Michal Maciejewski, Joschka Bischoff, Kai Nagel

An Assignment-Based Approach to Efficient Real-Time City-Scale Taxi Dispatching

Maciejewski, M.; Bischoff, J.; Nagel, K. (2016). An Assignment-Based Approach to Efficient Real-Time City-Scale Taxi Dispatching. IEEE Intelligent Systems, 31(1), 68–77. https://doi.org/10.1109/MIS.2016.2

An Assignment-Based Approach to Efficient Real-Time City-Scale Taxi Dispatching

Michal Maciejewski[1,2], Joschka Bischoff[2], and Kai Nagel[2]

[1] *Division of Transport Systems, Poznan University of Technology, Piotrowo 3, 60-965 Poznan, Poland*
[2] *Department of Transport Systems Planning and Transport Telematics, TU Berlin, Salzufer 17-19, 10587 Berlin, Germany*

This is a post-print of the original paper: M. Maciejewski, J. Bischoff and K. Nagel, "An Assignment-Based Approach to Efficient Real-Time City-Scale Taxi Dispatching," in IEEE Intelligent Systems, vol. 31, no. 1, pp. 68-77, Jan.-Feb. 2016.
doi: 10.1109/MIS.2016.2

1 Introduction

Traditionally, taxi-service research has been conducted from a long-term, strategic point of view and at the macroscopic level of detail. Issues such as competition and regulation, including fleet sizing or pricing, have been explored for more than 40 years [1, 2, 3]. Over the past decade, online management of taxis has become standard, and nowadays the focus is on operating a taxi fleet both at the microscopic level of detail and in real time. Due to advances in information and communications technology (ICT), various innovative solutions, such as shared taxis [4], electric taxis [5, 6], and shared autonomous vehicles [7], have reshaped the traditional taxi service.

Designing and implementing taxi and similar services requires the use of sophisticated dynamic routing algorithms that are hard to analyze theoretically. Thus, simulation tools are needed. However, the quality of results obtained with such tools depends on the simulation method. In transport-related problems, simulation must incorporate realistically modeled dynamism of customer demand, traffic-flow phenomena, and fleet-management operations. These aspects are even more crucial when considering taxi-like services, owing to the dynamics of urban traffic flow and the volatility of demand. As far as we know, of the many taxi simulation models built so far, the microscopic ones were created for Singapore [8, 9], Barcelona [10], Berlin [11], and Mielec, Poland [12]. This article presents the application of a wide-range microscopic model covering the city of Berlin and the neighboring Brandenburg region to assess the performance of a real-time dispatching strategy based on solving the taxi assignment problem.

2 Online Taxi Dispatching

Choosing the right simulator is of the utmost importance. Commercial micro- or nanosimulators, such as Vissim, Aimsun, Paramics, and DynusT, either are unable to perform network loadings with millions of persons or vehicles, or they do not trace persons or vehicles throughout the whole day. Noncommercial approaches include the Transportation Analysis Simulation System (TRANSIMS) [13], Simulation of Urban Mobility (SUMO) [14], Mezzo [15], and the Multi-Agent Transport Simulator (MATSim) [16]. Out of these, MATSim is arguably the least focused on traffic flow realism, but has the highest computing speed and the best behavioral model on the trip-planning side; therefore, we chose it as the simulation platform. To simulate taxi dispatching, we used MATSim's Dynamic Vehicle Routing Problem (DVRP) extension [17]. In this module, each driver is modeled as an agent whose schedule is dynamically reoptimized by the optimization algorithm in response to incoming events (such as request submissions and vehicle departures and arrivals). Fleet vehicles are simulated along with the whole traffic, and their movement is monitored in order to detect any divergences from the schedule (such as delays due to traffic congestion). Also, the interaction between the dispatcher, drivers, and passengers is simulated in detail, including such actions as calling a ride or picking up and dropping off passengers.

The following formulation of the taxi dispatching process describes the model implemented for the needs of this study. It is an extended version of an earlier model used for the assessment of rule-based dispatching strategies [11]. It deals with immediate requests with unknown destinationsthat is, taxis are not prebooked, and no destination information is provided in advance, which is typical for taxi services in Berlin.

Let $N = \{1, \ldots, n\}$ be the set of taxi requests (customers). The following sequence of events is related to serving each request $i \in N$ (see Figure 1). Taxi customer i calls a taxi (event E_i^{call}, time τ_i^{call}) specifying the pickup location p_i. Because only immediate requests are considered, the customer's desired departure time is $\tau_i^{\text{dep}} = \tau_i^{\text{call}}$. A selected taxi is dispatched toward p_i at time τ_i^{disp} (event E_i^{disp}), and immediately after arrival, the pickup starts (event E_i^{pick0}, time τ_i^{pick0}). Once the passenger is picked up (event E_i^{pick1}, time τ_i^{pick1}), he or she specifies the destination d_i and the taxi sets out immediately. After reaching d_i, the drop-off begins (event E_i^{drop0}, time τ_i^{drop0}). Once the passenger gets out (E_i^{drop1}, time τ_i^{drop1}), the taxi is ready to serve another request. Owing to the stochasticity of taxi dispatching, times $\tau_i^{\text{disp}}, \tau_i^{\text{pick0}}, \tau_i^{\text{pick1}}, \tau_i^{\text{drop0}}$ and τ_i^{drop1} are estimated

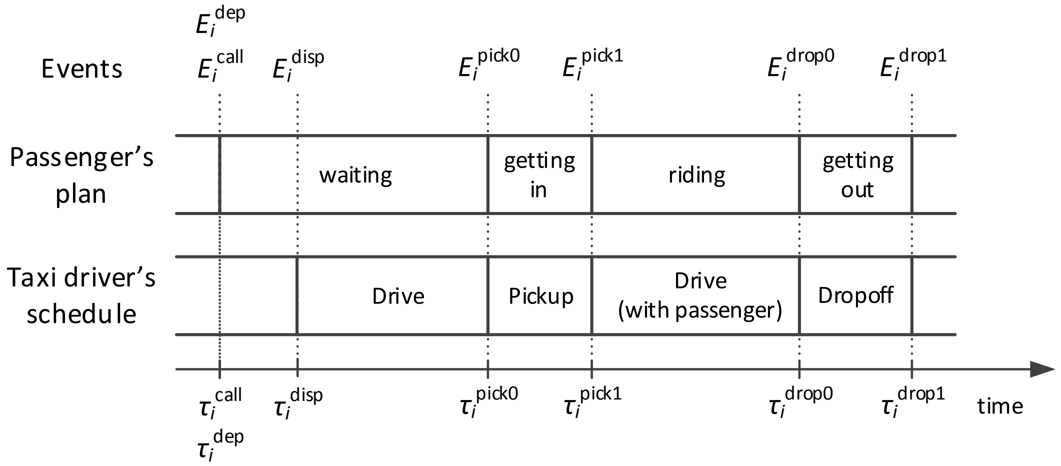

Figure 1: A taxi driver's schedule and a passenger's plan. A passenger has a daily plan made up of activities and legs, including the presented taxi leg. This leg is represented as a taxi request on the Dynamic Vehicle Routing Problem side and is handled by a sequence of tasks dynamically computed by the optimizer and executed by the driver.

until the respective events occur, and are therefore subject to change.

Request $i \in N$ is *open* if it either has not been planned yet (that is, τ_i^{disp} is unknown) or if it is planned to be served but the taxi has not been dispatched yet, $\tau_i^{\mathrm{disp}} > \tau^{\mathrm{curr}}$, where τ^{curr} denotes the current time. Let L be the list of all open requests ordered by τ_i^{dep}. Each request i is inserted into L on submission E_i^{call} and removed from L on taxi dispatch E_i^{disp}. We do not consider vehicle diversion in this study, so taxi dispatch is an irreversible operation.

Let $M = \{1, \dots, m\}$ be the set of vehicles. Each vehicle $k \in M$ is available at location o_k within the time window $[a_k, b_k)$. We assume that vehicles do not cruise, but remain at the last served customer's drop-off location. When no request has been assigned to k, o_k is k's initial location and a_k is the time the taxi starts operating. Otherwise, o_k is the drop-off location, d_i, and a_k is the time the drop-off ends, τ_i^{drop1}, of the last request assigned to k, i. Because d_i remains unknown until τ_i^{pick1}, both o_k and a_k are unknown temporarily as well, with the restriction that $a_k > \tau^{\mathrm{curr}}$. Let $M^{\mathrm{A}} \subseteq M$ be the set of all currently *available* vehicles, that is vehicles $k \in M$ such that o_k and a_k are known and $b_k > \tau^{\mathrm{curr}}$. In other words, while operating, a vehicle is cyclically unavailable between the time it sets out to a new customer and then picks the customer up. Let $M^{\mathrm{I}} \subseteq M^{\mathrm{A}}$ be the set of all currently *idle* vehicles; available vehicle $k \in M^{\mathrm{A}}$ is idle if $a_k \leq \tau^{\mathrm{curr}}$.

3 Rule-Based Dispatching Heuristics

Owing to past technological constraints, taxi services still use simple rule-based algorithms, such as first-come, first-served, to serve customers. In the era of radio taxi dispatching systems, the longest waiting taxi in the zone where a taxi trip starts was dispatched to the customer. Nowadays, because of recent ICT advances, the dispatching decision more often is based on the taxis' exact geographical locations, in order to send the nearest one. All these strategies seem myopic and far from optimal and do not exploit today's technology.

The most popular strategy, *nearest-idle-taxi*, dispatches the nearest idle taxi, $k^* \in M^{\mathrm{I}}$, to the first request in the queue L, denoted by $L[1]$, according to the following rule:

$$k^* = \underset{k \in M^{\mathrm{I}}}{\arg\min}\, t^{\mathrm{O}}_{k,L[1]}\left(\tau^{\mathrm{curr}}\right), \tag{1}$$

in which $t^{\mathrm{O}}_{ki}(t)$ is the travel time from o_k to p_i, given the departure time t. A dispatch decision is computed either when a taxi call is made and at least one vehicle is idle (oversupply) or when a taxi becomes idle (for example, starts operating or completes an earlier job) and at least one request is open (undersupply).

This strategy's weak point is a limited choice of taxis under high demand when no vehicle is idle, which results in ineffective, almost random, decisions. The performance of dispatching can be improved by extending this strategy with the capability of looking into the near future to predict availabilities of currently busy taxis in order to find the one that will arrive earliest at the customer's location. In this strategy, *nearest-taxi*, dispatch decisions are planned sequentially for each request in L according to the following rule:

$$k^* = \underset{k \in M^{\mathrm{A}}}{\arg\min} \left(\max\left(a_k, \tau^{\mathrm{curr}}\right) + t^{\mathrm{O}}_{k,L[1]}\left(\max\left(a_k, \tau^{\mathrm{curr}}\right)\right)\right), \tag{2}$$

and are updated whenever new information arrives (for example, a vehicle's availability prediction changes because of traffic delays, or a new request is submitted).

Despite this broadening of taxi choice, the nearest-taxi strategy still behaves myopically by processing requests sequentially according to first-in, first-out (FIFO) order. This becomes the main problem in the case of undersupply, where the main focus should be on providing the highest possible system throughput at the cost of violating the FIFO rule and thus losing some fairness. To partially mitigate this issue, it is necessary to shift the priority from customers to taxis. We can achieve this by modifying the behavior of the nearest-idle-taxi strategy in the case of undersupply by dispatching a vehicle that has just become idle, $k \in M^{\mathrm{I}}$, to the nearest open request, $i^* \in L$, according to the following rule.

$$i^* = \underset{i \in L}{\arg\min}\, t^{\mathrm{O}}_{k,i}\left(\tau^{\mathrm{curr}}\right). \tag{3}$$

This *nearest-idle-taxi/nearest-open-request* strategy is also called *demand-supply balancing*, because it tries to find a balance between demand and supply. It originates from a study by Pius Egbelu and Jose Tanchoco on scheduling of automated guided vehicles [18], in which the authors proposed order-initiated and vehicle-initiated assignment rules. In addition, more formal specifications of the three strategies presented in this section can be found in other work [11, 19].

4 Online Taxi Dispatching as the Assignment Problem

Although both extensions to the classic nearest-idle-taxi strategy lead to improvements in dispatching, the limited planning scope remains their apparent disadvantage, because dispatch decisions are optimized only locally by considering a single request or vehicle. To achieve higher-efficiency dispatching, we must take a more global look at its process. One solution could be to cyclically recalculate the globally best assignment of all available taxis to all open requests in response to new informationwhich we call the *assignment* strategy. This can be done by solving the linear assignment problem [20], in which variable x_{ki} represents the assignment of available vehicle $k \in M^{\mathrm{A}}$ to open request $i \in L$. The cost of serving request i by vehicle k, c_{ki}, is defined as the passenger wait time from τ^{curr} on, i.e.

$$c_{ki} = \max\left(a_k, \tau^{\mathrm{curr}}\right) + t^{\mathrm{O}}_{ki}\left(\max\left(a_k, \tau^{\mathrm{curr}}\right)\right). \tag{4}$$

If the number of open requests is larger than the number of available vehicles, $|L| > \left|M^{\mathrm{A}}\right|$, a set of dummy vehicles M^{D} must be included so that $|L| = \left|M^{\mathrm{A}}\right| + \left|M^{\mathrm{D}}\right|$. Assignment of a dummy taxi to a request means that the request will remain (temporarily) unplanned. The cost of assigning dummy vehicle $k \in M^{\mathrm{D}}$ to request $i \in L$, c_{ki}, is the cost of postponing i, and assumed to equal C. On the other hand, if the number of available vehicles is larger than the number of open requests, $|L| < \left|M^{\mathrm{A}}\right|$, a set of dummy requests L^{D} is included into the problem formulation so that $|L| + \left|L^{\mathrm{D}}\right| = \left|M^{\mathrm{A}}\right|$. Assignment of a taxi to a dummy request means that the vehicle will not be dispatched; if this vehicle is currently serving a request, it will become idle after completing it. The cost of not being dispatch is $c_{ki} = 0, k \in M^{\mathrm{A}}, i \in L^{\mathrm{D}}$.

Assignments are periodically recomputed in reaction to new request submissions E^{call}_i and upon provision of a trip's destination E^{pick1}_i. Reoptimization can also triggered by a change of the predicted taxi availability a_k.

5 Adaptation for Large-Scale Simulation

Although the idea of modeling taxi dispatching as the dynamic assignment problem seems natural, calculating the cost matrix for a large taxi fleet, such as the one in Berlin, poses a big computational challenge if we want to use this strategy for real-time taxi management. Because simple calculation of the cost for each vehicle-request pair separately would take a prohibitively large amount of time, we introduced the following enhancements:

- **Reduced reoptimization frequency.** The decision concerning rerunning optimization is made every T^{reopt} seconds, instead of every single second, which is the simulation time step. Besides the reduction in the amount of computation, it also makes dispatch decisions for larger sets of requests, which leads to better matchings in terms of overall system efficiency. The reduced reoptimization frequency has a minimal negative impact on the dispatching quality by introducing an extra delay of $T^{\mathrm{reopt}}/2$ seconds, which is negligible as long as T^{reopt} is reasonably small. In this study, T^{reopt} was 10 seconds.

- **Reduced planning horizon.** From all available vehicles, only those which are available within the time horizon T^{horiz} — that is, $M^{\mathrm{Ared}} = \{i \in M^{\mathrm{A}} | a_k \leq \tau^{\mathrm{curr}} + T^{\mathrm{horiz}}\}$ — are considered in the assignment procedure. The planning horizon should not be shorter than the period between two consecutive reoptimizations (T^{reopt}). In this research, the default value of T^{horiz} was 120 seconds. However, in the case of undersupply, $\left|M^{\mathrm{I}}\right| < |L|$ — that is, when the focus is shifted toward achieving a high throughput and thus not letting vehicles stay idle — T^{horiz} was reduced to 30 seconds.

- **Single-source to k-nearest-sinks (backward) shortest path search.** Because executing a shortest-paths search separately for each (non-dummy) vehicle-request pair is too slow, we used a multisink variant of Dijkstra's algorithm instead. It starts at a given source and ends after reaching a given number of sinks. If there are fewer idle and soon-idle vehicles than open requests, $\left|M^{\mathrm{Ared}}\right| < |L|$, the search starts from each vehicle's location and moves forward until reaching K^{req} nearest requests. If the opposite is the case, it is more efficient to start at each request's location and move backward until paths to the first K^{veh} are found. Limiting the search to only the subset of closest requests or vehicles implies that for many vehicle-request pairs, the travel time is unknown. The cost of such assignments is also unknown, and we must use a reasonably high constant value, T^{nopath}, instead. As long as the subsets are not too small, those assignments are unlikely to be selected, and therefore, this speedup technique does not affect the assignment-based strategy's performance. For the Berlin case study, we set K^{req} and K^{veh} to 40, whereas T^{nopath} was 172,800 seconds (2 days).

6 Taxi Berlin Scenario

At the time of this writing, 7,600 taxis are serving Berlin. Roughly 18,000 drivers are registered and organized in 3,000 taxi companies. There is no centralized taxi dispatch, but there are several radio taxi operators. In this article, we were able to utilize data of Taxi Berlin, the city's biggest radio taxi operator, with some 5,700 vehicles under dispatch. Most of the taxis are equipped with GPS trackers that submit their current location and occupation status at least once per minute. Apart from taxi dispatch, this data is also used for real-time travel time prediction in Berlin [21].

6.1 Supply and demand

The supply and demand data from one week (15 April to 22 April 2014) was available in this study, of which we chose the time frame between Tuesday 4:00 a.m. and Wednesday 4:00 a.m. for simulation. (Taxi demand and supply is the lowest around 4:00 a.m., so we took it as the cutting point between days.) We postprocessed this data to a zone-based

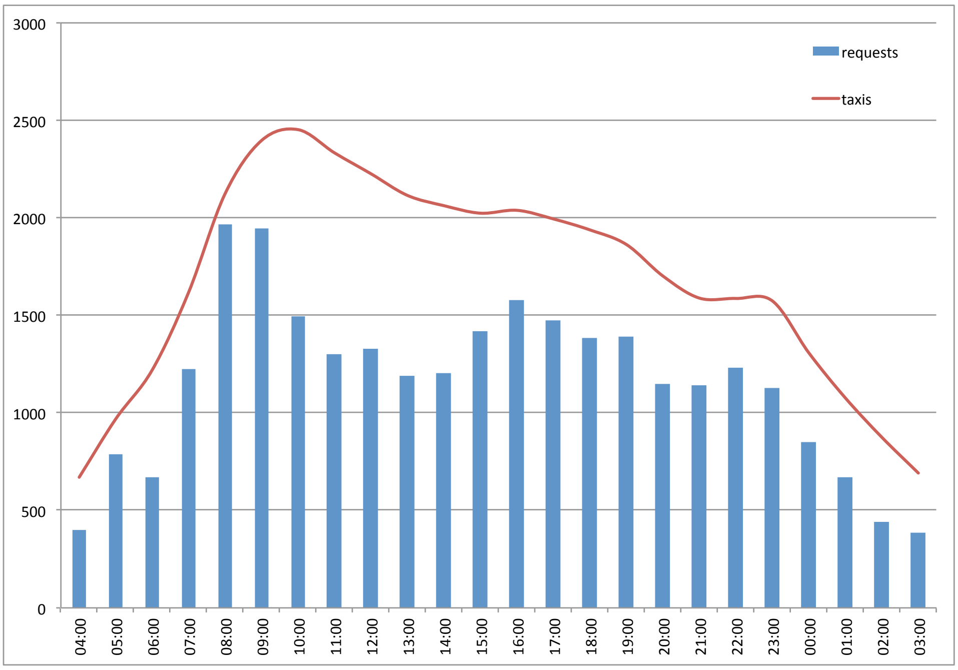

Figure 2: Hourly request submissions and active taxicabs. The average taxi serves less than one request per hour. Taxi drivers adapt to the current demand; both time profiles are highly correlated.

matrix of taxi trips for each simulation hour and used it as the simulated demand. On the supply side, neither actual driver nor shift durations were available, owing to an automatic data anonymization process that assigns vehicles new IDs every couple of minutes. However, we computed the amount of vehicles logged into the system at any time, as well as the amount of vehicles per zone in each occupation status in five-minute intervals. Figure 2 shows the amount of taxis and requests served during the simulation timeframe. Overall, 27,376 trips were registered; a strong morning peak was followed by two smaller peaks in the afternoon and evening. The number of available vehicles follows the demand.

We aggregated the extracted taxi demand into 518 zones. Within Berlin, these zones are standardized quarters defined by the city administration as *Lebensweltlich orientierte Räume (LOR)* [22], whereas in the surroundings, community boundaries are used. The hotspot for taxi traffic in Berlin is around the city's major airport, Tegel; the data showed 3,799 trips to and from here. Given Tegel's close distance to the city and no direct rail link, taxis make up a comparatively high share of trips to and from Tegel. Berlin's second airport, Schönefeld, which is farther from the city and caters mainly to budget airlines, is less relevant for the taxi business. Most other trips are either ending or beginning in the city center. Figure 3 shows the origin of taxi trips within the city.

In Berlin, there are roughly 400 taxi ranks in operation where taxis tend to aggregate. Figure 4 shows their locations and the average number of idle taxis in each zone. The rank at Tegel Airport is notably the one with the most idle vehicles, and taxis can wait

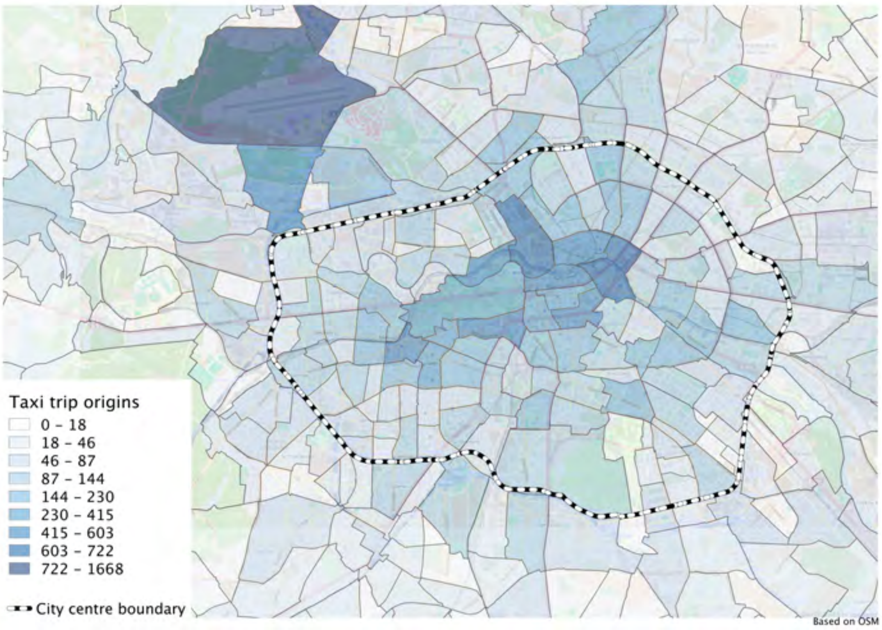

Figure 3: Daily taxi demand in Berlin by zone of trip origin. High demand, marked in dark blue, originates mainly in most central parts of the city center and at Tegel Airport.

there several hours for a ride. A more detailed analysis of the floating car data (FCD) has been published before [23].

6.2 Simulation in MATSim

For the MATSim simulation, we used a network based on OpenStreetMap data and comprising 11,353 nodes and 24,350 links. We generated time-specific link travel times using an earlier simulation [24], which let us keep the simulation runtime low, because we only needed to consider taxi traffic. We converted the extracted demand data from FCD into MATSim plans. Locations for origins (see Figure 5) and destinations (see Figure 6) within the zones were randomly distributed, as were the actual departure times within each hour. This resulted in 27,386 agent plans with exactly one taxi trip each. A simulation with these plans shows that there is very low stress on the taxi system and vehicles are idle for a long time. However, to accommodate a comparably high share of black-market (30 to 40 percent, according to estimations in [25]) and incorrectly tracked rides, we used a scaling factor of 1.5 for the taxi demand.

In certain situations, such as bad weather conditions, public transport breakdowns, or trade fairs or other big events, taxi demand can increase drastically. During a recent railway strike, the data shows that the number of taxi trips doubled during the afternoon peak, with the taxi supply increasing only by about 20 percent. To depict all these fluctuations, we decided to scale the demand step-wise up to 5.0 while keeping the original spatiotemporal distribution of requests. Figure 7 depicts a taxi's trips over the

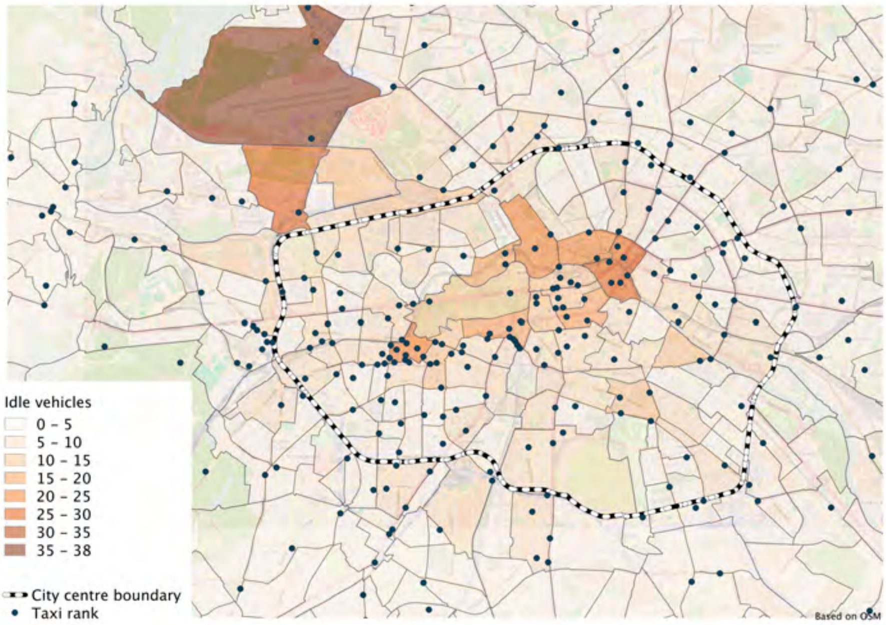

Figure 4: Average amount of idle taxis per zone. Attractive zones for waiting, marked in dark orange, are mainly around the rank at Tegel Airport and both the eastern city center (the area around Alexanderplatz) and the western city center (the area around the Zoo train station and Kurfürstendamm).

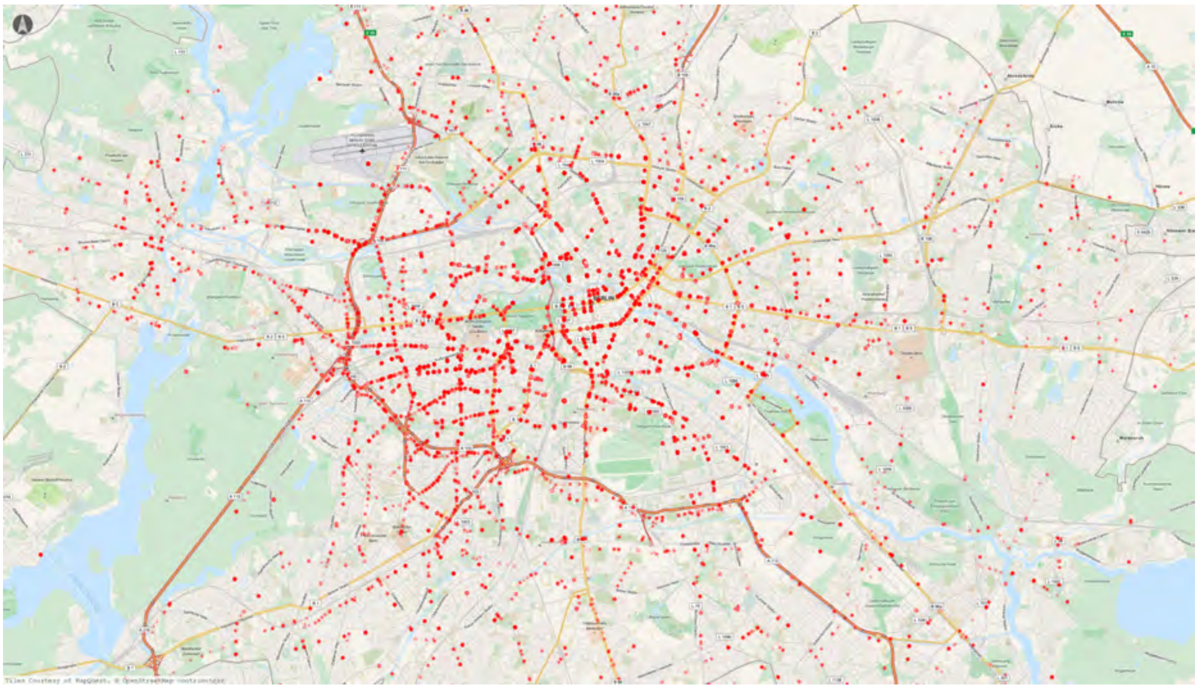

Figure 5: Departure locations of taxi trips in the simulation (red). The majority of taxi trips start either in the city center or at Tegel Airport (northwest).

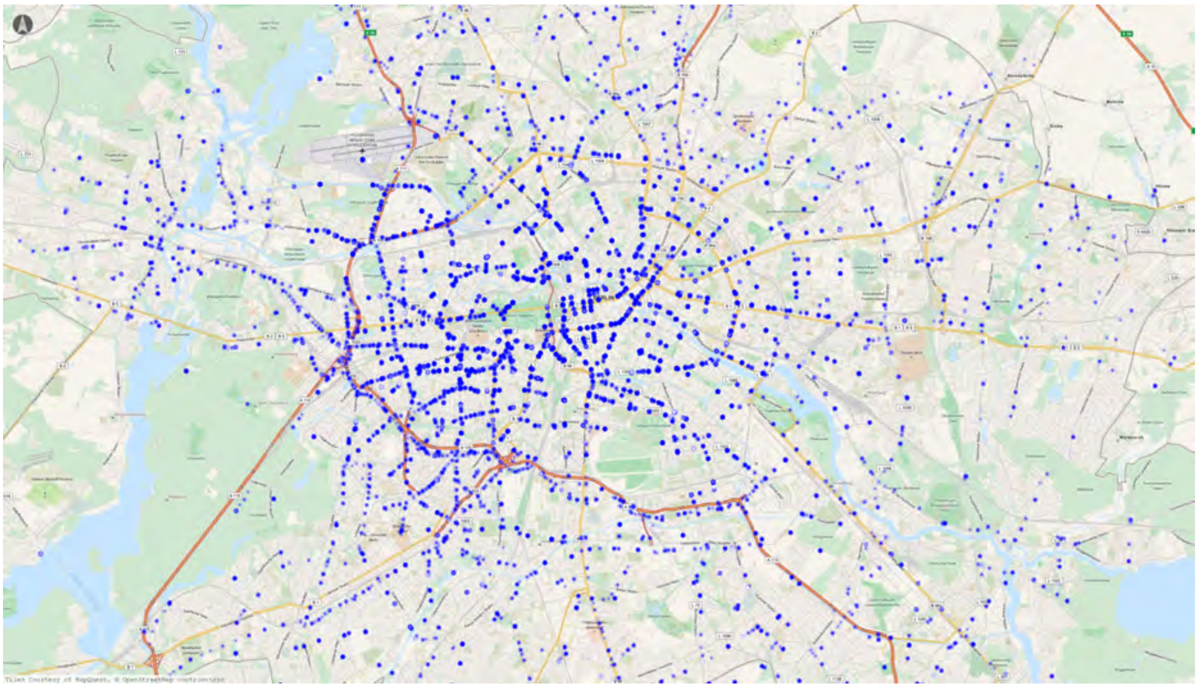

Figure 6: Arrival locations of taxi trips in the simulation (blue). Overall, the demand for taxi trips behaves rather symmetrically; that is, departure and arrival locations do not differ significantly.

course of the day. The vehicle has clearly distinguishable trips around the whole city. Just as in reality, one clear hotspot in the simulation is Tegel Airport. Figure 8 shows the area around it at 8:00 a.m. Several taxis have just started a trip toward the city center.

7 Results

We simulated the Berlin taxi scenario for each demand scale with three dispatching strategies: the nearest-idle-taxi strategy; the nearest-idle-taxi/nearest-open-request strategy, which can balance demand and supply; and the assignment strategy, which takes a more global look while making dispatch decisions.

We based our evaluation on the following performance measures:

- The average passenger wait time, T_{W}, in which the wait time of passenger i is defined as $\tau_i^{\mathrm{pick0}} - \tau_i^{\mathrm{dep}}$.

- The 95th percentile of passenger wait times, T_{W95}.

- The ratio of unoccupied (empty) to total drive time, R_{E}, in which the rides to and with passenger i take $\tau_i^{\mathrm{pick0}} - \tau_i^{\mathrm{disp}}$ and $\tau_i^{\mathrm{drop0}} - \tau_i^{\mathrm{pick1}}$ amount of time, respectively.

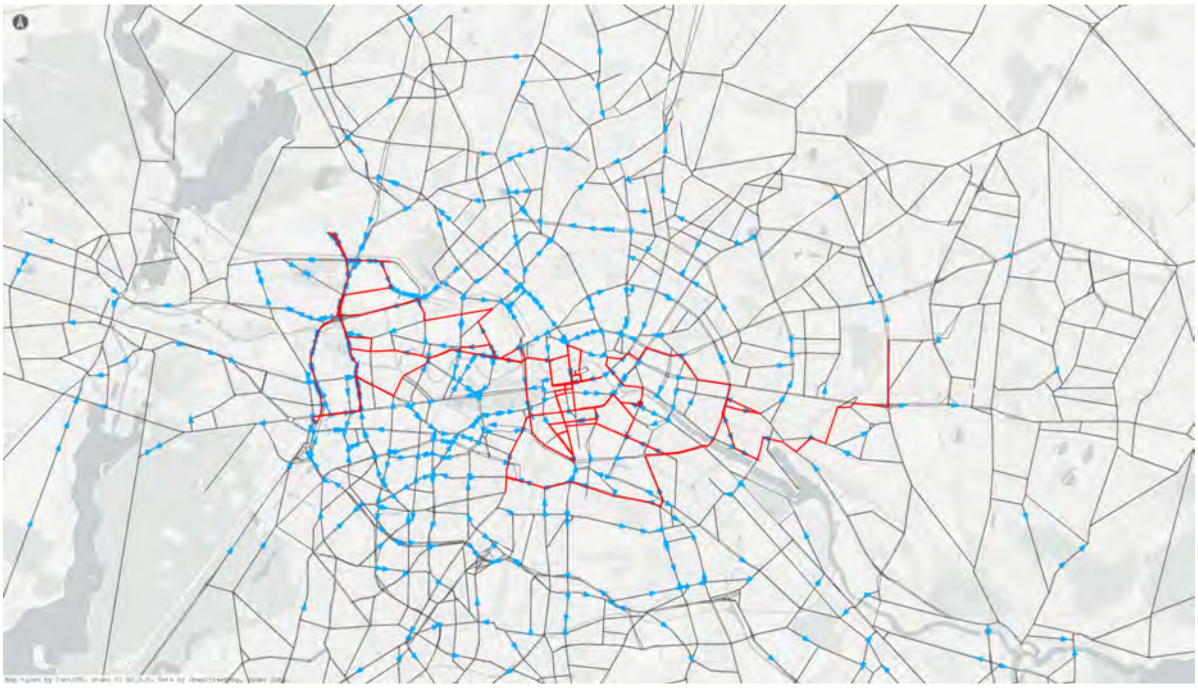

Figure 7: A simulated taxi's daily trajectory (red) and overall driving taxis (blue triangles) around 8:00 a.m. Vehicles tend to drive through the whole city as the demand goes.

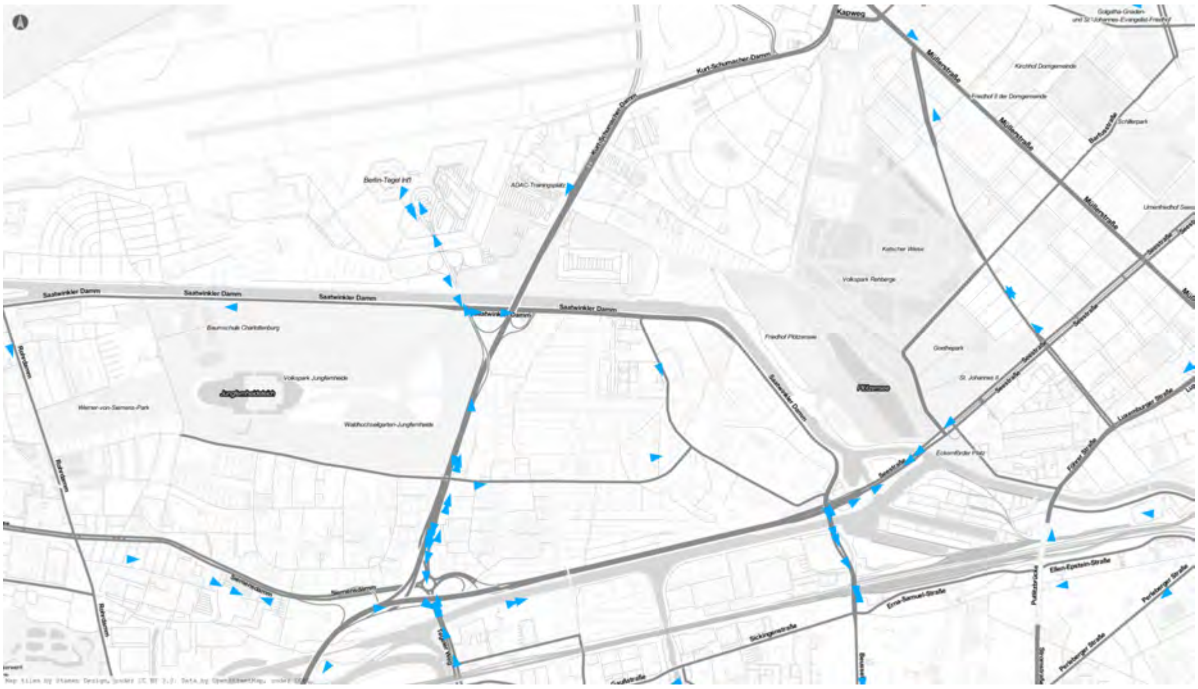

Figure 8: Taxi traffic (blue) around Tegel Airport during the morning peak. The airport is the most important hub for taxi traffic in the city, with a steady flow of pickups and drop-offs.

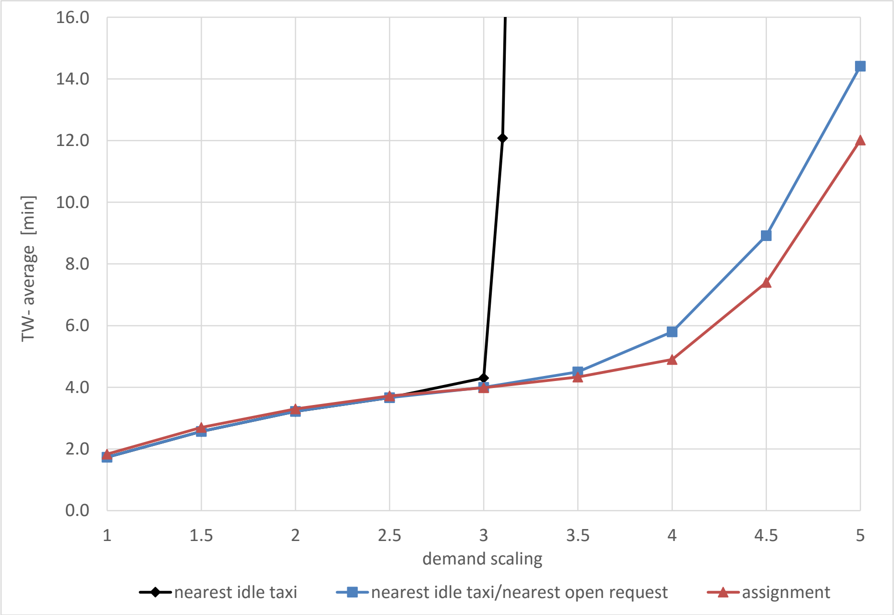

Figure 9: Average wait time, T_{W}, at different demand levels. Operation under high demand is selective for taxi dispatching strategies. When all taxis are busy, providing the highest possible throughput is the key to obtaining low wait times.

Figure 9 and 10 present the averages and 95th percentiles for passenger wait time at different levels of demand obtained with different strategies. Up to a scaling factor of 2.5, all strategies perform comparably. At low demand, both rule-based heuristics behave virtually the same way, and the assignment-based one does not offer any significant improvement, because open requests usually are served by the nearest idle vehicle. However, beyond 3.0, the FCFS way of handling requests results in a rapid increase of T_{W}, whereas shifting the focus from passengers to taxis assures an acceptable level of service quality even when the demand is five times higher. As the demand grows, the assignment strategy's advantages over the simpler demand-supply balancing strategy become more evident in terms of both passenger-oriented performance measures.

Figure 11 presents the ratio of empty to total drive time. As in the case of the passenger wait time statistics, the first strategy performs the worst. In particular, from the scaling factor of 3.2 on, trips to passengers are on average 50 percent longer than trips with passengers. At this level of demand, the FCFS rule dispatches vehicles randomly: whenever a taxi becomes idle, it is immediately dispatched to the longest waiting request, regardless of the distance between them. The two other strategies result in much lower shares of nonrevenue trips at high demand. In the worst case, pickup trips are only two times shorter than drop-off trips for the demand-supply balancing strategy, and almost three times shorter for the assignment-based approach. Better utilization of vehicles by the assignment strategy leads to a higher throughput and consequently to lower wait times. Interestingly, at high demand, R_{E} stabilizes and even decreases as the average

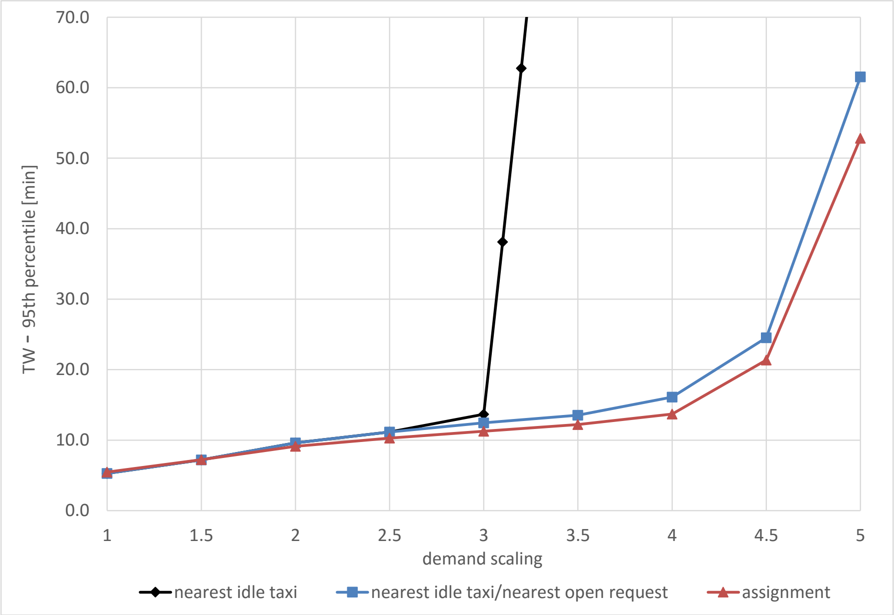

Figure 10: The 95th percentile of wait time, T_{W95}, at different demand levels. Although both the demand-supply balancing and assignment-based strategies shift priority from passengers to drivers by not following the FCFS rule, even the less-lucky passengers (represented here by the 95th percentile) are served quicker than if the rule is obeyed.

distance from vehicles to their nearest requests drops.

8 Conclusions

With ride-sourcing services such as Uber, Lyft, and Sidecar rapidly growing and entering new cities, the rather conservative taxi market is undergoing revolutionary changes. The growing competition has forced taxi operators to look for new opportunities of providing better services at lower prices. One possible way of improving the performance is to start thinking about a taxi fleet as a whole, and not as a collection of independent vehicles. The proposed assignment-based strategy is a step in that direction.

In the future, we will focus on incorporating location attractiveness into the assignment cost function to optimize the spatiotemporal distribution of idle taxis. We also plan to extend the formulation of the assignment problem to electric taxis by including short-term assignments of chargers to vehicles.

Acknowledgments

This article was cofunded by the Einstein Foundation Berlin, Germany, and the National Centre for Research and Development, Poland (grant PBS1/A6/11/2012).

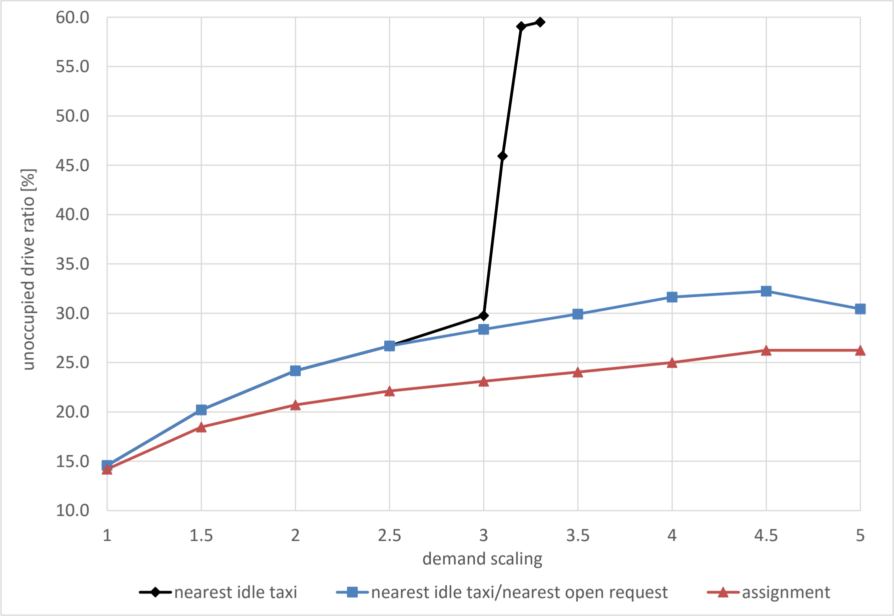

Figure 11: The unoccupied to total drive time ratio, R_{E}, at different demand levels. The high throughput offered by the assignment-based strategy results from the lowest share of the unoccupied drive time.

References

[1] G. W. Douglas, "Price regulation and optimal service standards: The taxicab industry," *Journal of Transport Economics and Policy*, pp. 116–127, 1972.

[2] R. D. Cairns and C. Liston-Heyes, "Competition and regulation in the taxi industry," *Journal of Public Economics*, vol. 59, no. 1, pp. 1–15, 1996.

[3] R. Arnott, "Taxi travel should be subsidized," *Journal of Urban Economics*, vol. 40, p. 316333, 1996.

[4] S. Ma, Y. Zheng, and O. Wolfson, "T-share: A large-scale dynamic taxi ridesharing service," in *Data Engineering (ICDE), 2013 IEEE 29th International Conference on*, pp. 410–421, April 2013.

[5] J.-L. Lu, M.-Y. Yeh, Y.-C. Hsu, S.-N. Yang, C.-H. Gan, and M.-S. Chen, "Operating electric taxi fleets: A new dispatching strategy with charging plans," in *Electric Vehicle Conference (IEVC), 2012 IEEE International*, pp. 1–8, March 2012.

[6] J. Bischoff and M. Maciejewski, "Agent-based simulation of electric taxicab fleets," *Transportation Research Procedia*, vol. 4, no. 0, pp. 191 – 198, 2014.

[7] D. J. Fagnant and K. M. Kockelman, "The travel and environmental implications of shared autonomous vehicles, using agent-based model scenarios," *Transportation Research Part C: Emerging Technologies*, vol. 40, pp. 1 – 13, 2014.

[8] D. Lee, H. Wang, R. Cheu, and S. Teo, “Taxi dispatch system based on current demands and real-time traffic conditions,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1882, no. -1, pp. 193–200, 2004.

[9] K. Seow, N. Dang, and D. Lee, “A collaborative multiagent taxi-dispatch system,” *Automation Science and Engineering, IEEE Transactions on*, vol. 7, no. 3, pp. 607–616, 2010.

[10] J. M. Salanova and M. Estrada, “Agent based modelling for simulating taxi services (case study in Barcelona),” *Procedia Computer Science*, vol. 52, pp. 902 – 907, 2015.

[11] M. Maciejewski and J. Bischoff, “Large-scale microscopic simulation of taxi services,” *Procedia Computer Science*, vol. 52, pp. 358–364, 2015.

[12] M. Maciejewski and K. Nagel, “A microscopic simulation approach for optimization of taxi services,” in *Proceedings of the 3rd International Conference on Models and Technologies for Intelligent Transportation Systems 2013* (T. Albrecht, B. Jaekel, and M. Lehnert, eds.), pp. 1–10, TUDpress, 2013.

[13] K. Nagel and M. Rickert, “Parallel implementation of the TRANSIMS microsimulation,” *Parallel Computing*, vol. 27, no. 12, pp. 1611 – 1639, 2001.

[14] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, “Sumo–simulation of urban mobility,” in *The Third International Conference on Advances in System Simulation (SIMUL 2011), Barcelona, Spain*, 2011.

[15] W. Burghout, H. Koutsopoulos, and I. Andrasson, “Hybrid mesoscopic-microscopic traffic simulation,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1934, pp. 218–255, 2005.

[16] A. Horni, K. Nagel, and K. W. Axhausen, eds., *The Multi-Agent Transport Simulation MATSim*. London: Ubiquity, 2016.

[17] M. Maciejewski, “Dynamic transport services,” in Horni *et al.* [16], pp. 151–160.

[18] P. J. Egbelu and J. M. Tanchoco, “Characterization of automatic guided vehicle dispatching rules,” *The International Journal of Production Research*, vol. 22, no. 3, pp. 359–374, 1984.

[19] M. Maciejewski, “Benchmarking minimum passenger waiting time in online taxi dispatching with exact offline optimization methods,” *Archives of Transport*, vol. 30, no. 2, pp. 67–75, 2014.

[20] H. W. Kuhn, “The hungarian method for the assignment problem,” *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[21] R. Ebendt, A. Sohr, L. Touko-Tcheumadjeu, and P. Wagner, “Dynamische Neuplanung der Touren von Express Trucks unter Einbeziehung einer FCD-basierten Verkehrslage,” in *Multikonferenz Wirtschaftsinformatik 2012 : Tagungsband der MKWI 2012*, pp. 124–136, 2012.

[22] Senate Department for Urban Development and the Environment, “Lebensweltlich orientierte Raeume (LOR) in Berlin,” 2015. http://www.stadtentwicklung.berlin.de/planen/basisdaten_stadtentwicklung/lor/.

[23] J. Bischoff, M. Maciejewski, and A. Sohr, “Analysis of Berlin’s taxi services by exploring GPS traces,” in *Models and Technologies for Intelligent Transportation Systems (MT-ITS), 2015 International Conference on*, pp. 209–215, June 2015.

[24] A. Neumann, *A paratransit-inspired evolutionary process for public transit network design*. PhD thesis, Technische Universität Berlin, 2014.

[25] S. Sauer, “Wie der mindestlohn das taxi-gewerbe veraendert,” 2014. http://www.berliner-zeitung.de/wirtschaft/mindestlohn-2015-in-deutschland-wie-der-mindestlohn-das-taxi-gewerbe-veraendert,10808230,29405460.html.