Modeling autonomous dynamic vanpooling services in SUMO by integrating a dynamic routing scheduler

Moeid Qurashi[1], Hai Jiang[2], and Constantinos Antoniou[1]

[1] Technical University of Munich, Munich, Germany
moeid.qurashi@tum.de, c.antoniou@tum.de
[2] Tsinghua University, Beijing, China
haijiang@tsinghua.edu.cn

Abstract

Dynamic vanpooling services, as a flexible, demand adaptive transport service with ridesharing, is rather a complex transport mode. To model such a service requires a similar behavior/environment of dynamic traffic simulation environment linked with a dynamic rerouting algorithm to control the service vehicles serving stochastic requests and incurring the time-dependent network state. We develop a modeling platform for autonomous dynamic vanpooling services based on the traffic simulator SUMO, using its TraCI's API by integrating a dynamic routing scheduler, along with the required simulation setup to model such ridesharing vans. The integrated routing algorithm solves the rerouting optimization problem for both stochastic requests and time-dependent travel times, hence enabling the developed platform to model dynamic vanpooling services stochastically (i.e. stochastic demand requests arrival and service optimization during the simulation execution). A simulation executer/controller module, written in python, loading TraCI API library and the routing algorithm (C++ code wrapped as a dll) is developed controlling the simulation execution integrated with the routing scheduler by predefined communication protocols through TraCI. The platform can provide benefits to model and evaluate dynamic vanpools for understanding their feasibility, behavior, and impacts for different situations (i.e. networks/service area, network conditions, demographics, peak-offpeak hours, etc.). The platform implementation is still in the final phases and some preliminary results are depicted for different van fleet sizes and passenger requests served, running the scheduler based on dynamic travel time information linked from SUMO.

Contents

1 Introduction

Dynamic vanpooling services is an emerging mode of transport based on smartphone communication, providing flexible, demand adaptive, and economical transport through ridesharing. It comes under the larger concept of 'Mobility as a Service' (MaaS), which is based on the modern advancements in the communication industry, directing towards more sustainable transportation systems that rely on real-time smartphone communication for on-demand service. Although, being emerged recently, dynamic vanpool services have already become popular, attracting much attention in research and industry i.e. many services already running in some cities (e.g. Didi Chuxing in Beijing, Door2Door and Allygator Shuttle in Berlin, Via in New York, and Ford in London). In a dynamic vanpooling service, passengers request point-to-point pick-up and drop-off by smartphones and vans serve these passenger requests through dynamic rerouting, providing passengers a flexible transit service with ride-sharing and door-to-door travel.

A fundamental need of modeling/operating such a service is to schedule the van fleet for dynamically reroute during operation. Scheduling a dynamic vanpool service corresponds to the dial-a-ride problem (DARP) [13] with the dynamic and stochastic DARP to be the most realistic one for dynamic vanpool services since it considers both dynamic requests and future stochastic information. Within this research, we develop a modeling platform for autonomous dynamic vanpooling services based on the traffic simulator SUMO. The platform integrates a dynamic routing scheduler develop by [17] under a simulation controller module built using SUMO's TraCI API, loading the routing algorithm (C++ code) wrapped as a dynamic link library. The integrated routing algorithm solves the rerouting optimization problem for both stochastic requests and time-dependent travel times, hence enabling the developed platform to model dynamic vanpooling services stochastically i.e. stochastic demand requests arrival and service optimization during the simulation execution. The dynamic operation also requires communication between SUMO and the scheduling algorithm dynamically, set based on some predefined communication protocols.

2 Modeling ridesharing vehicles

2.1 State of the art

To model the emerging trends of dynamic mobility services, many different efforts have been made to extend the current state of the art transportation models. But, to the best of our knowledge, no explicit research is done to model dynamic vanpooling services involving stochastic request and service optimization. Similar researches have been created out in the more general area of modeling Shared Autonomous Vehicles (SAVs) (i.e. car-sharing, ride-hailing, or autonomous taxi operations with or without ride-sharing), many that include ridesharing systems (two or more passengers) e.g. [24, 4, 11, 12, 15, 16, 19, 20, 31, 5, 6]. Dynamic traffic assignment (DTA) models provide a more realistic simulation dynamics with the interaction of detailed supply and demand models replicating better traffic dynamics. In modeling similar mobility services as ridesharing, except [4] and [3], most of the literature used agent-based modeling and in particular the MATsim simulator [33] [e.g. [8, 9, 10, 14, 18, 23]] and much recently Simmobility [1]. [26] provides a detailed review of simulating demand-responsive transportation with agent-based approaches and [27] compared three different approaches to model ad-hoc demand-responsive transportation, also mentioning different requirements to model them.

MATSim a demand centric microscopic simulator, does not simulate detailed individual vehicle interactions and behavior to run larger traffic networks and higher demands faster [28], compensating on the sensitivity of traffic flow dynamics over time. MATSim extensions were developed to incorporate a dynamic routing algorithm (DVRP) for modeling DRT [25] which has limitations to account for available traffic simulation dynamics due to lack of improvement in initially given plan of DRT user agents over simulation iterations for equilibrium. Much recently, Simmobility [2] focused on modeling Autonomous Mobility On Demand (AMOD) in a similar agent-based modeling framework, but mainly focuses on door-2-door ridesharing vehicles. Simmobility's modeling framework considers a more detailed micro-simulation suite (i.e. MITSIM [7]) to simulate AMOD in a dynamic traffic simulation that detailed supply models to cater for vehicular interactions. A major lack of activity-based models for modeling dynamic ridesharing services is the pre-planned daily activity schedules (including activity sequence, modes, and departure times), fixing the agents mode choice to be a ridesharing service without considering that the complexity of such a system can make its availability very stochastic and time-dependent and required a dynamic modeling system for allowing a dynamic mode choice, multimodal behavior, and stochastic behavioral interactions of dynamic vanpools (traffic congestions, network dynamics, etc.) critical for evaluating dynamic ridesharing services.

The scheduling of dynamic vanpool service corresponds to the dial-a-ride problem (DARP) whose variants have been studied extensively in the literature [13]. The dynamic and stochastic DARP is the most realistic one for dynamic vanpool services since it considers both dynamic requests and future stochastic information. Schilde et al. propose scheduling algorithms for dynamic vanpooling considering stochastic requests [29] or stochastic time-dependent travel times [30]. Li et al. [17] proposed a scheduler for dynamic vanpool services considering both stochastic requests and time-dependent travel time. It also considers prepositioning which means that it may send the vans to locations with potential requests and finds that it can improve scheduling performance.

2.2 Modeling requirements

With an extensive review of literature and traffic simulation environments, we summarized a set of modeling requirements needed to effectively model autonomous dynamic vanpooling systems:

Routing scheduler The core part of dynamic vanpool service operation is its dynamic routing algorithm. Referring it as a 'Scheduler', these algorithms are responsible to optimize van routes to serve new demand without breaching in-service passenger preference constraints set based on passenger preferences (i.e. waiting times, in-vehicle times, and trip costs/willingness to pay) and vehicle operational attributes (e.g. charging constraints for electric vehicles)

Modeling dynamic rerouting Dynamic vanpooling requires continuous dynamic rerouting optimized by the scheduler. This would require simultaneous interaction between the traffic simulator and the scheduler during the simulation. The information interchanging includes reading the network state (i.e. link speeds or travel times), vans information (i.e. its position, occupancy, and current route), new passenger request information from the simulation, and new van routes from the scheduler. The communication frequency would be defined based on the factors i.e. frequency of requests, the time required for optimization, and the time required for reading and communicating the simulation parameters.

Autonomous vanpool vehicles Autonomous dynamic vanpools should have vehicles replicating autonomous van behavior by autonomous driving, van capacity (between 8 to 14 passengers), and supporting infrastructure to provide flexible stoppage for a door to door service.

Modeling individual passenger trips Dynamic vanpooling requires to model discrete passenger trips to assess individual trip attributes i.e. waiting times, travel times, and trip costs. Due to the flexible door to door service, the level of service experienced by each passenger variates, hence, discrete passenger modeling infrastructure is required to model individual passenger trips and evaluate the overall level of service of the mode appropriately.

3 Autonomous dynamic vanpools in SUMO

3.1 Modeling architecture

To model autonomous dynamic vanpools using SUMO, the modeling architecture shown in figure 1 is used. A SUMO simulation model is set up by adding dynamic vanpooling service vans (based on pre-defined autonomous vehicle class) in the simulation network along with the loaded network demand. In addition, the supporting infrastructure to run the door-2-door ridesharing service is also incorporated as supply enhancements. For dynamic passenger requests, a choice based preference module is set up containing the interaction of network demand (trips XML file), the perceived serving attributes, and passenger preferences profiled with different user groups (based on the demographical data). The scenario executer is a simulation controller module written in python, loading: i) the scheduler (written in C++) as dynamic link library, ii) preference module, iii) SUMO simulation model using the TraCI library.

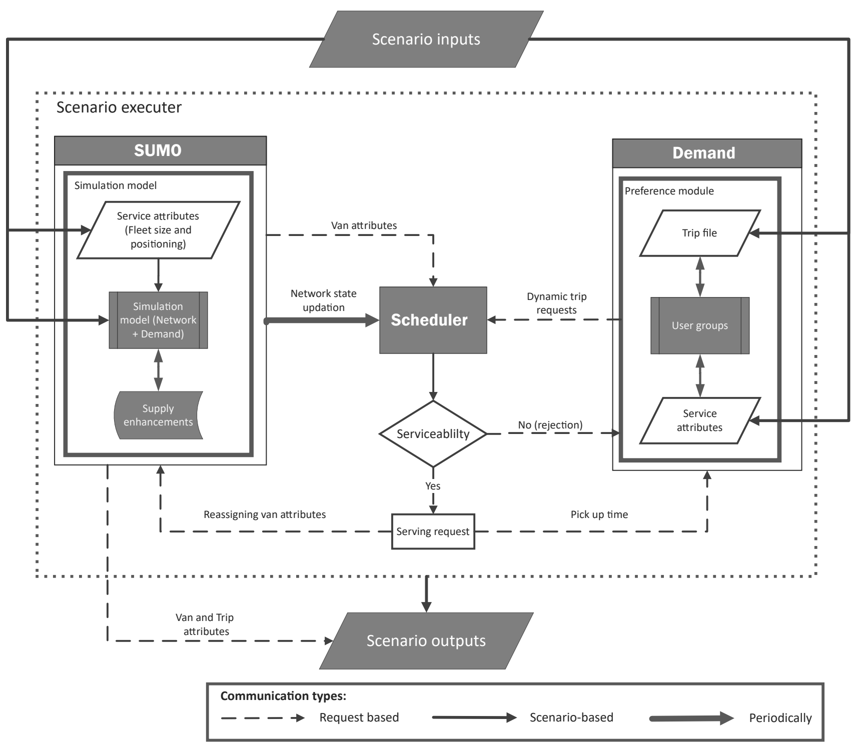

Figure 1: Modeling architectiue for SUMO

The scenario executor controls the simulation using the TraCI library functions and interacts with the scheduler and the preference module to serve dynamic requests by scheduler's optimization based on the given van service attributes. The communication protocol between these three components is on two different levels (both during simulation) i.e. request-based (when a new request emerges or a van is serving on a pick-up/drop-off point) and interval/period based to update the network state of link speeds/ travel times. The third communication level in figure 1 is scenario-based to communicate the scenario inputs/outputs to and from the scenario executor components.

3.2 Scheduling algorithm

[17] proposed a scheduler for dynamic vanpool services, modeling the scheduling problem as a DARP. The goal is to optimize the profit and the passenger level of service considering stochastic requests and time-dependent travel time. The operating profit is modeled as the operating cost minus the service revenue. The passenger level of service is modeled based on the waiting time and the detour. The objective function is defined as the linear combination of the cost, revenue, waiting, and detour times.

Each request has its pickup node, delivery node, status (i.e. new, rejected, waiting for pickup, picked, or delivered), pickup time window, and delivery time window. The pick-up time window defines the maximum waiting time while the drop off time window defines the constraints for maximum allowed detour ratio against direct trip time.

The procedure to schedule service vans uses scenario/based approaches to decide request acceptance and van routes. The scenario represents stochastic information about future requests and traffic conditions. Within a scenario, a deterministic problem gets solved using a tabu search algorithm. The inputs include 1) Set of van positions 2) Set of requests i.e. newly received and already accepted with their status. The scheduler first decides to accept each new request and then routes of each van.

3.3 SUMO - Scheduler integration

To optimize the dynamic vanpools' simultaneously along with the traffic simulation, the scenario executor module runs based on an interfacing workflow defined in figure 2. Outside the interface module, the components depict the inputs given to the scenario executor i.e. van requests, van service attributes, and simulation configuration file loaded through TraCI. To run the simulation first the simulation configuration is loaded and then the service vans are generated in the simulation. Next, the simulation time steps are run sequentially checking the occurrence of an event at each step. There are in total of three events, request event (arrival of a new request), stop event (a van stopped for service), and network update (to update network state). After completion of the simulation time, the outputs are loaded, including van trip outputs and passenger (person) trip outputs.

Out of the three events, the request and van stop event have the similar protocol, first, the van positions are read from the simulation and communicated to the scheduler (also request trip attributes are communicated in the request event), then, the scheduler runs the optimization and returns new van routes which are updated in the simulation. For network update events, network link speeds are read from the simulation and are sent to the scheduler for network state updates and the event is triggered by a pre-defined interval.

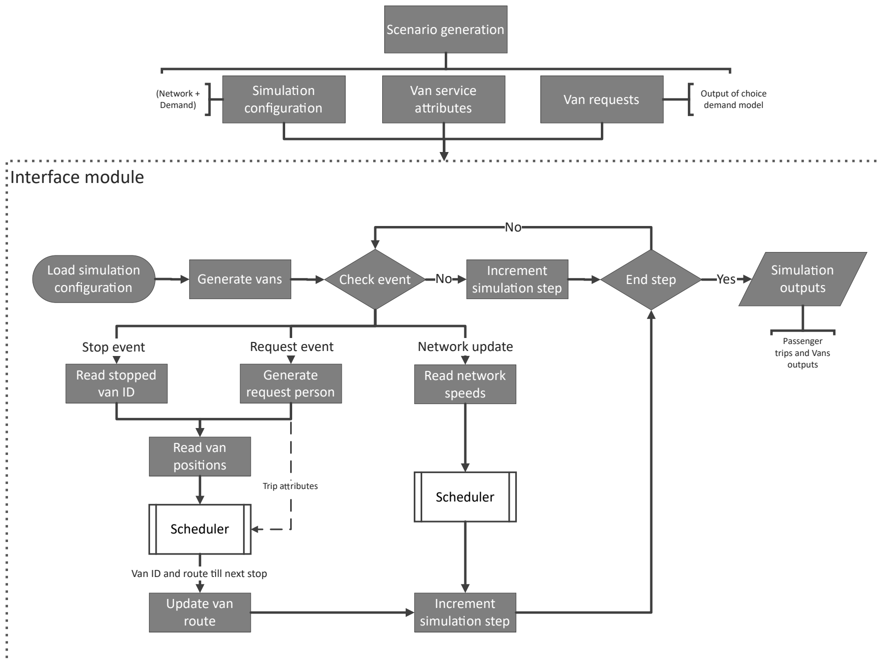

Figure 2: SUMO - Scheduler integration

3.4 SUMO enhancements

Modeling dynamic vanpooling service in a microscopic simulator like SUMO requires some additional enhancements to replicate the unique ridesharing service behavior. Following are the enhancements added in the SUMO simulation model:

Preference module Dynamic vanpooling passenger requests are generated by defining a preference module. This module incorporates passenger preference information towards service usage for travel attributes such as waiting time, travel time, and trip cost. These preferences variate among passengers by their demographics e.g. age, income, car ownership, household attributes etc and hence, user groups can be created to profile different users demographics. Dynamic trip requests are generated by evaluating mode choice for all trips available in SUMO trip xml file (based on the service attributes of all modes).

Passenger trip variables Dynamic vanpools serve each passenger on-demand and each trip's attributes variate significantly. Hence, individual trips are modelled by generating persons for each trip request and changing their trip commuting attributes based on the scheduler outputs. TraCI simulation API is used to generate persons upon request acceptance and also to assign the corresponding van for trip commute. Once the simulation is finished, we use the person trip outputs to evaluate individual trip attributes e.g. waiting time, travel time, level of service, utility experienced etc.

Autonomous vanpools vehicle class To model autonomous dynamic vanpools, the vans are generated at the start based on a predefined vehicle class. The vehicle class (`vType`) include vehicle classified as default (`passenger`), vehicle capacity, dimensions, appropriate speeds, less agile acceleration/deceleration patterns. For adequate passenger boarding/alighting, dwell times of service van are defined along defining the stop location for van to serve the passenger. To model autonomous behavior, the ACC controller based on the car following models developed by [22] and [34] is used. The vehicle parameters including maximum acceleration, deceleration, reaction time and minimum gap as per [21] and [32].

Flexible stoppage infrastructure Dynamic vanpools provide door-2-door or point-2-point service which requires on-road stoppage for passenger pick-up and drop-off. In author's knowledge, SUMO requires parking infrastructure on the network to do parking based stops for serving the passenger. In current implementations, we do on-road service for pick-up and drop-off (considering shorter dwell times for a single passenger), and also plan to add parking infrastructure in future for van stops as parking stops. But, on-road stops are not adequate for idle empty van waiting for service call-ups, to address this issue we used another SUMO's TraCI based functionality of `moveToXY` which allows moving empty vans outside the network. The vehicles at simulation start or when they are empty during the service, are moved to the an arbitrary location outside the network. Upon call for serving, the vans are moved back to their last network position.

4 Preliminary results and discussion

Given the modeling architecture and its implementation method in SUMO (section 3). The implementation to create the dynamic vanpooling modeling platform is still in its final phases. Under this section, we show some preliminary results extracted from running the scheduler algorithm linked with dynamic travel time information from SUMO simulation model (i.e. the van travel behavior is approximated having dynamic travel times information of links from SUMO). The case study for the experiment is based on Munich city centre network (figure 3).

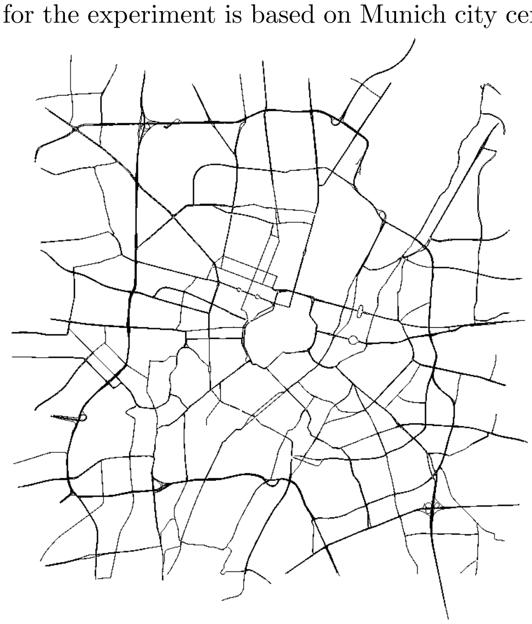

Figure 3: Network of Munich city center

Multiple scenarios have been simulated for all penetration rates to account for the involved stochasticity of request arrival time and location, and van positioning. Figure 4 shows the averaged results for dynamic vanpools for different penetration rates i.e. different van fleet sizes and different number of requests. Each data point depicts a combination of fleet size and requests arrived. The results show the effects of different fleet sizes and number of requests served in terms of passenger experience i.e. additional detour times (shown by hue) and the extent of reduction in Total Vehicles Kilometer Travelled-TVKT (by data point size) due to ridesharing. The results show that given the munich network scenario, the increase in service fleet size after 15 vans doesn't improve the passenger level of service or TVKT reduction significantly, implying that the extent of ridesharing won't increase significantly after this fleet size for serving 100 or less requests.

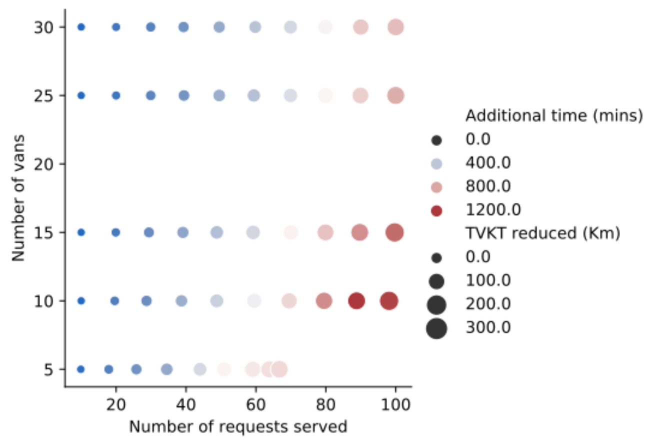

Figure 4: Scenarios for different penetration rates of dynamic vanpools

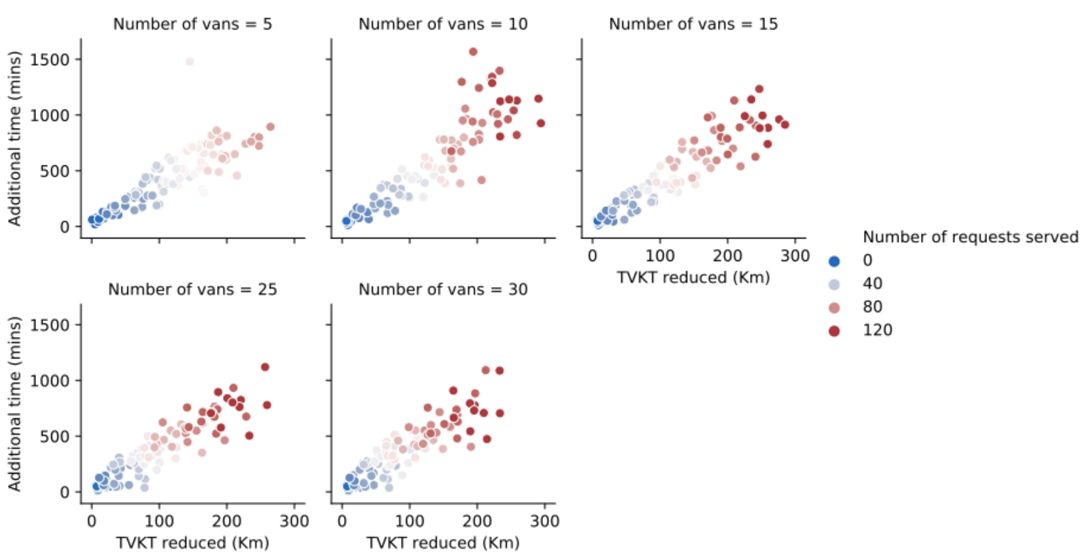

Figure 5: Additional trip time (delays) vs TVKT reduction

Figure 5 shows the relationship plots between additional detour time versus the TVKT reduced for serving different number of requests by a single fleet size. The plots depict that: 1) after an optimal service fleet of 15 vans TVKT benefits are reduced without improving the level of service for passengers significantly, 2) similar TVKT targets are achieved for lesser fleet size but may incur longer additional times (10 versus 15 vans fleet).

5 Conclusion

Dynamic vanpooling services is rather a complex transport mode due to its nature of ridesharing to serve stochastic requests economically as per the passenger constraints. Also, to model such a service requires the similar behavior/environment of dynamic traffic simulation environment linked with dynamic rerouting algorithm to control the service vehicles serving stochastic requests and incurring the time dependent network state. This research presents the modeling architecture and implementation method to model dynamic vanpooling service in the microscopic traffic simulation SUMO. Using SUMO to model such dynamic services can allow to evaluate their behavior, availability, and impacts in a dynamic traffic simulation environment with stochastic time dependent network state. The proposed platform addresses the modeling requirements require to model dynamic vanpools i.e. modeling adqeuate vehicle class with supporting infrastructure, a preference model to do mode choice for generating request based on service attributes, integration of dynamic rerouting scheduler that considers stochastic request arrival and dynamic travel times, doing the optimization of van routes to serve requests along the simulation run. The platform have benefits to model and evaluate dynamic vanpools for understanding their feasibility, behavior, and impacts for different situations (i.e. networks/service area, network conditions, demographics, peak-offpeak hours etc.). Preliminary results are shown for different penetration rates for dynamic vanpools to show the relationship between fleet size, passenger requests, detour times and service benefits of TVKT.

Acknowledgement This research is funded by the German Research Foundation - Deutsche Forschungsgemeinschaft (DFG) [Project number 392047120, Research grants in collaboration with China].

References

[1] Muhammad Adnan, Francisco C Pereira, Carlos Miguel Lima Azevedo, Kakali Basak, Milan Lovric, Sebastián Raveau, Yi Zhu, Joseph Ferreira, Christopher Zegras, and Moshe Ben-Akiva. Simmobility: A multi-scale integrated agent-based simulation platform. In *95th Annual Meeting of the Transportation Research Board Forthcoming in Transportation Research Record*, 2016.

[2] Muhammad Adnan, Francisco C Pereira, Carlos Miguel Lima Azevedo, Kakali Basak, Milan Lovric, Sebastián Raveau, Yi Zhu, Joseph Ferreira, Christopher Zegras, and Moshe Ben-Akiva. Simmobility: A multi-scale integrated agent-based simulation platform. In *95th Annual Meeting of the Transportation Research Board Forthcoming in Transportation Research Record*, 2016.

[3] Md Jahedul Alam and Muhammad Ahsanul Habib. Investigation of the impacts of shared autonomous vehicle operation in halifax, canada using a dynamic traffic microsimulation model. *Procedia Computer Science*, 130:496–503, 2018.

[4] Sabina Alazzawi, Mathias Hummel, Pascal Kordt, Thorsten Sickenberger, Christian Wieseotte, and Oliver Wohak. Simulating the impact of shared, autonomous vehicles on urban mobility – a case study of milan. In *SUMO 2018 - Simulating Autonomous and Intermodal Transport Systems*, volume 2, pages 94–110. 2018.

[5] Carlos Lima Azevedo, Katarzyna Marczuk, Sebastián Raveau, Harold Soh, Muhammad Adnan, Kakali Basak, Harish Loganathan, Neeraj Deshmunkh, Der-Horng Lee, Emilio Frazzoli, et al. Microsimulation of demand and supply of autonomous mobility on demand. *Transportation Research Record*, 2564(1):21–30, 2016.

[6] Rounaq Basu, Andrea Araldo, Arun Prakash Akkinepally, Bat Hen Nahmias Biran, Kalaki Basak, Ravi Seshadri, Neeraj Deshmukh, Nishant Kumar, Carlos Lima Azevedo, and Moshe Ben-Akiva. Automated mobility-on-demand vs. mass transit: a multi-modal activity-driven agent-based simulation approach. *Transportation Research Record*, 2672(8):608–618, 2018.

[7] Moshe Ben-Akiva, Haris N Koutsopoulos, Tomer Toledo, Qi Yang, Charisma F Choudhury, Constantinos Antoniou, and Ramachandran Balakrishna. Traffic simulation with mitsimlab. In *Fundamentals of traffic simulation*, pages 233–268. Springer, 2010.

[8] Joschka Bischoff and Michal Maciejewski. Autonomous taxicabs in berlin – a spatiotemporal analysis of service performance. *Transportation Research Procedia*, 19:176–186, 2016.

[9] Patrick M. Boesch, Francesco Ciari, and Kay W. Axhausen. Autonomous vehicle fleet sizes required to serve different levels of demand. *Transportation Research Record: Journal of the Transportation Research Board*, 2542(1):111–119, 2016.

[10] Daniel J. Fagnant and Kara M. Kockelman. Dynamic ride-sharing and fleet sizing for a system of shared autonomous vehicles in austin, texas. *Transportation*, 45(1):143–158, 2018.

[11] Krishna Murthy Gurumurthy and Kara M. Kockelman. Analyzing the dynamic ride-sharing potential for shared autonomous vehicle fleets using cellphone data from orlando, florida. *Computers, Environment and Urban Systems*, 71:177–185, 2018.

[12] Michael Heilig, Tim Hilgert, Nicolai Mallig, Martin Kagerbauer, and Peter Vortisch. Potentials of autonomous vehicles in a changing private transportation system – a case study in the stuttgart region. *Transportation Research Procedia*, 26:13–21, 2017.

[13] Sin Ho, W Szeto, Yong-Hong Kuo, Janny Leung, Matthew Petering, and Terence W.H. Tou. A survey of dial-a-ride problems: Literature review and recent developments. *Transportation Research Part B: Methodological*, 111, 02 2018.

[14] Sebastian Hörl. Agent-based simulation of autonomous taxi services with dynamic demand responses. *Procedia Computer Science*, 109:899–904, 2017.

[15] Benedikt Jäger, Fares Maximilian Mrad Agua, and Markus Lienkamp. Agent-based simulation of a shared, autonomous and electric on-demand mobility solution. In IEEE Intelligent Transportation Systems Conference, editor, *IEEE ITSC 2017*, pages 250–255. IEEE, 2017.

[16] Benedikt Jäger, Carsten Brickwedde, and Markus Lienkamp. Multi-agent simulation of a demand-responsive transit system operated by autonomous vehicles. *Transportation Research Record: Journal of the Transportation Research Board*, 77(4):036119811878664, 2018.

[17] Donghui Li, Constantinos Antoniou, Hai Jiang, Qianyan Xie, Wei Shen, and Weijian Han. The value of prepositioning in smartphone-based vanpool services under stochastic requests and time-dependent travel times. *Transportation Research Record*, 2673(2):26–37, 2019.

[18] Mustafa Lokhandwala and Hua Cai. Dynamic ride sharing using traditional taxis and shared autonomous taxis: A case study of nyc. *Transportation Research Part C: Emerging Technologies*, 97:45–60, 2018.

[19] Luis M. Martinez and José Manuel Viegas. Assessing the impacts of deploying a shared self-driving urban mobility system: An agent-based model applied to the city of lisbon, portugal. *International Journal of Transportation Science and Technology*, 6(1):13–27, 2017.

[20] Neda Masoud and R. Jayakrishnan. Autonomous or driver-less vehicles: Implementation strategies and operational concerns. *Transportation Research Part E: Logistics and Transportation Review*, 108:179–194, 2017.

[21] K. Mattas, M. Makridis, P. Hallac, M. A. Raposo, C. Thiel, T. Toledo, and B. Ciuffo. Simulating deployment of connectivity and automation on the antwerp ring road. *IET Intelligent Transport Systems*, 12(9):1036–1044, 2018.

[22] Vicente Milanés and Steven E Shladover. Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data. *Transportation Research Part C: Emerging Technologies*, 48:285–300, 2014.

[23] Ana T. Moreno, Andrzej Michalski, Carlos Llorca, and Rolf Moeckel. Shared autonomous vehicles effect on vehicle-km traveled and average trip duration. *Journal of Advanced Transportation*, 2018(1):1–10, 2018.

[24] Bat-hen Nahmias-Biran, Jimi B Oke, Nishant Kumar, Carlos Lima Azevedo, and Moshe Ben-Akiva. Evaluating the impacts of shared automated mobility on-demand services: an activity-based accessibility approach. *Transportation*, pages 1–26, 2020.

[25] Nicole Ronald, Russell Thompson, John Haasz, and Stephan Winter. Determining the viability of a demand-responsive transport system under varying demand scenarios. In *Proceedings of the Sixth ACM SIGSPATIAL International Workshop on Computational Transportation Science*, pages 7–12, 2013.

[26] Nicole Ronald, Russell Thompson, and Stephan Winter. Simulating demand-responsive transportation: A review of agent-based approaches. *Transport Reviews*, 35(4):404–421, 2015.

[27] Nicole Ronald, Russell Thompson, and Stephan Winter. Simulating ad-hoc demand-responsive transportation: a comparison of three approaches. *Transportation Planning and Technology*, 40(3):340–358, 2017.

[28] Mustapha Saidallah, Abdeslam El Fergougui, and Abdelbaki El Belrhiti El Alaoui. A comparative study of urban road traffic simulators. *MATEC Web of Conferences*, 81:05002, 01 2016.

[29] Michael Schilde, Karl F Doerner, and Richard F Hartl. Metaheuristics for the dynamic stochastic dial-a-ride problem with expected return transports. *Computers & operations research*, 38(12):1719–1730, 2011.

[30] Michael Schilde, Karl F Doerner, and Richard F Hartl. Integrating stochastic time-dependent travel speed in solution methods for the dynamic dial-a-ride problem. *European journal of operational research*, 238(1):18–30, 2014.

[31] Ahmed Sherif, Ahmad Alsharif, Jacob Moran, and Mohamed Mahmoud. Privacy-preserving ride sharing organization scheme for autonomous vehicles in large cities. In *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, pages 1–5. IEEE, 2017.

[32] Steven E Shladover, Dongyan Su, and Xiao-Yun Lu. Impacts of cooperative adaptive cruise control on freeway traffic flow. volume 2324, pages 63–70. SAGE Publications Sage CA: Los Angeles, CA, 2012.

[33] Kay W Axhausen, Andreas Horni, and Kai Nagel. *The multi-agent transport simulation MATSim*. Ubiquity Press, 2016.

[34] Lin Xiao, Meng Wang, and Bart van Arem. Realistic car-following models for microscopic simulation of adaptive and cooperative adaptive cruise control vehicles. *Transportation Research Record*, 2623(1):1–9, 2017.