Network Fundamental Diagram Based Routing of Vehicle Fleets in Dynamic Traffic Simulations

Florian Dandl [1], Gabriel Tilg [2], Majid Rostami-Shahrbabaki [2], and Klaus Bogenberger [2]

[1]Technical University of Munich
[2]Affiliation not available

October 30, 2023

Abstract

The growing popularity of mobility-on-demand fleets increases the importance to understand the impact of mobility-on-demand fleets on transportation networks and how to regulate them. For this purpose, transportation network simulations are required to contain corresponding routing methods. We study the trade-off between computational efficiency and routing accuracy of different approaches to routing fleets in a dynamic network simulation with endogenous edge travel times: a computationally cheap but less accurate Network Fundamental Diagram (NFD) based method and a more typical Dynamic Traffic Assignment (DTA) based method. The NFD-based approach models network dynamics with a network travel time factor that is determined by the current average network speed and scales free-flow travel times. We analyze the different computational costs of the approaches in a case study for 10,000 origin-destination (OD) pairs in a network of the city of Munich, Germany that reveals speedup factors in the range of 100. The trade-off for this is less accurate travel time estimations for individual OD pairs. Results indicate that the NFD-based approach overestimates the DTA-based travel times, especially when the network is congested. Adjusting the network travel time factor based on pre-processed DTA results, the NFD-based routing approach represents a computationally very efficient methodology that also captures traffic dynamics in an aggregated way.

Network Fundamental Diagram Based Routing of Vehicle Fleets in Dynamic Traffic Simulations

Florian Dandl, Gabriel Tilg, Majid Rostami-Shahrbabaki and Klaus Bogenberger

Abstract— **The growing popularity of mobility-on-demand fleets increases the importance to understand the impact of mobility-on-demand fleets on transportation networks and how to regulate them. For this purpose, transportation network simulations are required to contain corresponding routing methods. We study the trade-off between computational efficiency and routing accuracy of different approaches to routing fleets in a dynamic network simulation with endogenous edge travel times: a computationally cheap but less accurate Network Fundamental Diagram (NFD) based method and a more typical Dynamic Traffic Assignment (DTA) based method. The NFD-based approach models network dynamics with a network travel time factor that is determined by the current average network speed and scales free-flow travel times. We analyze the different computational costs of the approaches in a case study for 10,000 origin-destination (OD) pairs in a network of the city of Munich, Germany that reveals speedup factors in the range of 100. The trade-off for this is less accurate travel time estimations for individual OD pairs. Results indicate that the NFD-based approach overestimates the DTA-based travel times, especially when the network is congested. Adjusting the network travel time factor based on pre-processed DTA results, the NFD-based routing approach represents a computationally very efficient methodology that also captures traffic dynamics in an aggregated way.**

I. Introduction

Ride-hailing and ride-pooling services enjoy increasing popularity in many cities worldwide. In order to evaluate the impact of such systems in a regulated or unregulated environment, traffic planners and policy-makers need to integrate these services into traffic models that often only include public transportation and private vehicles. With the inclusion of these mobility services into mode-choice models, predicting the impact of measures on the modal split becomes possible. Considering traditional modes, Dynamic Traffic Assignment (DTA) represents a method to compute the flows in multimodal networks. However, mode choice of individual travelers in the presence of the new mobility services depends on the level of service of the respective providers, mostly user waiting and service time (time until drop-off), as well as the price [1]. This level of service, in turn, depends on the status of the individual vehicles of the fleet, the network state and the routing strategies of the operator. The resulting vehicle routing problems are typically NP-hard integer problems and computationally very expensive, especially for larger street networks and fleet sizes that affect traffic dynamics in a city (e.g. [2]). The advancement of routing algorithms, e.g. by contraction hierarchies [3]–[5], enables fleet control in real-time, but traffic planners interested in the bigger picture and researchers not focusing on the routing aspect might want to use even faster routing algorithms to simulate more scenarios in the same time. Since advanced fleet control is computationally very demanding, integrating the fleet routing problem into a DTA further limits the number of scenarios that can be investigated [6], [7].

Florian Dandl, Gabriel Tilg, Majid Rostami-Shahrbabaki and Klaus Bogenberger are with the Chair of Traffic Engineering and Control, Technical University of Munich, Germany. Corresponding author: florian.dandl@tum.de

Mobility service operators require a lot of routing function calls in order to respond to requests of travelers. Based on the current estimation of travel times, the operator generates hypothetical routes between vehicle locations and each request's pick-up point and another point-to-point route from the latter to the traveler's desired drop-off point. Depending on the vehicle-customer assignment, repositioning and charging policies as well as re-assignment constraints of the operator, even a single vehicle might generate many routing function calls before a route is finally locked [8]–[10].

In order to gain fast routing calls, some studies focusing on fleet operation ignore network dynamics and pre-process the street network in the free-flow state to create travel time tables between all nodes [11], [12]. This pre-processing step also costs some time, but only has to be done once as long as travel times remain constant. By pre-processing the network based on travel times from DTA for multiple time intervals, exogenous traffic congestion can be represented [13]. However, this approach cannot capture traffic dynamics generated by increased traffic efficiency due to pooling or induced demand from mode-choice decisions [14].

In simulations with endogenous capacity utilization and travel times, the loops shown in Fig. 1 iterate multiple times. Therefore, the importance of efficient routing functionality for simulation frameworks becomes evident. In large-scale networks, finding an approximate but very efficient routing solution can be preferable over a more exact but costlier approach.

This paper combines very efficient routing based on travel time tables with the capabilities of network fundamental diagrams (NFD), also called macroscopic fundamental diagram, to capture city-wide traffic states in an aggregated fashion. Thereby, we trade off a more realistic routing for substantial gains in computational efficiency. As travel times influence the mode choice of travelers much more than the exact sequence of route segments, we present a comparison of origin-destination travel times based on a DTA to quantify the mentioned trade-off. We expect that the performance of this approach strongly correlates with the degree of scatter in the NFD. Hence, scenarios with varying network sizes and

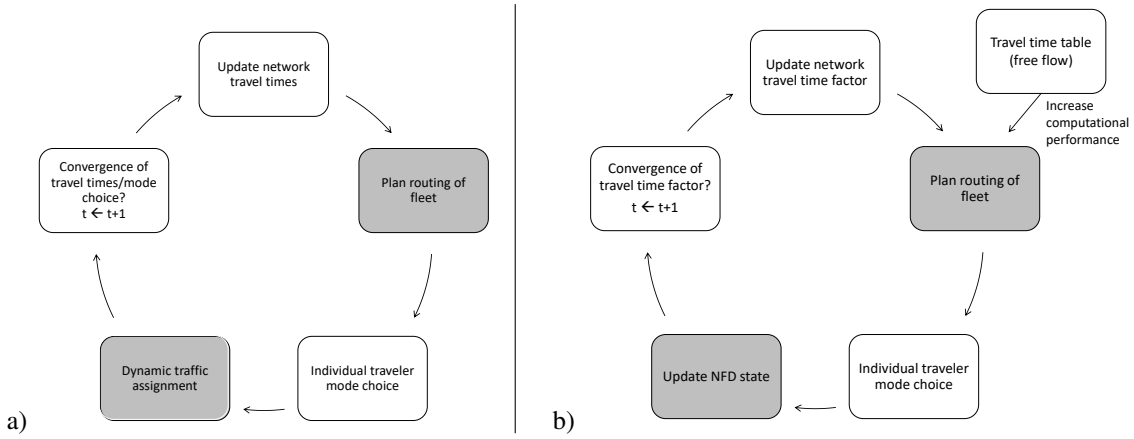

Fig. 1. Possible workflows of traffic and on-demand fleet simulations with endogenous demand from a traveler mode choice model. a) illustrates an DTA-based approach and b) an NFD-based approach. The processes of interest for this study are highlighted in grey.

origin-destination distributions are analyzed in a case study for Munich, Germany.

The remainder of this paper is structured as follows. First, we describe the problem considered in Section II. Subsequently, Section III elaborates on the general workflow, the travel time estimation based on the DTA and the NFD, as well as on the used routing algorithms. A case study is described in Section IV and its results are presented and discussed in Section V. Finally, Section VI draws conclusions and outlines future research.

II. Problem Description

The main goal of this paper is the comparison of routing approaches that can be used to simulate fleet coordination within a transportation network graph $G = (N, E)$ with nodes N and edges E. The edge travel times are assumed to be depending on the outcome of travelers' mode choice and therefore endogenous simulation parameters. In the first approach, which we denote by DTA-based approach, travel times t_e^i for each edge $e \in E$ are extracted from a DTA valid for time interval i. The second, NFD-based approach uses travel times $f^i \cdot t_e$, where t_e denotes free-flow travel times and f^i is a network travel time factor for time interval i derived from the NFD. The computational effort and the resulting routes are compared.

Instead of the edge sequence of a vehicle's route, it is often considered as sufficient to estimate the origin-destination (OD) travel times (or distances) during the route planning phase. The mode choice of hypothetical travelers is mostly affected by this OD travel time, while the network state will be influenced by the actual route. Hence, we compare both complete routing calls (returning the complete list of sections) and travel time calls (returning only the travel time between two nodes) for the DTA- and the NFD- based approaches for both use cases.

The routes between stops, i.e. the pick-up and drop-off locations, represent one-to-one routing problems. Additional to these one-to-one routing calls, on-demand fleet operators might use many-to-one routing calls to determine which vehicles are in the network vicinity and available to serve a new customer. Instead of this procedure, it is also possible to use some filter criterion (e.g. geometric vicinity [15]) and then call multiple one-to-one routing calls. For this study, the computational effort is only examined for one-to-one routing calls, together with the required time for network evaluation.

III. Background and Methodology

A. Workflow of Transportation Models

The workflow of the intended NFD-based transportation model that motivated this study (shown on the right of Fig. 1) is similar to other, typically DTA-based transportation models. This includes on-demand mobility fleet models implemented in MATSim [16] or mobitopp [17] (shown on the left of this Figure). On-demand fleet operators need to query a very large number of calls to a routing engine based on the last network state to plan the movements of their fleets and generate realistic offers. Travelers choose their mode (e.g. private vehicle, public transportation, on-demand provider) and thereby affect traffic states across the entire network. However, there are some aspects of our proposed NFD-based method that lead to substantial computational performance improvements. First, routing calls can be executed much faster by pre-processing, which has a large impact due to the large amount of routing calls. Secondly, the update of network-wide traffic states is much simpler: instead of keeping track of every single vehicle and traveler, the NFD merely requires to trace the total number of vehicles in the network. Moreover, this efficiency gain allows us to update travel times in every time step of the simulation and guarantees consistency of network-wide traffic states and route planning. Contrarily, DTA-based approaches are typically applied for longer time intervals and the work flow for a single interval might require several iterations before the routes are planned on a convergent network state.

In this study, we compare one iteration (at time interval instead of daily level) of DTA-based and NFD-based simulation workflows and focus on these differences (marked grey in Fig. 1). For a very large number of one-to-one routing

calls, the computational effort and quality of solution are evaluated, whereas the DTA-based approach is assumed to return the correct route and route travel time. Since the NFD and the free-flow travel time table can be pre-processed, these steps do not count into the computational effort of an iteration.

B. Dynamic Traffic Assignment

Dynamic Traffic Assignment models represent the complex dynamic characteristics of a transport network and describe the transition of traffic states and the evolution of recurrent and non-recurrent congestion. OD flows, for instance, may be used as a main input on the demand side, and are loaded into the network along with various parameters such as capacities and the driving behavior of travelers that capture the performance of the network on the supply side. In essence, the role of DTA is to provide an interaction between the demand for mobility and the network supply which results in traffic propagation considering the capacity of the traffic network. An excess of demand leads to congestion and delays experienced by the drivers. As a consequence, drivers may reconsider their route of travelling or shift to alternative modes of transportation [18]. DTA models are conventionally categorized into two wide-ranging groups: analytical models and simulation-based models. Simulation-based DTA models use a traffic simulator to replicate the complex traffic flow dynamics which are critical for developing meaningful operational strategies for real-time deployment. The use of a traffic simulator eliminates the need of any analytical formulations and provides a solution procedure for general networks [19].

Simulation-based DTA systems are particularly suited to evaluate a wide range of Advanced Traffic Management Systems (ATMS) and Advanced Traveler Information Systems (ATIS) [20]. With an increasing availability of computational power, DTAs have been substantially evolved in the last decade to evaluate network performance in various applications. The critical objective is to model the real-time evolution of travel demand and network conditions in the context of a DTA system and capture the within-day dynamics. There are various examples of successful adoption of simulation-based DTA in literature for traffic planning, route planning, and for control strategy generation [21]–[23].

C. Network Fundamental Diagram

The NFD describes the functional relationship of network production and network-wide vehicle accumulation. Early ideas were introduced in [24]–[26]. The macroscopic explicit function of the network-wide outflow and the aggregated accumulation was formulated by [27] and verified in [28]. The authors found a well-defined and low-scatter relationship of the space-mean flow and the network accumulation for empirical data from Yokohama. Amongst other domains of applications in traffic control and management, the NFD has been applied for route guidance, see for example [29]–[31].

In this paper, we aim to exploit the NFD for routing purposes as well. More specifically, we derive estimates for a scaling factor of free-flow travel times that approximate dynamic travel times in the network. For the sake of simplicity and computational performance, we scale the travel time on all edges of the network with the same factor. Therefore, the fastest routes are unaffected and can be completely pre-processed and saved in a travel time table. Dynamically changing travel times reflecting congestion effects are adjusted by scaling with the network-wide factor. We are aware that not partitioning a large network might increase the observed scatter in the NFD [32].

For this study, the NFD is derived from the results of the DTA. Please note that one advantage of the NFD-based approach is that no DTA is generally necessary. The NFD can be estimated based on loop detector or floating car data [33], [34]. In the absence of data, analytical estimation methods can be applied to derive the NFD [35]–[38]. For each time interval i, we extract the average flow q_e^i [veh/h] and density k_e^i [veh/km] for each edge e in the network based on the simulation outputs. We calculate the link-weighted averages as suggested in [28] as follows:

$$\mathbf{q^i} = \frac{\sum_{e \in E} q_e^i l_e}{\sum_{e \in E} l_e} \tag{1}$$

$$\mathbf{k^i} = \frac{\sum_{e \in E} k_e^i l_e}{\sum_{e \in E} l_e} \tag{2}$$

where l_e is the length of link e. Based on $\mathbf{q^i}$ and $\mathbf{k^i}$ for each time interval i we derive an average speed $\mathbf{v^i} = \mathbf{q^i}/\mathbf{k^i}$ for a certain average density in the network, which then can be used to derive an average travel time in the network. This average travel time serves as input for the vehicle routing problem described in the next section.

D. Routing Algorithms

Since the introduction of Dijkstra's routing algorithm [39], more memory, faster CPUs and algorithms exploiting the improved hardware made great leaps. As an example, a bidirectional Dijkstra with two priority queues in memory can reduce the number of nodes that have to be explored to find the optimal route. Estimating the costs to the destination during the local search process also allows to reduce the number of explored nodes [40]; this approach is called A* and a typical heuristic for shortest path problems is the Euclidian distance between the next node and the destination. Instead of the Euclidian distance, nodes of the network can be used as landmarks to derive a heuristic; this approach is called ALT (A*, Landmark, Triangle inequality) [41]. The more recent ideas of highway and (customizable) contraction hierarchies allow solving one-to-one routing problems even for the network of the whole of Europe with 42 million edges in milliseconds, which enables every-day use of routing queries in map applications [3], [5], [42].

The application of advanced and more efficient routing algorithms also increases the performance of transportation simulation models. Recently, Schneck and Noekel integrated customizable contraction hierarchies in the bi-conjugate Frank-Wolfe algorithm [43]. For their use case, this approach

achieves a speedup by a factor of 42 compared with a typical Dijkstra implementation.

Of course, having a complete travel time table available reduces the effort to retrieve travel time information to a lookup in a matrix, which is by far faster than any routing algorithm can be. For city-scale networks with a node count in the range of 1,000 to 100,000, it is still feasible to save one of these matrices for the free-flow case. Actual routing, i.e. determining the sequence of edges a vehicle has to drive to get from A to B, is also as efficient as it can be: an exact landmark is available for every neighboring node during the search process. Hence, not a single node that is not on the optimal route has to be explored and no priority queue is required. This reduces the number of travel time lookups to $N_r \cdot N_n$, where N_r is the average number of nodes in a route and N_n is the average number of neighbors of a node. As N_n is typically not higher than 3 in street networks, the complexity of the routing algorithm run time scales linearly with the number of nodes, i.e. $O(|N|)$. For the remainder of this paper, this routing solution is denoted by *Lookup Dijkstra*. Creating the travel time table can be performed with the Floyd-Warshall algorithm [44], [45] within minutes for a typical street network (i.e. a sparse adjacency matrix) for city scale.

Remark regarding the evaluation of computational effort: routing function calls in this study include some overhead compared to node-to-node routing as the functions are taking points on edges as input as well. This is necessary to enable en-route rerouting of fleet vehicles.

E. Approaches for Routing in Network Simulations with Endogenous Travel Times

This study compares following routing approaches for simulation frameworks with endogenous edge travel times (as illustrated in Fig. 1):

a) a DTA-based workflow with the use of bidirectional Dijkstra algorithm
b) an NFD-based workflow with Lookup Dijkstra

For the NFD-based approach, the Floyd Warshall is pre-processed based on free-flow travel times. In order to show the effect of network dynamics on travel times, a network travel time factor scales the travel times. For this study, we assume that the travel times within the network scale linearly with the inverse of the network average speed $\mathbf{v^i}$. The network travel time factor f^i is chosen as the inverse ratio of the current average speed during a time interval and a network average speed v_0, which should reflect the free-flow condition.

$$t_e^i = f^i \cdot t_e \tag{3}$$

$$f^i = \frac{v_0}{\mathbf{v^i}} \tag{4}$$

In a study area with homogeneous speed limits, the choice $v_0 = \max_i(\mathbf{v^i})$ is reasonable. For this study, the network average velocity at 05:00 is used because the maximum value of the network average velocity is not representative of the average of free flow velocities. With large speed limit differences (or even no speed limit on motorways), a few vehicles on these high-speed roads can generate unrepresentative network-average velocities during the night. The maximum value of a 5-minute interval can be used if the speed limits are homogeneous in the network.

Fig. 2. Studied street networks of the city of Munich. The three areas (polygon color) are denoted as A99 (pink), B2R(red) and Schwabing (blue). Highways and major roads are highlighted in violet and green, respectively.

IV. Case Study

For the case study, three areas of a large-scale network of Munich, Germany, i.e. A99 area spanning 475 km^2, B2R area spanning 49 km^2 and Schwabing area spanning 4.5 km^2 are used as illustrated in Fig. 2. We utilize a micro-simulation model of the network developed in Aimsun micro-simulator [46], [47]. It consists of 37,650 street sections (about 4,962 km in length) with 16,790 nodes [7]. After reducing the network to the minimum routing network, the networks contain 8,380 nodes with 20,574 edges (A99 area), 5,818 nodes with 14,642 edges (B2R area) and 959 nodes with 2,364 edges (Schwabing area).

A 24-hour simulation was performed with break points at 06:00, 10:00, 14:00 and 18:00. We recorded the edge travel times every 5 minutes and created one NFD data point according to equations 1 and 2, as well as the average network speed derived from it.

Since demand distribution is quite different for the morning and evening hours in Munich, we analyzed the travel times from the DTA- and NFD-based methods in the morning (05:00 - 10:00) and afternoon/evening (14:00 - 19:00) hours separately. For the evaluation of computational effort and quality of the solution, 10,000 OD pairs were chosen at random for each study area.

The routing algorithms were implemented in Python 3.7 and computations were performed on an Intel® Core™ i7-8665 CPU with 48 GB RAM.

V. Results and Discussion

A. Network Fundamental Diagram

Based on the DTA, we estimated the NFD as described in Section III-C. We applied a locally estimated scatter plot smoothing (LOESS) regression [48] to the data set to reduce the scatter. The LOESS method allows us to keep the

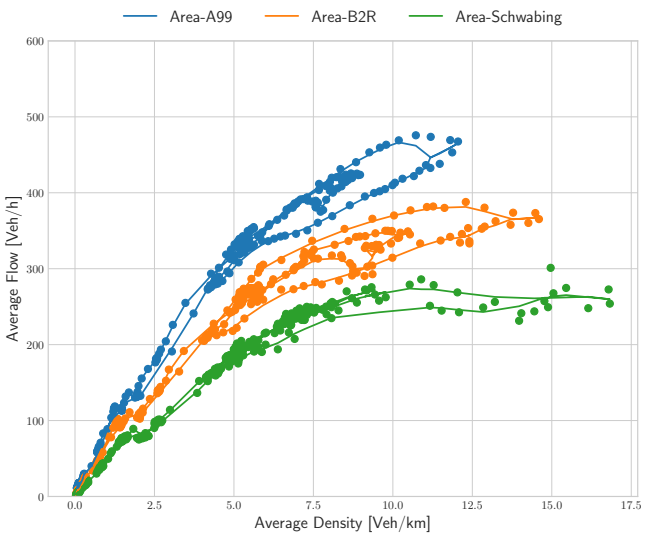

Fig. 3. LOESS regression to NFD data points generated from DTA 5-minute interval results according to equations (1) and (2).

TABLE I
COMPUTATIONAL EFFORT FOR TRAVEL TIME (TT) AND ROUTE FUNCTION CALLS IN THE MUNICH AREA-A99 SCENARIO

Number Queries	Unit	10^4	10^5	10^6	10^7
CPU Time TT/Route DTA	[h]	0.07	0.42	3.9	39.2
CPU Time TT NFD	[sec]	0.29	2.9	28.8	288
Speedup Factor TT NFD		911	536	499	495
CPU Time Route NFD	[min]	0.02	0.23	2.3	23.3
Speedup Factor Route NFD		153	90	84	83

temporal evolution of the data. Thus, this procedure is not neglecting any hysteresis phenomena. The results are shown in Figure 3. The x-axis shows the average network density in veh/km, and the y-axis displays the average network flow in veh/h. We show the NFDs for an entire day for each of the studied regions. The displayed NFDs regard to both the loading and unloading of the network. Generally, we observe clock-wise hysteresis loops for each network studied. That is, that the production is higher during loading periods than during unloading periods. Moreover, we see that the maximum production increases with the network size. The reason for this is the fact that the larger networks include more arterials and city-highways with higher speed limits. This leads to a higher production.

Moreover, we can distinguish between a larger hysteresis loop which occurs during the morning peak and a smaller one at the afternoon/evening peak for the two larger areas. For 'Schwabing', the hysteresis loops are small. This can be retraced to the fact that the network is not fully unloading after the morning peak, and thus the effect of loading and unloading is smaller for the evening peak.

B. Computational Effort

The main motivation of the NFD-based approach is to reduce computational effort. Such an improvement for function calls returning the travel time only is given by design as a simple lookup will always be much faster than

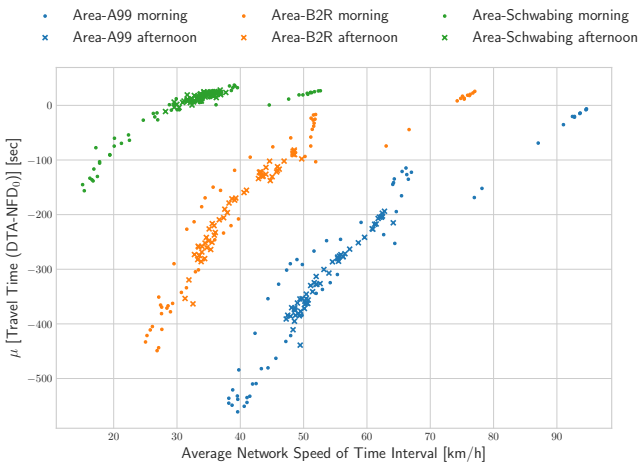

Fig. 4. Mean difference between DTA- and NFD-based approaches to determine travel times for all OD-pairs over the average network speed (one point per time interval).

any routing algorithm making multiple travel time lookups. Additionally, the computation times of processes that are reiterated according to the workflow need to be considered. For the A99 area, the computation time of the DTA lasted approximately 2 minutes per 5-minute time interval. This additional contribution dominates the total computation time for a smaller number of travel time or routing queries. As the NFD-based approach does not require the DTA simulation in each workflow iteration, the speed-up factor for a rather small number of queries is larger than 1,000. With an increasing number of routing calls the speedup factor converges to the ratio of average computation time for a bidirectional Dijkstra over a simple table lookup, which is approximately 500.

Next, we analyzed the computational gain when actual routes, i.e. the sequence of edges, are computed. As mentioned in section III-A, the computation times of the DTA workflow has a constant component from the computation of the DTA. The component dependent on the number of routing function calls is determined by the average route computation time, which we derived from the average value of the 10,000 sample OD-pairs in the A99 scenario. In this case, the NFD-based approach with Lookup Dijkstra generates a speedup between 80 and 150. Since routing computations based on Dijkstra's algorithm already last about 20 minutes per 5-minute time interval for 100,000 routing calls, the importance of routing for the total computation time becomes apparent.

C. Origin-Destination Travel Times

For the evaluation of the network travel time factor and the resulting OD travel times, the fastest path travel times were determined for each OD-pair in the chosen OD-pair samples. For each time interval, the average of the differences between the OD-travel times from the DTA-based and the NFD-based approach is computed and form one data point in Fig. 4. The first observation is that the average network speeds (x-axis) are higher for the larger areas. While the Schwabing

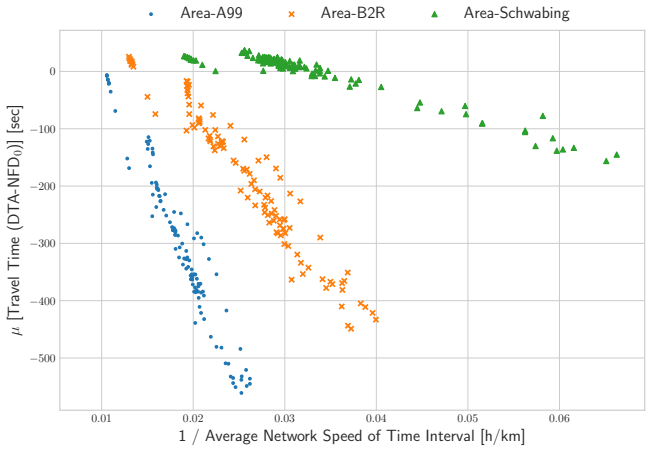

Fig. 5. Mean difference between DTA- and NFD-based approaches to determine travel times for all OD-pairs over the inverse of average network speed (one point per time interval).

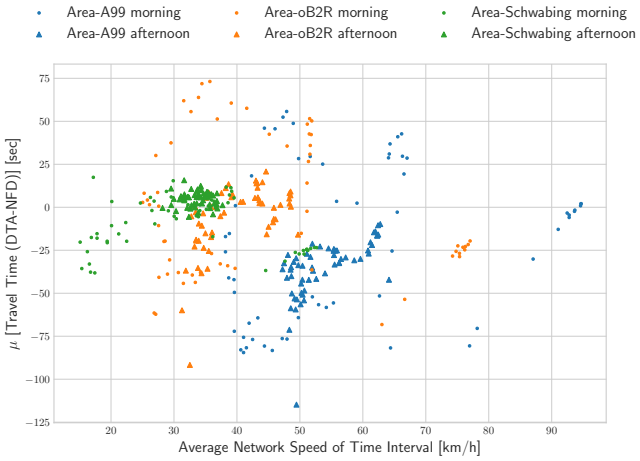

Fig. 6. Mean difference between DTA- and NFD-based approaches to determine travel times for all OD-pairs in the Area-A99 scenario over the average network speed (one point per time interval)

area mostly consists of residential streets with low speed limits, the B2R area contains the city's inner highway belt, which has 2-3 lanes, a speed limit of 50 and 60 km/h and therefore supplies a lot of traffic. The A99 area also contains the outer highway belt, which has 3-4 lanes and a speed limit of 120 km/h during the day. Furthermore, Fig. 4 depicts the consistency for the morning and evening hours. The more homogeneous the area is, the better the match. However, routes computed from the NFD approach are larger than their counterparts from the DTA approach in all cases. The difference increases with decreasing average speed within the network. The hyperbolic shape (especially for the Schwabing) of the difference over the average network speed motivated the plot of the difference over the inverse average network speed in Fig. 5.

The functional form and the choice of the maximum network average speed in the NFD as v_0 in equation 4 are a first approach to determine the network travel time factor f^i, but resulted in too large f^i values. Reducing v_0 to account for the different free-flow speeds across the network would lead to lower slopes of the curves illustrated in Fig. 5. This could substantially reduce the difference in travel time as these are linear for the most part. Considering the non-zero y-intercept of the curves, a functional form:

$$f^i = v_1 \left(\frac{1}{\mathbf{v^i}} - \frac{1}{v_2} \right) \tag{5}$$

We apply a least-squares fitting procedure to determine the coefficients v_1 and v_2. Then we recompute the difference in travel times for all OD-pairs in the sample with this network travel time factor. Fig. 6 shows that the mean of the travel times are matching much better for both approaches. For comparison, the average of all trip times in Area-A99 is more than 900 seconds.

As the means are matching better after the fit, the standard deviations can be analyzed for further insight. The variance/standard deviation of the travel time differences for all OD-pairs is a measure for homogeneity within the network.

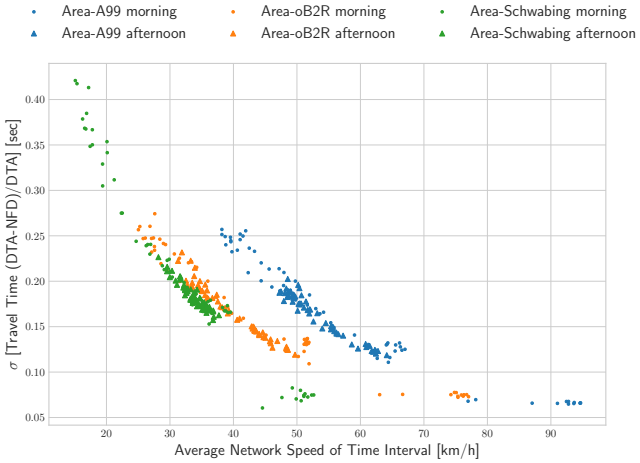

Fig. 7. Mean standard deviation of the relative difference between DTA- and NFD-based approaches to determine travel times for all OD-pairs in the Area-A99 scenario over the average network speed (one point per time interval).

In order to classify the results between the different areas, we show the standard deviations of the relative differences. As expected, the standard deviation increases for lower average network speeds (see Fig. 7). Interestingly, the relative deviations are largest for the smallest and (infrastructure-wise) most homogeneous area. The total standard deviation reaches up to 150 seconds for the smallest area (Schwabing), 200 seconds for B2R and 300 seconds for the largest area (A99).

All in all, the fit increases the quality of the model quite a lot. However, choosing the network travel time factor according to equation 5 requires a DTA, whereas the choice of equation 4 is valid independently of the database on which the NFD is estimated. The computational efficiency discussed in Section V-B can be achieved independently of the method to determine f^i as the fit of v_1 and v_2 can be performed prior to the routing simulations.

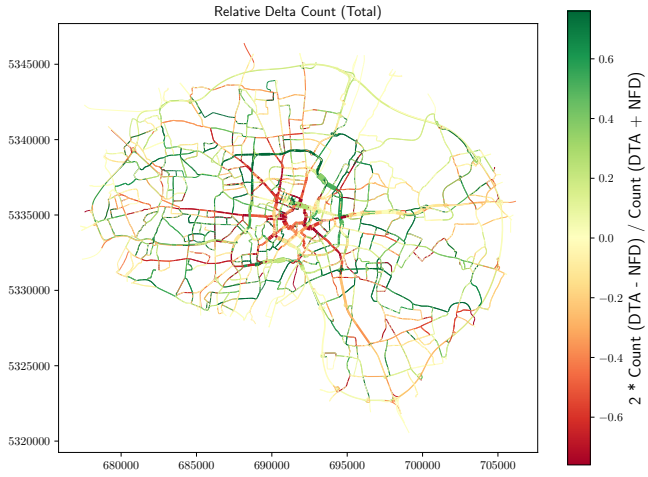

Fig. 8. Difference in street sections' utilization (excluding residential streets for clarity) for routes of all OD-pairs in the Area-A99 scenario for DTA and NFD based approach. The width relates to the sum of the counts of DTA- and NFD-based approaches.

D. Routing

In the NFD-based approach, the route choice is independent of congestion. Hence, the appearance of an edge in a route is also independent of the choice of f^i. Streets that are drawn in green colors in Fig. 8 are part of more routes in the DTA based approach, which is able to represent the spatial character of congestion. Therefore, the relative usage of these streets is higher in congested network states than in free-flow conditions. Even though the inner highway belt (B2R) is heavily congested in Munich, the relative time loss of routes through the inner city is even higher. Hence, fleet operators in simulations based on the DTA-based approach route less vehicles into the inner city than in NFD-based simulations.

It is noteworthy that this result is also replicated in the B2R-area simulation whereas the outer highway belt in the A99-area simulation only shows a smaller amount of increased use in the DTA approach. It requires further analysis to what extent the choice of the OD samples is responsible for this result.

VI. Conclusion

This study compared NFD-based and DTA-based approaches to simulate the vehicle routing problem in networks with endogenous travel times. The overall goal was to find a method to utilize the computational efficiency, which a pre-processed travel time table provides for routing computations, while including network dynamics effects on the resulting route travel times. Increased travel times due to congestion are modeled with a network travel time factor, which is first derived only from the NFD. The analysis was conducted for a case study of the city of Munich, Germany, for which we studied three different network sizes. The results show that the NFD-based approach substantially reduces computational cost: travel time queries could be performed more than 500 times as fast and the route queries returning the sequence of the fastest route are returned more than 80 times as fast. However, this severe speedup also leads to differing travel time estimates compared to the more detailed DTA-approach for all network sizes. Generally, we see larger differences for slower speeds. An analysis of OD travel times for both approaches reveals that an affine relation between the network travel time factor and the inverse network speed is possible to fit the average OD travel times quite well to the DTA counterpart. This fit to DTA results can also be pre-processed and does not limit the computational gains of the shown procedure. The standard deviations of the OD travel time differences per time interval reveal that the variance of the two models increases for lower speed, which can be explained by congestion generating additional inhomogeneity in the network.

In the end, the choice of the routing methodology depends on the use case. Modelers can decide whether they require the more realistic DTA-based approach or computation time is of higher significance.

Future work includes the investigation of network partitioning in the context of the NFD-based routing approach. A possible method to gain the computational efficiency of the shown methodology and the benefits from partitioning is the application of partition travel time factors for OD trips within one region and another travel time factor for OD trips between different regions. Additionally, the variance of aggregated results like mode share and traveled miles by mode will be investigated for both NFD- and DTA-based routing within full transportation network simulations.

References

[1] Y. Liu, P. Bansal, R. Daziano, and S. Samaranayake, "A framework to integrate mode choice in the design of mobility-on-demand systems," *Transportation Research Part C: Emerging Technologies*, no. 105, pp. 648–665, 2019.

[2] A. Tirachini and A. Gomez-Lobo, "Does ride-hailing increase or decrease vehicle kilometers traveled (vkt)? a simulation approach for santiago de chile," *International Journal of Sustainable Transportation*, vol. 14, no. 3, pp. 187–204, 2020.

[3] R. Geisberger, P. Sanders, D. Schultes, and D. Delling, "Contraction hierarchies: Faster and simpler hierarchical routing in road networks," in *Experimental Algorithms*, ser. Lecture notes in computer science, C. C. McGeoch, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, vol. 5038, pp. 319–333.

[4] H. Bast, D. Delling, A. Goldberg, M. Müller-Hannemann, T. Pajor, P. Sanders, D. Wagner, and R. F. Werneck, *Route Planning in Transportation Networks*. Cham: Springer International Publishing, 2016, pp. 19–80.

[5] D. Delling, A. V. Goldberg, T. Pajor, and R. F. Werneck, "Customizable route planning in road networks," *Transportation Science*, vol. 51, no. 2, pp. 566–591, 2017.

[6] C. L. Azevedo, K. Marczuk, S. Raveau, H. Soh, M. Adnan, K. Basak, H. Loganathan, N. Deshmunkh, D.-H. Lee, E. Frazzoli, and M. Ben-Akiva, "Microsimulation of demand and supply of autonomous mobility on demand," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2564, no. 1, pp. 21–30, 2016.

[7] F. Dandl, B. Bracher, and K. Bogenberger, "Microsimulation of an autonomous taxi-system in munich," in *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, 2017, pp. 833–838.

[8] M. Hyland and H. S. Mahmassani, "Dynamic autonomous vehicle fleet operations: Optimization-based strategies to assign avs to immediate traveler demand requests," *Transportation Research Part C: Emerging Technologies*, vol. 92, pp. 278–297, 2018.

[9] F. Dandl and K. Bogenberger, "Comparing future autonomous electric taxis with an existing free-floating carsharing system," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2037–2047, 2019.

[10] F. Dandl, M. Hyland, K. Bogenberger, and H. S. Mahmassani, "Evaluating the impact of spatio-temporal demand forecast aggregation on the operational performance of shared autonomous mobility fleets," *Transportation*, vol. 46, no. 6, pp. 1975–1996, 2019.

[11] P. Santi, G. Resta, M. Szell, S. Sobolevsky, S. H. Strogatz, and C. Ratti, "Quantifying the benefits of vehicle pooling with shareability networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 111, no. 37, pp. 13 290–13 294, 2014.

[12] J. Alonso-Mora, S. Samaranayake, A. Wallar, E. Frazzoli, and D. Rus, "On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment," *Proceedings of the National Academy of Sciences*, pp. 462–467, 2017.

[13] R. Engelhardt, F. Dandl, A. Bilali, and K. Bogenberger, "Quantifying the benefits of autonomous on-demand ride-pooling: A simulation study for munich, germany," in *IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 2992–2997.

[14] F. Dandl, K. Bogenberger, and H. S. Mahmassani, "Autonomous mobility-on-demand real-time gaming framework," in *2019 6th International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*. IEEE, 2019, pp. 1–10.

[15] S. Hörl, C. Ruch, F. Becker, E. Frazzoli, and K. W. Axhausen, "Fleet control algorithms for automated mobility: A simulation assessment for zurich," *TRB*, 2017.

[16] S. Hörl, M. Balac, and K. W. Axhausen, "Dynamic demand estimation for an amod system in paris," *IEEE Intelligent Vehicle Symposium*, 2019.

[17] G. Wilkes, L. Briem, M. Heilig, T. Hilgert, M. Kagerbauer, and P. Vortisch, "Identifying service provider and transport system related effects of different ridesourcing service schemes through simulation within the travel demand model mobitopp." in *ICoMaaS – 2nd International Conference on Mobility as a Service*, 2019.

[18] L. Lu, Y. Xu, C. Antoniou, and M. Ben-Akiva, "An enhanced SPSA algorithm for the calibration of dynamic traffic assignment models," *Transportation Research Part C: Emerging Technologies*, vol. 51, pp. 149–166, 2015.

[19] S. Peeta and A. K. Ziliaskopoulos, "Foundations of dynamic traffic assignment: The past, the present and the future," *Networks and spatial economics*, vol. 1, no. 3-4, pp. 233–265, 2001.

[20] H. S. Mahmassani, "Dynamic network traffic assignment and simulation methodology for advanced system management applications," *Networks and Spatial Economics*, vol. 1, no. 3/4, pp. 267–292, 2001.

[21] A. Paz and S. Peeta, "Behavior-consistent real-time traffic routing under information provision," *Transportation Research Part C: Emerging Technologies*, vol. 17, no. 6, pp. 642–661, 2009.

[22] F. Viti, C. M. Tampère *et al.*, "Dynamic traffic assignment: Recent advances and new theories towards real-time applications and realistic travel behaviour," *New developments in transport planning. Edward Elgar, Cheltenham*, 2010.

[23] S. Sundaram, H. N. Koutsopoulos, M. Ben-Akiva, C. Antoniou, and R. Balakrishna, "Simulation-based dynamic traffic assignment for short-term planning applications," *Simulation Modelling Practice and Theory*, vol. 19, no. 1, pp. 450–462, 2011.

[24] J. Godfrey, "The mechanism of a road network," *Traffic Engineering & Control*, vol. 8, no. 8, 1969.

[25] R. Herman and I. Prigogine, "A two-fluid approach to town traffic," *Science*, vol. 204, no. 4389, pp. 148–151, 1979.

[26] H. Mahmassani, J. C. Williams, and R. Herman, "Performance of urban traffic networks," in *Proceedings of the 10th International Symposium on Transportation and Traffic Theory*, 1987, pp. 1–20.

[27] C. F. Daganzo, "Urban gridlock: Macroscopic modeling and mitigation approaches," *Transportation Research Part B: Methodological*, vol. 41, no. 1, pp. 49–62, 2007.

[28] N. Geroliminis and C. F. Daganzo, "Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings," *Transportation Research Part B: Methodological*, vol. 42, no. 9, pp. 759–770, 2008.

[29] V. Knoop, S. Hoogendoorn, and J. Van Lint, "Routing strategies based on macroscopic fundamental diagram," *Transportation Research Record*, vol. 2315, no. 1, pp. 1–10, 2012.

[30] M. Yildirimoglu, I. I. Sirmatel, and N. Geroliminis, "Hierarchical control of heterogeneous large-scale urban road networks via path assignment and regional route guidance," *Transportation Research Part B: Methodological*, vol. 118, pp. 106 – 123, 2018.

[31] S. F. A. Batista and L. Leclercq, "Regional dynamic traffic assignment framework for macroscopic fundamental diagram multi-regions models," *Transportation Science*, vol. 53, pp. 1563–1590, 2019.

[32] R. Saedi, M. Saeedmanesh, A. Zockaie, M. Saberi, N. Geroliminis, and H. S. Mahmassani, "Estimating network travel time reliability with network partitioning," *Transportation Research Part C: Emerging Technologies*, vol. 112, pp. 46–61, 2020.

[33] L. Ambühl and M. Menendez, "Data fusion algorithm for macroscopic fundamental diagram estimation," *Transportation Research Part C: Emerging Technologies*, vol. 71, pp. 184–197, 2016.

[34] A. Loder, L. Ambühl, M. Menendez, and K. W. Axhausen, "Understanding traffic capacity of urban networks," *Scientific Reports*, vol. 9, no. 1, p. 16283, dec 2019.

[35] C. F. Daganzo and N. Geroliminis, "An analytical approximation for the macroscopic fundamental diagram of urban traffic," *Transportation Research Part B: Methodological*, no. 42, pp. 771–781, 2008.

[36] L. Leclercq and N. Geroliminis, "Estimating mfds in simple networks with route choice," *Transportation Research Part B Methodological*, vol. 57, no. EPFL-ARTICLE-188334, pp. 468–484, 2013.

[37] J. A. Laval and F. Castrillón, "Stochastic approximations for the macroscopic fundamental diagram of urban networks," *Transportation Research Part B: Methodological*, vol. 81, pp. 904–916, 2015.

[38] G. Tilg, S. Amini, and F. Busch, "Evaluation of analytical approximation methods for the macroscopic fundamental diagram," *Transportation Research Part C: Emerging Technologies*, vol. 114, pp. 1 – 19, 2020.

[39] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[40] P. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[41] R. Gutman, "Reach-based routing: A new approach to shortest path algorithms optimized for road networks," in *Proceedings 6th SIAM International Workshop on Algorithm Engineering and Experiments*, 2004.

[42] A. V. Goldberg, H. Kaplan, and R. F. Werneck, "Reach for A*: Efficient point-to-point shortest path algorithms," in *2006 Proceedings of the Eighth Workshop on Algorithm Engineering and Experiments (ALENEX)*, R. Raman and M. F. Stallmann, Eds. Philadelphia, PA: Society for Industrial and Applied Mathematics, 01212006, pp. 129–143.

[43] A. Schneck and K. Nökel, "Accelerating traffic assignment with customizable contraction hierarchies," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 73, no. 2, 2020.

[44] R. W. Floyd, "Algorithm 97: Shortest path," *Communications of the ACM*, vol. 5, no. 6, p. 345, 1962.

[45] S. Warshall, "A theorem on boolean matrices," *Journal of the ACM (JACM)*, vol. 9, no. 1, pp. 11–12, 1962.

[46] TSS, "Aimsun 7 dynamic simulator user's guide," *Transport Simulation Systems, Barcelona, Spain*, vol. 34, 7.

[47] J. Barceló and J. Casas, "Dynamic network simulation with aimsun," in *Simulation approaches in transportation analysis*. Springer, 2005, pp. 57–98.

[48] W. S. Cleveland, "Robust locally weighted regression and smoothing scatterplots," *Journal of the American statistical association*, vol. 74, no. 368, pp. 829–836, 1979.