which means that there have to be estimations on fleet availability. This is a non-trivial problem that can lead to many agents trying to use a supply-restricted on-demand system throughout the iterations only to be rejected and learning that the system may not be a good option. In the final iterations, where innovation is usually "shut off", the demand may then drop considerably such that there is a sudden oversupply which could actually accommodate additional agents (see e.g., Schlenther et al. (2025) for a description of the problem).

In terms of external demand models, Ziemke et al. (2021a) describe how the aspatial activity scheduling model actiTopp (Hilgert et al., 2017) has been integrated with MATSim. Here, actiTopp is responsible for generating activity schedules which are then fed into MATSim. The initial schedules do not have any spatial component and no information on travel (i.e., which mode or route) between activities. These dimensions are completed within the MATSim simulation. Moeckel et al. (2020) present the agent- and trip-based travel demand model MITO, which uses MATSim as a traffic assignment simulator. The Java-based coupling was later extended to the FABILUT integrated land use transport modeling suite by running MITO and MATSim as a transport model update from within the land use model SILO (Ziemke et al., 2022). In another study, Ziemke et al. (2021b) present the integration of the demand model FEATHERS that acts as an outer loop around MATSim. As FEATHERS is proprietary software written in C++, the integration relied on a file-based information exchange. Similarly, Ziemke et al. (2015) describe how the output of the CEMDAP activity-based demand model has been used as an initial solution for MATSim. In both of the latter cases, the external demand models serve as a starting solution in which MATSim acts as a - as Ziemke et al. (2021b) phrases it - *demand-adaptation model*.

In conclusion (and from the view of our study's goal), previous examples either suffered from a lack of code interoperability (Ziemke et al., 2021b), trip- and agent-based dichotomy (Moeckel et al., 2020), or limited live fleet status of on-demand systems (Schlenther et al., 2025). Since on-demand systems like ride-pooling are very sensitive to the temporal demand-supply balance, the actual fleet status is an important factor for whether a ride may be available to an agent or not - or under which conditions (waiting times, detours, etc.). Within the ALIKE project, we aim to couple mobiTopp with MATSim to achieve a live feedback of current fleet status *during* the actual decision process.

mobiTopp is an agent- and activity-based travel demand modeling framework written in Java (Mallig et al., 2013). It consists of a long-term and a short-term module. In the long-term module a synthetic population is generated, including household-relations, weekly activity plans and assignment of mobility tools and fixed destinations on a person-level. In the short-term module, a weekly simulation is performed, modeling chronologically all persons' activity plans including their trips in between. During this simulation, destinations and modes of transport are chosen using discrete choice models. mobiTopp was successfully applied in models with several millions agents (Wilkes et al., 2022).

mobiTopp does not cover the step of assigning travel demand onto a network. By default, it uses externally calculated travel times as an input for all parts where travel times are needed (mainly destination and mode choice decisions). These travel time values can be time-dependent (having, e.g., different travel times in peak hours and in off-peak hours). By default, travel times are calculated using macroscopic travel demand models, losing the agent-based nature in assignment. However, mobiTopp was already combined with MATSim in an iterative approach, i.e., where separate modeling steps are undertaken with the different frameworks and they exchange information after their individual runs. In one approach activity plans and destinations are built using mobiTopp, MATSim is applied for traffic assignment only (Briem et al., 2019), in another, MATSim is used for mode choice and traffic assignment (Zwick et al., 2022b). A live coupling of mobiTopp with another framework directly inside the mobiTopp simulation was performed with the FleetPy framework. Thereby, fleet simulation was performed in FleetPy and the rest of the travel demand was simulated in mobiTopp. The two simulation frameworks communicated with each other using sockets (Wilkes et al., 2021).

3. 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

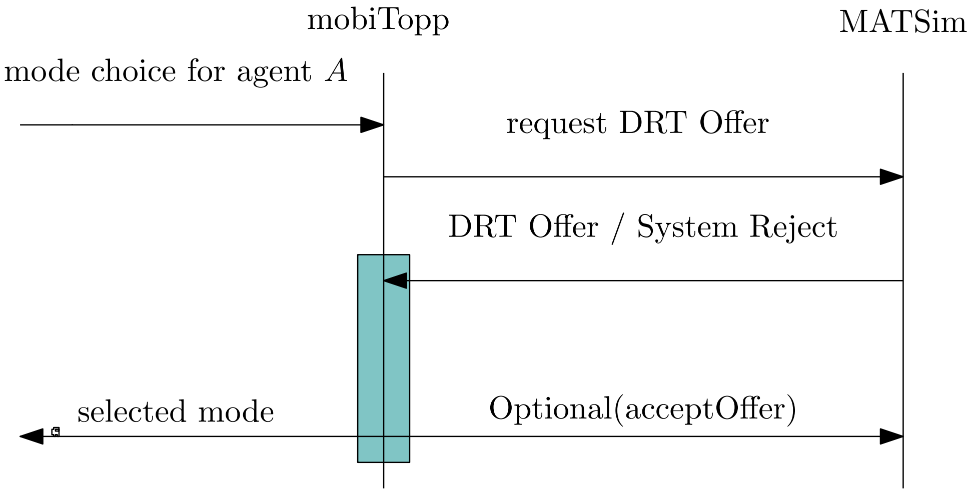


Figure 1: Example communication between mobiTopp and MATSim

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.

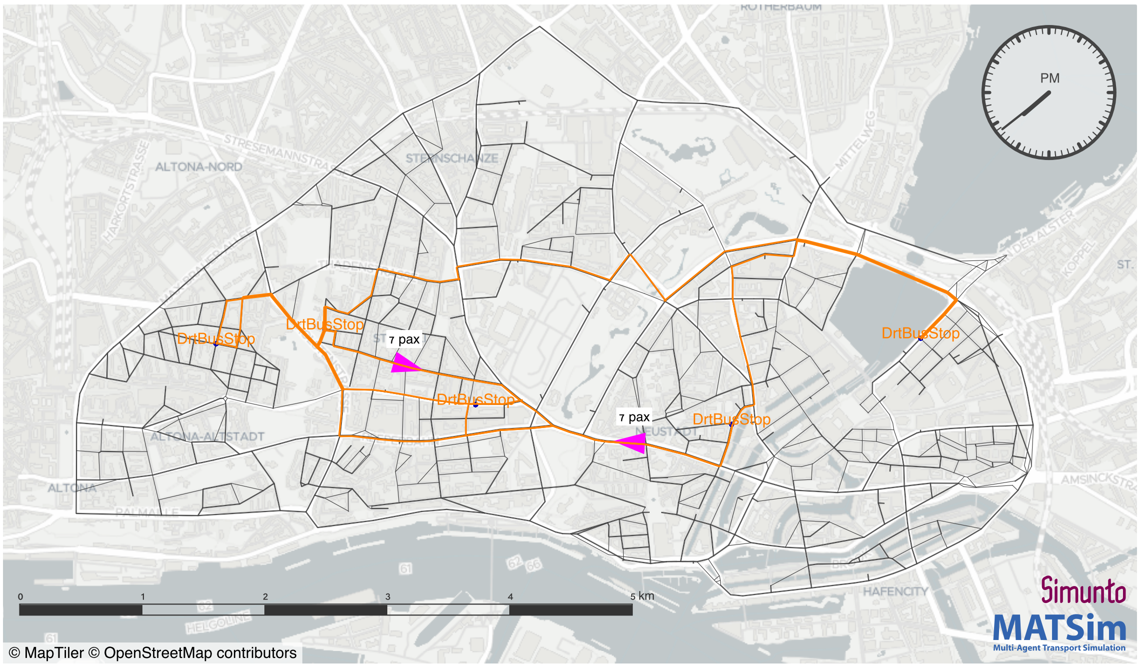
Figure 1 visualizes the communication procedure to generate a coupled mode choice situation. Once an agent in mobiTopp needs to decide for a transport mode, the utility for the ride-pooling mode is calculated using live fleet information. In the current state, this is achieved by

1. cloning the mobiTopp agent for each trip in MATSim
2. setting its leg to the DRT mode with an immediate departure
3. routing the leg using MATSim's trip router to correctly insert access/egress trips and
4. inserting it into the running QSim.

The optimizer in MATSim will then search for an insertion of the passenger request into the current fleet schedule based on vehicle availability. The potential insertion (including anticipated waiting time and arrival time window) is then passed to mobiTopp as an offer, which may be accepted or rejected by the agent. Using MATSim's event handler infrastructure, mobiTopp also listens to DRT rejection events to identify the cases where no feasible insertion could be found by the optimizer. 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. The acceptance of an offer comes from the agent performing its mode choice decision based on the calculated utility of each mode. If the agent decides for ride-pooling, the offer is accepted, if another mode is chosen, the offer is rejected.

4. Results and discussion

The current prototype has been implemented for a small test scenario in Hamburg. The scenario consists of a mobiTopp population with around 13,000 agents with around 278,000 trips. Besides ride-pooling, agents can use the modes bicycle, public transport, car as passenger, car as driver, walking, among others. As typical for mobiTopp, the scenario runs for a whole week. We use a very basic mode choice model, consisting only of alternative specific constants and travel time influence.



© MapTiler © OpenStreetMap contributors

Figure 2: Visualization of the example scenario with five stop locations between which agents request rides.

We run the scenario with our live coupling in which MATSim runs with a DRT fleet of two vehicles and five DRT stops. We use two different setups: in one scenario, the bookings are performed as described above. In another scenario we disable bookings, so the live-coupling is only used to retrieve travel times. This enables to review the impact of supply constrains. Furthermore, we use a standalone mobiTopp-scenario with assumptions on travel and waiting times to measure the difference in computation time compared to the coupling. We always use unrealistically high base utility for ride-pooling (alternative specific constant) is used to achieve a large ride-pooling demand.

Figure 2 shows the extent of the test scenario and the two vehicles that serve the demand for ride-pooling in the MATSim output (visualized in Via). When comparing the number of ride-pooling trips chosen by mobiTopp agents in both setups, we observe that 87,333 rides have been selected in the mobiTopp standalone version whereas 16,820 rides result in the live fleet coupling setup. While neither of these numbers reflect a realistically calibrated scenario, it demonstrates the principal working of the integration as the demand is constrained by the actual supply at each point in time and space.

Figure 3 shows the *system-side* and *customer-side* rejections over the course of the week. This demonstrates the important aspect of fluctuating fleet availability, which may be hard to take into account by aggregated estimates. Compared to an ideal autonomous fleet used here, real-world operational constraints may have an even bigger impact on the fleet availability (Zwick et al., 2022a).

The experiments were run on an Intel(R) Xeon(R) E-2288G using Eclipse Temurin OpenJDK 21.0.6. Both coupled scenarios take 45 minutes for completion, irrespective of the usage of booking calls. The standalone mobiTopp scenario takes 2.5 minutes to complete. A significant amount of time is spent in JVM internal thread synchronization methods, indicating that the software synchronization steps between mobiTopp and MATSim, as well as hidden internal parallelization structures within the programs are responsible for the increase in simulation time.

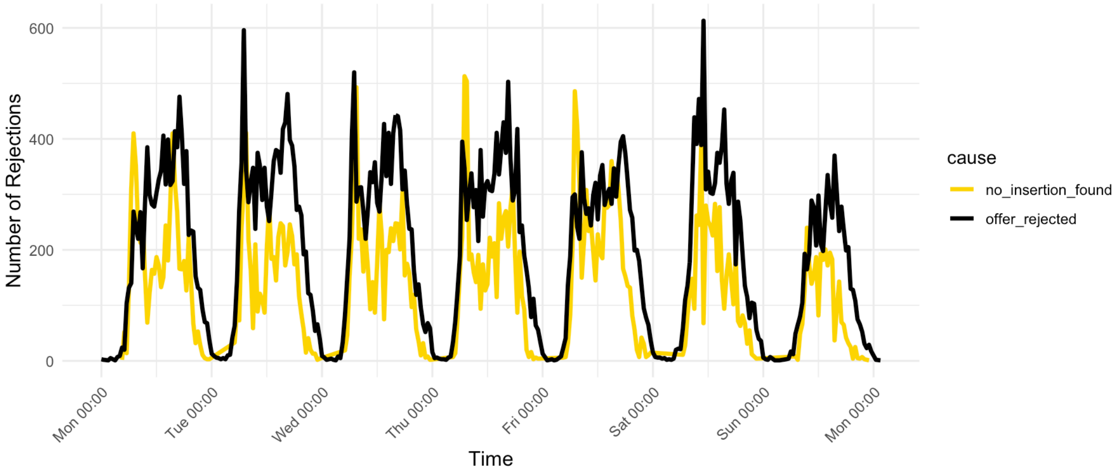


Figure 3: System-side and customer-side rejections over the course of the week.

5. Conclusions

With this integration of mobiTopp with MATSim a detailed and consistent agent-based modeling pipeline is built in a weekly travel demand model. Discrete choice models for mode choice are applied using trip properties based on actual supply.

In this paper we showed the integration for the case of ride-pooling, which has especially high interdependencies between demand and supply. Further research and development efforts will be taken to extend it to larger scenarios, to further improve performance, 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.

Improving the software communication bottleneck will allow for more efficient simulations and expansive interaction between multiple components. One potential use case is the integration of multiple travel modes, to incorporate reciprocal effects such as network saturation using ride-pooling and car traffic. A problem of the current approach is the unfiltered amount of requests submitted to the DRT system (as every agent always submits a request for every trip), which primarily result in forseeable 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. In the case of rejection, a ’fallback’ decision has to be made.

6. Acknowledgments

This work has received funding through the ALIKE project funded by the German Federal Ministry for Digital and Transport (grant number 45AVF3A011 45AVF3G071).

REFERENCES

Briem, L., Mallig, N., and Vortisch, P. (2019). Creating an integrated agent-based travel demand model by combining mobitopp and matsim. *Procedia computer science*, 151:776781.

Hilgert, T., Heilig, M., Kagerbauer, M., and Vortisch, P. (2017). Modeling week activity schedules for travel demand models. *Transportation Research Record*, 2666(1):69–77.

Hörl, S. (2021). Integrating discrete choice models with matsim scoring. *Procedia Computer Science*, 184:704–711.

Mallig, N., Kagerbauer, M., and Vortisch, P. (2013). mobitopp a modular agent-based travel demand modelling framework. *Procedia computer science*, 19:854859.

Moeckel, R., Kuehnel, N., Llorca, C., Moreno, A. T., and Rayaprolu, H. (2020). Agent-based simulation to improve policy sensitivity of trip-based models. *Journal of Advanced Transportation*, 2020(1):1902162.

Rakow, C. and Nagel, K. (2023). An annealing based approach to informed mode choice in agent-based transport simulations. *Procedia Computer Science*, 220:667–673.

Schlenther, T., Volotskiy, T., Leich, G., Zwick, F., Kuehnel, N., Smirnov, J., and Nagel, K. (2025). Ridepooling and public transit - how pricing schemes reveal the trade-off between intermodality and on-demand efficiency. *submitted to Transportation Research Record*.

Wilkes, G., Engelhardt, R., Briem, L., Dandl, F., Vortisch, P., Bogenberger, K., and Kagerbauer, M. (2021). Self-regulating demand and supply equilibrium in joint simulation of travel demand and a ride-pooling service. *Transportation research record*, 2675(8):226–239.

Wilkes, G., Engelhardt, R., Kostorz, N., Dandl, F., Vortisch, P., Bogenberger, K., and 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*.

Ziemke, D., Charlton, B., Hörl, S., and 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.

Ziemke, D., Knapen, L., and 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.

Ziemke, D., Kuehnel, N., Llorca, C., Moeckel, R., and Nagel, K. (2022). Fabilut: The flexible agent-based integrated land use/transport model. *Journal of Transport and Land Use*, 15(1):pp. 497–526.

Ziemke, D., Nagel, K., and Bhat, C. (2015). Integrating cemdap and matsim to increase the transferability of transport demand models. *Transportation Research Record*, 2493(1):117–125.

Zwick, F., Kuehnel, N., and Hörl, S. (2022a). Shifts in perspective: Operational aspects in (non-)autonomous ride-pooling simulations. *Transportation Research Part A: Policy and Practice*, 165:300–320.

Zwick, F., Wilkes, G., Engelhardt, R., Axer, S., Dandl, F., Rewald, H., Kostorz, N., Fraedrich, E., Kagerbauer, M., and Axhausen, K. W. (2022b). Mode choice and ride-pooling simulation: A comparison of mobitopp, fleetpy, and matsim. *Procedia Computer Science*, 201:608613.