Simulated Annealing in a Co-Evolutionary, Agent-Based Transport Modeling Framework - The Example of Ride-pooling Driver Supply Optimization

Nico Kuehnel*[1], Shivam Arora[2], Felix Zwick[1], and Qin Zhang[3]

[1]MOIA GmbH, Germany
[2]Department of Civil Engineering, IIT Roorkee, India
[3]School of Engineering and Design, Technical University of Munich, Germany

Short summary

This paper introduces an integrated simulated annealing optimization method within the co-evolutionary agent-based transport modeling framework MATSim, using a small illustrative ride-pooling service as an example to optimize driver shift supply for a given and static demand. Simulated annealing is a metaheuristic optimization algorithm that has already been employed in a wide range of problems and domains. MATSim makes use of a co-evolutionary design in which individual agents try to optimize their daily schedule by finding optimal transport options. The iterative nature of both simulated annealing and MATSim's co-evolutionary design makes the implementation straightforward and compatible. The outcomes validate the feasibility of the approach in optimizing specific components of the transport model and indicate its potential for future use in comparable applications. The presented case of driver supply optimization may help to design scenarios for new services and to better assess the efficiency and costs of such a service.

Keywords: agent-based transport model, demand-supply-matching, MATSim, on-demand mobility, operations research, simulated annealing.

1 Introduction

Transport systems are complex and involve various stakeholders, multiple modes of transportation, and numerous decision-making processes. Agent-based transport models, such as MATSim (Multi-Agent Transport Simulation, Horni et al., 2016), enable researchers to simulate and analyze the behavior of individual travelers and their interaction in the context of the wider transport system. The central feature of MATSim's is a co-evolutionary algorithm that enables individual agents to optimize their daily activity schedules through the identification of the most optimal travel options. Consequently, the decisions made by one agent can have implications for every other agent, often impacting shared resources such as road or bus capacity.

In numerous simulation studies, the supply side of the transportation system is treated as static while searching for a stable demand equilibrium. Typically, the effects of alterations to the supply side are analyzed *across* multiple simulations rather than *within* a single simulation. While this approach is suitable for many use cases, there are situations where the supply side must also react dynamically within the simulation. One example is the simulation of competing minibus operators that do not operate on fixed schedules, but are instead demand-driven and can be modeled using an evolutionary algorithm (Neumann, 2014). Other examples include public transport pricing and supply planning (Kaddoura et al., 2015), the implementation of traffic actuated or traffic adaptive transport signals that dynamically respond to current traffic flows (Kühnel et al., 2018), the optimization of charging infrastructure placement (Fadranski et al., 2023) or tour planning in freight applications (Zilske & Joubert, 2016). An in-depth discussion about optimization problems in (iterative) and stochastic simulation frameworks is given in (Flötteröd, 2017).

A prime example of an inherent necessity for supply-side optimization can be found in recent demand-responsive transport (DRT) systems such as online ride-hailing and ride-pooling, wherein an operator must dynamically respond to passenger requests. To address such cases, a central dispatcher optimizes the fleet supply using various algorithms such as insertion heuristics (Maciejewski, 2016) or integer linear programming (Alonso-Mora et al., 2017). A recent study highlighted the need for explicit simulation of operational aspects such as charging, hub facilities, and driver shift supply to yield realistic results for current human-operated fleet operations in which the drivers are employees of the operator (Zwick et al., 2022). This underscores the importance of generating realistic driver shift plans that minimize operating costs while maintaining high-quality service. For simulation studies without available historical data, or those subject to new policies or other changing conditions, the driver shift plans must be scheduled to match the anticipated demand. Given that these scheduling problems are typically NP-hard (Chuin Lau, 1996), simulated annealing (SA) has proven to be a valuable tool for addressing this challenge (Thompson, 1996).

SA is a metaheuristic optimization algorithm that has been applied across a range of domains, including transportation, to address complex optimization problems. SA can be integrated with co-evolutionary algorithms to optimize distinct facets of the transport system, as both approaches share important similarities and fall into the category of *general iterative algorithms* (Youssef et al., 2001). In this study, we provide a case illustration of SA employed within MATSim to optimize the supply side of DRT systems in the form of driver shift planning, based upon initial findings (Arora, 2021).

We demonstrate that SA is well-suited for the iterative design of a co-evolutionary transport model framework and can generate favorable outcomes for optimizing specific simulation components, as exemplified by driver shift planning. The versatility of the SA approach lies in its adaptability to various mobility-related optimization problems, including but not limited to DRT stop or hub placement, charging strategies, traffic signal plans, or fleet sizes. Consequently, the generic SA implementation within the MATSim framework will be made available as an open-source feature to encourage further optimization research in the field.

2 Methodology

Based on the DRT extension by Maciejewski (2016) including its default re-positioning strategy (Bischoff & Maciejewski, 2020) and the operational aspects described in Zwick et al. (2022) we simulate a human-operated ride-pooling service in MATSim and use SA to optimize the driver shift plan.

The general outline of SA is as follows:

1. Initialize the system with an initial solution λ_0 and set the initial temperature T_0 (to a high value).
2. Choose a candidate solution λ_i for iteration i by making a perturbation to the current solution.
3. Calculate the energy (or cost) difference between the candidate solution cost $c(\lambda_i)$ and the current accepted solution cost $c(\lambda_a)$.
4. If the energy difference is negative, accept the candidate solution as the new current solution.
5. If the energy difference is positive, accept the candidate solution with a probability $P_i(\lambda_i)$ that depends on the current temperature T_i and the energy difference. This probability decreases as the temperature decreases and is designed to allow the algorithm to escape from local minima.
6. Decrease the temperature according to a cooling schedule.
7. Repeat steps 2-6 until the stopping criterion is met (e.g., a maximum number of iterations is reached).

The acceptance probability in step 5 is calculated as:

$$P_i(\lambda_i) = e^{-(\frac{k\cdot(c(\lambda_i)-c(\lambda_a))}{T_i})} \tag{1}$$

Multiple cooling schedules exist to adjust temperature T_i. In this study we use an exponential multiplicative schedule:

$$T_i = T_0 * \alpha^i, \tag{2}$$

with a constant $0 < \alpha < 1$.

In our application of optimizing driver shifts, we assume an infinite pool of drivers and limit the optimization to a single day. Shifts s are characterized by their start and end times and are of a fixed duration $t_{d,s}$ of either 5 or 8 hours. An 8-hour shift requires a mandatory break of duration $t_{b,s}$= 60 min, which must occur no earlier than 3.5 hours and no later than 5.5 hours into the shift. The solution λ is defined as a shift plan, i.e., the set of n shifts $\lambda = \{s_1, s_2, \ldots, s_n\}$.

For the cost function, we propose a function that

1. sums up the driving hours of all shifts (shift durations $t_{d,s}$ minus optional break durations $t_{b,s}$) and multiplies it with the cost per operational hour θ,
2. subtracts from the costs the sum of revenues ϵ_r of all served rides $r \in R(\lambda)$ served with solution λ,
3. adds a penalty δ for each time bin t in which the rejection rate η_λ was greater than a predefined threshold η_{max}:

$$c(\lambda) = \theta \cdot \sum_{s \in \lambda} t_{d,s} - t_{b,s} - \sum_{r \in R(\lambda)} \epsilon_r + \sum_t \Gamma(t, \lambda) \cdot \delta, \tag{3}$$

$$\text{with } \Gamma(t, \lambda) = \begin{cases} 1 & \eta_\lambda(t) > \eta_{max}, \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

For the revenue of a ride, we propose a generic cost function that consists of a base fare β_0 and a distance-dependent price per kilometer β_{km}:

$$\epsilon_r = \beta_0 + \beta_{km} \cdot d_r, \tag{5}$$

where d_r is the distance of ride r.

For step 2 of the algorithm, multiple perturbations were defined to allow an extensive but guided exploration of possible solutions:

Add shift This strategy randomly adds a new shift to the shift plan by drawing from a weighted random distribution of possible time spans. Using a sliding window approach, each possible time span over time bins t gets a weight that relates to the request rejection rates $\eta(t)$ of the iteration of the last accepted solution. The more rejections a possible time span covers, the higher the probability of being selected. The time spans have fixed durations of either 5 or 8 hours, which are the two possible shift durations employed in this study.

Remove shift This strategy randomly removes a shift from the plan. Similar to the *add shift* strategy, a weighted selection from existing shifts in the plan is performed. Here, the weight is calculated by the efficiency of the shift, defined as the ratio of revenue earned over the cost of the shift (duration times the cost per hour) during the last accepted solution's iteration.

Move shift This strategy randomly moves the start of a shift forwards or backwards in time. The time difference is randomly drawn from a uniform distribution and respects the service times of the service.

Duplicate shift This strategy randomly duplicates an existing shift, by drawing from a weighted distribution. Similar to the removal of shifts, the weights are defined by the efficiency of a shift during the last accepted solution's iteration, with more effective shifts being more likely of being duplicated.

Change shift duration This strategy changes the duration of an existing shift by randomly choosing between a 5- and an 8-hour shift.

The SA is implemented in parallel to MATSim's usual iterative cycle as shown in figure 1. The mobility simulation is used in both, the SA algorithm and MATSim's standard demand co-evolution and represents the joint environment to allow the evaluation of the solution (set). The actual evaluation (i.e. scoring/cost updates) and preparation of new solutions (replanning/solution update) are performed in separate cycles. The solution update includes the cooling schedule, the decision for accepting the latest solution and perturbing the accepted solution. For the present study, we assume the demand to be static and only optimize the driver shift plan given a fixed demand to show the applicability of the approach. Thereby, the simulation framework mimics an iterative traffic assignment model for fleet simulation only, ignoring any additional modes such as private cars.

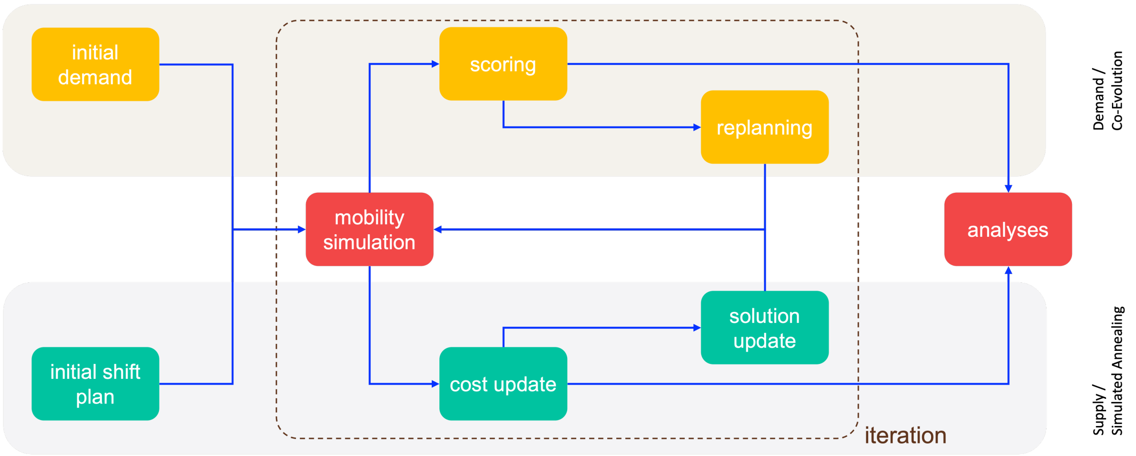

Figure 1: Updated MATSim cycle adapted from Horni et al. (2016) to include the implemented simulated annealing cycle which runs in parallel to the original MATSim cycle.

To test our implementation, we make use of a small existing scenario for the city of Holzkirchen in southern Bavaria, Germany. The scenario has been described by Zwick et al. (2021) and is available open source[1]. The temporal distribution of DRT requests is shown in figure 2.

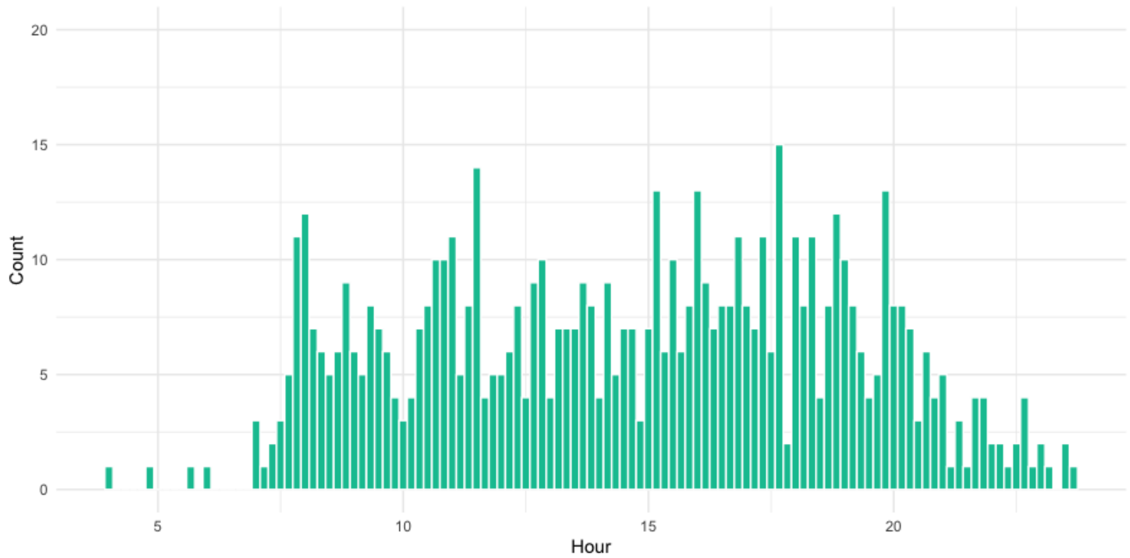

Figure 2: DRT requests over the time of day in the Holzkirchen scenario (Zwick et al., 2021).

In total, we let the simulation run for 300 MATSim iterations. As the rebalancing algorithm of the DRT also needs to adapt to previous iterations, we apply a ratio of 2 MATSim iterations per SA iteration. In addition, we choose to have 3 SA iterations per cooling cycle (making it 6 MATSim iterations per cycle). The best solution found over all iterations will be fixed for the last 3 iterations and serves as the final output. The initial temperature T_0 is set to 1,500 and α to 0.85. In every SA iteration, we randomly choose between 1 and 10 perturbations from the strategies defined above. The cost per driver hour θ is set to EUR 30, to assume somewhat realistic costs of bus drivers,

[1] https://github.com/matsim-org/matsim-libs/tree/master/examples/scenarios/holzkirchen

including some overhead (Frank et al., 2008). The cost penalty δ equals EUR 50, which has been found to be a good value to ensure minimum service level, with the rejection rate threshold η_{max} set to 0.15. On the revenue side, we choose $\beta_0 = 5\ EUR$ and $\beta_{km} = 0.7\ EUR/km$. The values given above do not reflect a coherent business case and are chosen for illustrative purposes. The shift breaks are not part of the optimization in this study, although perturbations on their time windows would also be possible. The initial shift plan is a manually created plan with a lot of oversupply (see iteration 0 in figures 4 and 5). In total, a maximum of 20 DRT vehicles may be employed in the simulation.

3 Results and discussion

The figures presented below illustrate the progression of solution quality and associated costs. Figure 3 displays the initial, accepted, current, and best overall costs across iterations, as well as the temperature curve that depicts the cooling schedule. The initial solution is characterized by high costs resulting from a considerable oversupply. In the first few iterations, the costs improve considerably and converge towards a minimum of approximately -1334 EUR. The erratic behavior of the current cost curve indicates that the algorithm is searching around the accepted solution space. The accepted cost curve reveals that, particularly around iteration 100 when the temperature remains high, the current accepted solution may be allowed to be inferior to a prior solution from earlier iterations. In general, the asymptotic nature of the curves suggests a relatively stable and optimized solution, which the algorithm reached in iteration 232.

It is worth noting that negative costs indicate a profitable service, as revenues exceed driver costs. However, the cost and revenue factors presented here are for illustrative purposes only and may not accurately reflect actual scenarios. Additionally, the fixed demand in the Holzkirchen scenario was estimated based on the assumption of an autonomous service with lower cost factors.

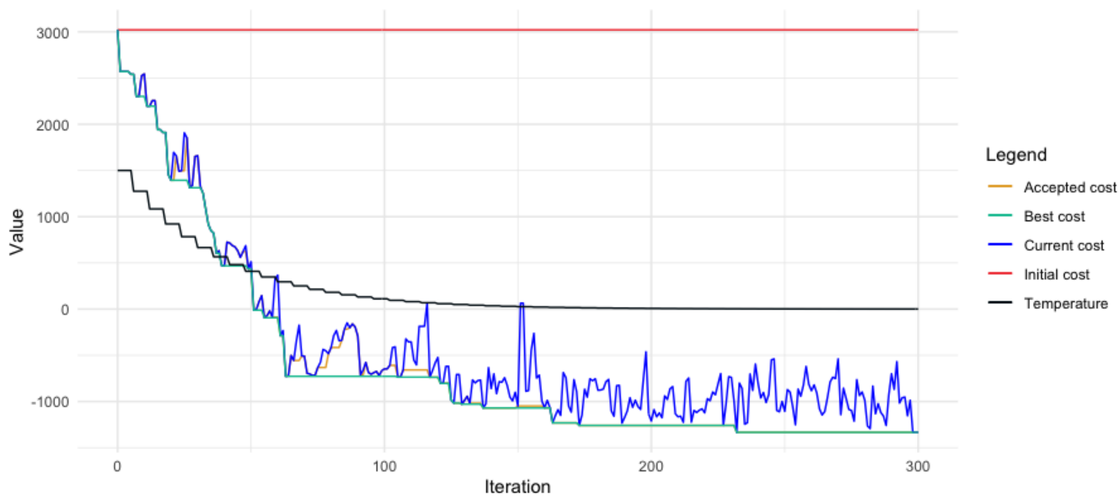

Figure 3: The different cost values (in EUR) and temperature over the course of the simulation.

Figure 4 depicts six plots of shift histograms and rejections at different iterations during the simulation. The algorithm aims to minimize driver hours by limiting the number of rejections to below a specified threshold. In the initial iteration, shifts are evenly distributed throughout the day, resulting in few rejections. Subsequent shift histograms exhibit a more detailed shift schedule with fewer total shifts. The final shift setup displays a significantly reduced shift histogram with an acceptable rejection rate (overall rejection rate of 4 %). The plots illustrate that the algorithm occasionally adds meaningless shifts randomly, such as shifts starting at midnight in iteration 60, which are subsequently eliminated. Additionally, the final shift plan is consistent with the demand pattern in figure 2, with a noticeable peak around 5 pm. In this hypothetical scenario, it shows that a fleet size of around 10 vehicles may be sufficient.

Figure 5 illustrates the vehicle occupancy at the same six iterations as the shift histograms in

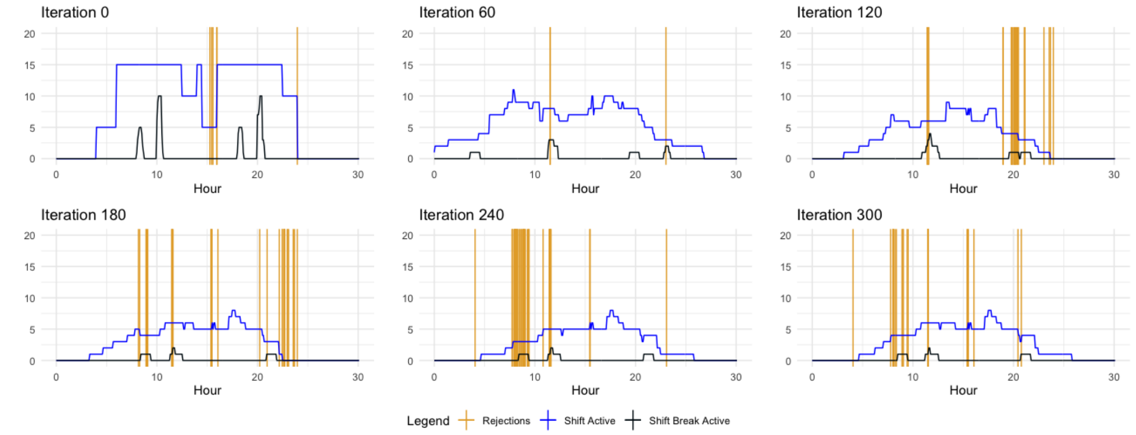

Figure 4: Shift histograms and rejections over the iterations 0, 60, 120, 180, 240 and 300. Each vertical line represents a rejected request. The blue and black lines depict the number of active shifts and shift breaks, respectively.

figure 4. The occupancy plot for iteration 0 reveals that too many shifts were scheduled given the demand, with "STAY" indicating vehicles on shift without tasks. A similar yet less pronounced pattern is present in iteration 60. In contrast, the shift schedules in iterations 240 and 300 demonstrate an efficient utilization of shifts based on demand, indicating that the algorithm has likely found a satisfactory solution to the problem.

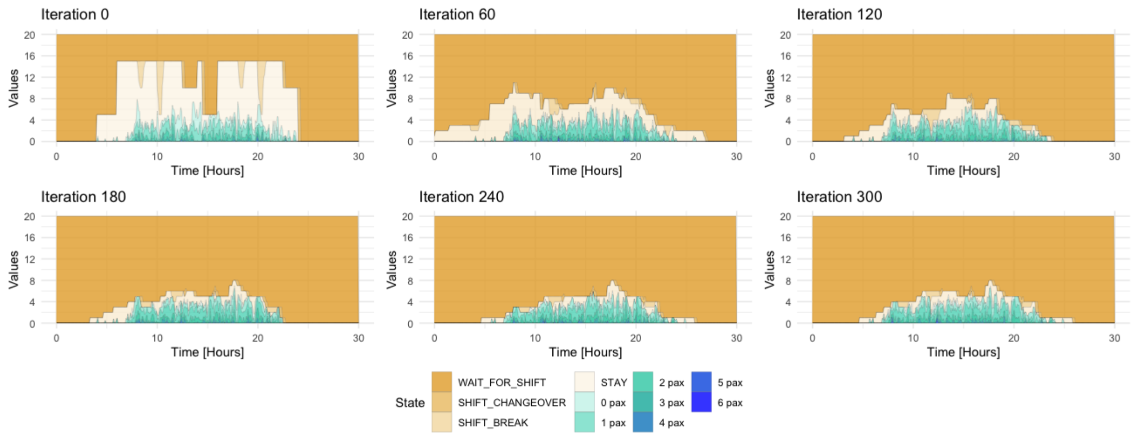

Figure 5: Occupancy plots over the iterations 0, 60, 120, 180, 240 and 300. It shows the distribution of vehicle states over the time of day, including the number of boarded passengers.

4 Conclusions

In conclusion, the results presented in this study demonstrate the applicability of the simulated annealing algorithm for the given problem. However, it is important to note that these results are only illustrative and require further refinement of input parameters, particularly in relation to driver costs, revenues, and demand. In addition, the cost optimization could be extended to also include other operating costs such as the distance-dependent costs of electricity for electric vehicles. Given that one can also infer the maximum amount of simultaneously operating vehicles one can also estimate the number of required vehicles in the fleet. Given the unpredictable nature of demand, an approximate and valid solution of supply is sufficient for the problem, which supports the idea of using heuristic approaches such as SA. Future research should focus on exploring adaptive and co-evolving demand, testing larger and realistic scenarios, and optimizing other components of the simulation to improve the accuracy of the results. The findings presented here may also be transferable to other iterative transport models.

Declaration of interests

We acknowledge that Nico Kuehnel and Felix Zwick are employed at the ride-pooling operator MOIA.

References

Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., & Rus, D. (2017). On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences of the United States of America*, *114*(3), 462–467. doi: 10.1073/pnas.1611675114

Arora, S. (2021). *Optimization of driver shift (and break) schedule using simulated annealing in ride-pooling services* (Master's thesis, School of Engineering and Design, Technical University of Munich). doi: 10.13140/RG.2.2.16415.64162

Bischoff, J., & Maciejewski, M. (2020). Proactive empty vehicle rebalancing for demand responsive transport services. *Procedia Computer Science*, *170*, 739-744. Retrieved from https://www.sciencedirect.com/science/article/pii/S1877050920306220 doi: https://doi.org/10.1016/j.procs.2020.03.162

Chuin Lau, H. (1996). On the complexity of manpower shift scheduling. *Computers Operations Research*, *23*(1), 93-102. Retrieved from https://www.sciencedirect.com/science/article/pii/0305054894000940 doi: https://doi.org/10.1016/0305-0548(94)00094-O

Fadranski, D., Syré, A. M., Grahle, A., & Göhlich, D. (2023). Analysis of charging infrastructure for private, battery electric passenger cars: Optimizing spatial distribution using a genetic algorithm. *World Electric Vehicle Journal*, *14*(2). Retrieved from https://www.mdpi.com/2032-6653/14/2/26 doi: 10.3390/wevj14020026

Flötteröd, G. (2017). A search acceleration method for optimization problems with transport simulation constraints. *Transportation Research Part B: Methodological*, *98*, 239–260.

Frank, P., Friedrich, M., & Schlaich, J. (2008). Betriebskosten von Busverkehren schnell und genau ermitteln. *Der Nahverkehr*, *11*.

Horni, A., Nagel, K., & Axhausen, K. (Eds.). (2016). *Multi-Agent Transport Simulation MATSim.* London: Ubiquity Press. doi: 10.5334/baw

Kaddoura, I., Kickhöfer, B., Neumann, A., & Tirachini, A. (2015). Agent-based optimisation of public transport supply and pricing: impacts of activity scheduling decisions and simulation randomness. *Transportation*, *42*, 1039–1061.

Kühnel, N., Thunig, T., & Nagel, K. (2018). Implementing an adaptive traffic signal control algorithm in an agent-based transport simulation. *Procedia Computer Science*, *130*, 894-899. Retrieved from https://www.sciencedirect.com/science/article/pii/S1877050918304484 doi: https://doi.org/10.1016/j.procs.2018.04.086

Maciejewski, M. (2016). Dynamic Transport Services. In *The Multi-Agent Transport Simulation MATSim* (pp. 145–152). Andreas Horni, Kai Nagel and Kay W. Axhausen. Retrieved from https://matsim.org/the-book

Neumann, A. (2014). A paratransit-inspired evolutionary process for public transit network design. Retrieved from https://depositonce.tu-berlin.de/bitstream/11303/4393/1/neumann_andreas.pdf

Thompson, G. M. (1996). A simulated-annealing heuristic for shift scheduling using non-continuously available employees. *Computers Operations Research*, *23*(3), 275-288. Retrieved from https://www.sciencedirect.com/science/article/pii/0305054895000127 doi: https://doi.org/10.1016/0305-0548(95)00012-7

Youssef, H., M. Sait, S., & Adiche, H. (2001). Evolutionary algorithms, simulated annealing and tabu search: a comparative study. *Engineering Applications of Artificial Intelligence*, *14*(2), 167-181. Retrieved from https://www.sciencedirect.com/science/article/pii/S0952197600000658 doi: https://doi.org/10.1016/S0952-1976(00)00065-8

Zilske, M., & Joubert, J. W. (2016). Freight Traffic. In *The Multi-Agent Transport Simulation MATSim* (pp. 155–156). Andreas Horni, Kai Nagel and Kay W. Axhausen. Retrieved from https://matsim.org/the-book

Zwick, F., Kuehnel, N., & Hörl, S. (2022). Shifts in perspective: Operational aspects in (non-)autonomous ride-pooling simulations. *Transportation Research Part A: Policy and Practice*, *165*, 300-320. Retrieved from `https://www.sciencedirect.com/science/article/pii/S0965856422002294` doi: https://doi.org/10.1016/j.tra.2022.09.001

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. Retrieved from `https://www.sciencedirect.com/science/article/pii/S1877050921007195` doi: https://doi.org/10.1016/j.procs.2021.03.083