The Thesis Committee for Haonan Yan
Certifies that this is the approved version of the following thesis:

Shared Autonomous Vehicle (SAV) Fleet Operations Across the Minneapolis-Saint Paul Region, with Emphasis on Empty Travel, Response Times, and No-Idling Laws over Space and Time of Day

APPROVED BY
SUPERVISING COMMITTEE:

Supervisor: ______________________________

Kara M. Kockelman

____

Daniel J. Fagnant

Shared Autonomous Vehicle (SAV) Fleet Operations Across the Minneapolis-Saint Paul Region, with Emphasis on Empty Travel, Response Times, and No-Idling Laws over Space and Time of Day

by

Haonan Yan

Thesis

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Engineering

The University of Texas at Austin

December 2019

Acknowledgements

I would like to thank my advisor, Dr. Kara Kockelman, for her support of my thesis and for her teaching me everything I know about autonomous vehicles. I would also like to thank ETH Zurich for providing the Autonomous Mobility on Demand (AMoD) software with SAV simulation codes. I want to thank K. Murthy Gurumurthy for his technical support and guidance in using MATSim. He is third author on a paper that Dr. Kockelman and I seek to publish, based on this work. Much of the following thesis benefits from very extensive editing by Dr. Kockelman and Murthy.

I also am grateful to Dr. Dan Fagnant for reviewing my thesis and for his useful suggestions, and to Jooyong Lee for help with emissions rates and cost calculations. Finally, thanks go to Minnesota's Metropolitan Council for providing the trip data needed to simulate the Minneapolis-St Paul region's travelers and the National Science Foundation (NSF) for its financial support of this research under the Sustainable Healthy Cities Research Network.

Finally, thank you to my family and my friends for their support and concern all these years.

.

Abstract

Shared Autonomous Vehicle (SAV) Fleet Operations Across the Minneapolis-Saint Paul Region, with Emphasis on Empty Travel, Response Times, and No-Idling Laws over Space and Time of Day

Haonan Yan, M.S.E.

The University of Texas at Austin, 2019

Supervisor: Kara M. Kockelman.

Many well-known enterprises are road-testing fully-automated vehicles (AVs), including General Motors, Waymo, Uber, Tesla, and Apple. Most AVs are expected to be used in shared AV (SAV) fleets initially, for daily trip-by-trip use, as an autonomous ride-hailing service. SAVs will allow savings on vehicle ownership and maintenance costs, parking search time, and parking access times.

This study micro-simulates passenger travel throughout the Minneapolis–Saint Paul (MSP) region of Minnesota, when relying on a system of SAVs. The extended region includes 9.5 million person-trips per weekday, 7 counties, 2485 traffic analysis zones (TAZs), and about 42,000 roadway links (obtained using OpenStreetMap). An agent-based toolkit, MATSim, allows tracking of individual travelers throughout the day and across their activity locations. The region's metropolitan planning organization, Metropolitan Council, provided all travelers' itineraries, trip purposes, origins, and destinations, along with land use data (jobs and population counts) by TAZ.

To simulate SAV assignments to each traveler requesting a trip, along with traveler wait times and arrival times at their destinations, the code form Hörl (2017) - who extended Bischoff and Maciejewski's (2016) MATSim codes - and MATSim's autonomous mobility-on-demand (AMoD) simulator were used here. The SAV fleet size and starting locations were specified before a typical weekday's simulation for 2015. All travel demands sampled from the MSP population must be met by the SAV fleet if they can be met within a pre-specified max-wait-time duration of 1 hour. Travelers are assumed to cancel their SAV request after waiting more than 1 hour. Finally, special SAV parking lots or waiting areas were created to avoid SAVs idling on busy streets in the downtown and other popular locations, between serving trips, to see how such curb-use policies affect wait times and other fleet performance metrics.

Using supercomputers, this work simulated 180,000 person-trips and 450,000 person-trips (2% and 5% of the region's 9.2 million daily person-trips) and 480,000 person-trips for the Twin Cities over a 24-hour weekday. Results suggest that the average SAV in this region can serve at most 30 person-trips per day with less than 5 minutes of average wait time for travelers, thus replacing about 10 household vehicles (assuming no one needs to leave the region) but generating another 13 % vehicle-miles traveled (VMT) each day, thereby adding some congestion to the network. By enabling and encouraging active use of for dynamic ride-sharing (DRS), where strangers share rides together, the SAV fleetwide VMT fell, on average, by 17% - and empty VMT (eVMT) fell by 26%, as compared to scenarios without DRS. Interestingly, the 81% and 84% of TAZs with less than 6 minutes average wait times (in the AM and PM peak periods, respectively) are uniformly distributed over this large, 7-county region, suggesting that MSP residents will enjoy similar SAV service levels everywhere (though response times do rise during peak times of day).

For the Twin Cities region, most eVMT emerges in the northern and southern sub-regions, rather than in the cities' CBDs. eVMT and wait times are relatively high during the AM and PM peak periods (6 am to 9 am and 3 pm to 6 pm) but fall significantly during the PM peak period if DRS is offered and actively used by travelers. When compared to idling-at-curb scenarios, the no-idling-on-busy-downtown-road-segment scenarios (using central SAV parking lots) generated 8% more VMT, while eVMT rose by 9 percentage points on average, across all 4 companion scenarios. This study also estimated various energy and emissions savings of SAVs versus the U.S. status quo. Compared to the average household passenger car (a 4-door sedan), which uses 31 miles per gallon), a fleet of 52 mi/gallon hybrid electric SAVs are estimated here to lower the energy demands by 21% and emission-related health costs by roughly 30%. sulfur dioxide (SO_2) by 20%, carbon monoxide (CO) by 46%, oxides of nitrogen(NOx) by 30%, volatile organic compounds (VOC) by 48%, particulate matter that is 10 micrometers or less in effective diameter (PM10) by 20%, carbon dioxide (CO2) by 20% and methane (CH4) by 35%. Such fleet shifts would save roughly 30% in emissions-related health costs and 64% in energy use.

Keywords: Self-driving Vehicles; Shared Autonomous Vehicles; Dynamic Ride-Sharing; Travel Demand Modeling; Empty Vehicle-Miles Traveled; Curb Idling Policies

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

Autonomous vehicle (AV) technology has rapidly developed over the last decade. With AVs expected to be used in shared fleets, as shared AVs (SAVs), many researchers are working to optimize SAV strategies in the realms of operations and pricing, while minimizing negative urban and regional impacts. SAVs will save travelers parking search and access times, while freeing households from the burdens of vehicle maintenance and storage. SAVs are expected to substitute for current private vehicle use in many settings, since they are able to self-drive to those requesting rides, even find a parking space and refueling or recharging spots. SAVs are also expected to be an important transportation mode option before individuals can afford and/or are permitted to purchase their own AVs (Liu, 2016).

Many AV impacts are anticipated. For example, AVs can readily follow optimal routes to reach their destinations with self-adjustments in real-time (Claudel and Ratti, 2015). AVs may offer opportunities for dynamic allocation of lanes (if there is no median dividing opposing lanes) during peak periods and before entering bottlenecks by connecting to traffic management systems in real-time (Skinner and Bidwell, 2016). Such traffic management systems can reduce network congestion and the associated emissions and energy use (Ticoll, 2015). Human error while driving is the dominant cause of traffic crashes, including alcohol and drug use, use of mobile devices, fatigue and lack of driving knowledge or experience (Eugensson et al., 2013). By avoiding such mistakes, AVs are expected to considerably improve motorized-travel safety (Hars, 2010; Rodoulis, 2014; Li and Kockelman, 2016). SAVs are also expected to reduce travel costs (Chen and Kockelman, 2015; Liu et al., 2016; Fagnant and Kockelman, 2018; Simoni et al., 2019;

Gurumurthy et al., 2019) and impact long-distance travel (Perrine et al., 2018; LaMondia et al., 2016)

This study microsimulates personal trip-making throughout the Minneapolis-St Paul (MSP) region of Minnesota State, USA using a system of SAVs. The input files rely on network data from OpenStreetMap and 24-hour trip-making data from the region's metropolitan planning organization, called Metropolitan Council. The agent-based MATSim toolkit allows one to track individual travelers or "agents" throughout the day, between all addresses or activity sites. Metropolitan Council provided all travelers' itineraries, trip purposes, origins, and destinations, along with land use data by traffic analysis zone (TAZ). The SAV fleet size and starting locations are determined in a 24-hour initial simulation, so that a new SAV is generated whenever a traveler's wait time exceeds a desired window of 1 hour. In the subsequent 24-hr simulations, some travelers may wait longer, and the SAV fleet's response radius will expand until an SAV can be assigned. In other words, all travel demands will be met unless travelers cancel their SAV requests after waiting 1 hour. Finally, all SAVs are assumed to be able to remain at the curb where they dropped off their passenger(s) in most scenarios, but several restricted-curb-parking scenarios allow one to appreciate the reality of congested curb settings and likely public policy responses to SAVs idling anywhere.

LITERATURE REVIEW

In recent years, many researchers have attempted to understand and optimize AV and SAV operations. For example, Kornhauser et al. (2013) simulated autonomous taxi/SAV services with coordinated access to rail and bus transit networks, with SAV stations placed every half mile across the US state of New Jersey. Individuals could only use these SAVs after arriving at those stations, rather than waiting for SAVs to pick them up at their true origins. They also have to get off at SAV stations and then walk to their final destinations. Spieser et al. (2014) simulated an SAV transportation system for Singapore's travelers, and estimated that it could reduce current passenger vehicle ownership by two-thirds.

Fagnant et al. (2014) programmed an agent-based model for SAVs across a small town (10 miles x 10 miles) with a perfectly gridded network of links (every ¼ mile). Destination choices from Poisson-generated trip origins (mimicking average population densities around Austin, Texas' central zones) reflected probability of trip direction from the origin, based on location and time of day (with more trips heading to the town's core in the morning than in the evenings, for example). The authors presented four strategies for proactive SAV relocations while waiting between trip assignments, in order to anticipate evolving trip demands and thereby shorten wait times for travelers. The first strategy (labeled R1) was to pursue a better balance between expected (next-period) SAV demand and "supply" (available SAVs) in each of 25 2-mile by 2-mile blocks. The second strategy (R2) was very similar, but used 100 one-mile by one-mile blocks. The third strategy (R3) sought to fill any zero-SAV ¼-mile by ¼-mile cells by relocating SAVs from cells with at least two available (idling/not en route) SAVs. The fourth strategy (R4) worked to transfer available SAVs from adequately supplied cells to those with potential SAV shortages. In Fagnant and Kockelman's "small town" simulation, average wait times were less than 20 seconds (ignoring the 5-minute time step used for SAV assignments to travelers, to keep the code simpler) or under 2.8 minutes to reflect the fact that real travelers will call at any time in the time step for a vehicle. Less than 0.5% travelers waited more than 5 minutes and just 3 people per day (0.005% of travelers), on average, waited 10 minutes or more, suggesting a very a high level of service. During peak times of day, more than 97% vehicles were occupied, delivering high SAV utilization rates. For their small-city simulation, where no travelers could leave the town, each SAV was estimated to replace about 11 household vehicles, assuming standard US trip-making rates per vehicle per day, with 10% more travel distance (due to empty-SAV travel between drop-offs and pickups of travelers).

AVs can reduce travel costs in the future after the costs of AV production and operation fall sufficiently (Chen and Kockelman, 2016; Liu et al., 2016; Gurumurthy et al., 2019). It is also expected to play a main role in long-distance trips (LaMondia et al., 2016) and in trips made by the disabled and young people. With added travel comes added vehicle-miles traveled (VMT). Simoni et al. (2019) simulated AVs and SAVs across the City of Austin and estimated daily passenger-VMT increases of 16.2% for an AV-oriented scenario (where personal AVs are widely used) and 22.4% for an SAV-oriented scenario (where shared mobility is more prevalent). Fagnant et al. (2014) used an agent-based model with a gridded representation of Austin streets and 25 2-mile by 2-mile neighborhoods to evaluate different SAV relocation strategies. Average waiting times in their 10 mi x 10 mi town fell, and less than 0.5% of travelers waited for more than 5 minutes. During peak periods, more than 97% of their SAVs were occupied, delivering high SAV utilization levels. They estimated that each SAV could replace around 11 conventional vehicles if no travel outside the region was required but added up to 10% more VMT. Gurumurthy et al. (2019) simulated empty VMT (eVMT)by SAVs across the wider Austin region to vary from 3.8% to 18.9 % of the total passenger-VMT by SAV. If SAVs are not permitted to sit at their most recent destination, before responding to a new trip call, such relocation will add new VMT. Maciejewski et al. (2016) concluded that small autonomous taxi fleets in a limited area cannot generate extra congestion. They also argued that sharing rides could probably address any SAV fleet congestion-contribution issues.

Huang et al. (2019) studied the traffic effect of self-driving vehicles in the Texas. Compared with the original statewide analysis model (SAM), the model used in the research included other 3 transportation modes: AVs, SAVs and Atrucks which were added to the model using the four-step method. According to the data from the SAM, a 15% trip generation was added to the AV scenario and trip distribution was divided into 2 parts: destination choice model for general passenger trips and doubly constrained gravity model for freight trips. In this study, the AVs and hybrid vehicles (HVs) had operating costs of $0.6 per mile, and SAV prices of $1.5, $1.0, and $0.5 per mile were used for sensitivity analysis. The result of the traffic assignment was obtained after 10 times iteration. The result showed trips by automobile increased by 21.0% and long-distance trips by automobile increased by 88.5%. For freight trips, the trips of trucks with different commodities increased by varying degrees. Among them, the freight trips with coal saw an enormous increase (51.3%). The average VMT of automobiles increased by 39.5% but that of several main cities had more significant increases (e.g. 46.5% in Austin). The authors concluded that people would be more willing to shift to further places, as shown by the increased average travel length from 14 miles to 16 miles. Meanwhile, the trips by air will significantly decrease by more than 90%.

Loeb et al. (2018) put forward a way to extend the benefits of shared autonomous vehicles through an electric vehicle (EV) fleet. EVs are more economic and environmental-friendly than conventionally fueled vehicles. However, they have limited range and charge times. This research discussed a method to determine the placement of charging stations in 6 counties in Austin, Texas for the optimal use of shared autonomous electric vehicles (SAEVs) through 3 steps. The first was a tour generation, where the travel data of the Capital Area Metropolitan Planning Organization (CAMPO) trip-making predictions and National Household Travel Survey (NHTS) data were used to generate activity plans by Liu et al. (2016). The second was a Dynamic traffic assignment which aimed at a network-wide quasi-user equilibrium by a muilt-agent transport simulation model (MATSim)

(Horni et al., 2016). The third was an SAV simulation, in which the authors added a new SAEV mode to the existing simulation code (Bösch et al., 2016). The study concluded that the percentage of eVMT in this simulation was higher than what had been found in other papers, and that it could be further decreased by offering more charging locations. Moreover, a longer charging time and more travelers sharing an EV could also increase the average response time. Loeb et al. (2017) discussed the estimates of the costs of this SAEV fleet. The study focused on the costs of electric vehicles, including purchasing and maintenance costs, electricity, charger construction and maintenance, insurance, registration and general administrative costs. The simulation of SAEVs was similar to the previous one. However, the authors added a mode choice and a dynamic ridesharing model as additional modifications. A logit model was used to calculate the probability of rejecting the service. The authors used a first-in-last-out (FILO) pattern to solve the problem of the limitation of dynamic ridesharing capabilities. The study concluded that a fast-charging, long-range fleet was the best EV option and a short-range and slow-charging vehicle was the worst option, with the former being the most profitable. According to the financial analysis, a fully electrified fleet was not prepared to operate at the moment, but EVs offer a viable alternative owing to the uncertain future of our climate and fossil fuel prices.

Perrine et al. (2018) considered the long-distance factor about AVs for a long-distance passenger travel demand model from the Federal Highway Administration (FHWA) and added a new AV mode to make the travel demand model and destination choice models more complete. The existing model was expanded by emphasizing on the effect of AVs mid-way and long distance, for example, the 240-mile (385 km) route between Houston and Dallas. Long distance travel was modeled in almost all pairwise combinations of 4,486 National Use Microdata Area (NUMA) zones. The authors, using pre-existing parameters, defined the trip distributions for each mode (car, bus, rail and air) by a nested logit model. Following this, several parameters ($0.20 per mile operational costs and $6.00 value of time) of AVs were assumed to build a new modal alternative for AV. The size for NUMA and the purposes were quantified as coefficients added to the mode choice model, which gave the possibilities of choosing each mode. The authors showed the resulting number of trips after the AV mode was added. Following this, they analyzed the impacts of trip mode choices for all trip purposes after the introduction of AVs and the market penetration of AVs. The authors concluded that, as shown by the parameters set, the value of air travel trip generation for shorter and further long-distance trips was cut to 53%. Air travel trips were largely replaced by increased AV trips. Car and AV trips for shorter distances and for longer distances increased by 5% and 12%, respectively.

Dynamic ride-sharing (DRS) is considered as an effective mode alternative for users to access available automobiles with lower costs. Bhat (2016) made a comparison of the current taxi implementation and dynamic ride-sharing scheme through the New York City Area. It confirmed that DRS has a significantly higher average vehicle occupancy than non-ride-sharing schemes. Jung et al. (2013) developed a shared-taxi algorithm by using hybrid-simulated annealing to dynamically assign passenger requests efficiently. The simulation results revealed that the algorithm can maximize the system efficiency of dynamic ride-sharing. Fagnant et al. (2018) improved the algorithm from Jung et al. (2013) to strengthen the efficiency of anticipatory SAV relocation and simulated SAVs fleets in Austin. The results showed DRS decreased the total service time (from 15.0 to 14.7 minutes) and travel costs, depending on different scenarios for SAV users. Furthermore, the VMT decreased by over 8% with DRS, which means the congestions of the network was improved. With SAV services at $1.00 per mile of a non-shared trip, SAVs operation companies can earn a 19% annual (long-term) return on investment with $70,000 per SAV per day. Hörl (2017) provided agent-based models for DRS in MATsim, while the models also generated datasets of people and detailed information about the trips for dynamic traffic simulation. Gurumurthy and Kockelman (2018) simulated SAVs with DRS in Orlando using MATLAB. This simulation used datasets from AirSage's cellphone-based trip tables for over 30 days. Approximately 60% of single trips were willing to be shared with other individual trips with less than 5 minutes added travel times from sharing. This value would increase to 80% if the waiting time or travel time increased to 15 and 30 minutes. With the 1 SAV per 22 person-trips, SAVs could satisfy almost half of total demand in that region for improving congested traffic condition.

Zachariah et al. (2014) simulated a fleet of autonomous taxis across the state of New Jersey, with many SAV stations, named as aTaxiStand, covering the entire state to store SAVs. The whole state was divided into 0.5 mi *0.5 mi square pixels. Unlike many researches focusing on the customers' waiting time, this work used departure delays (DD) to represent the tolerance of SAVs for additional passengers who were waiting, when initial agents were already in autonomous taxis. Another parameter was called the number of common destinations (CD). It could reflect the number of destinations for ridesharing, with CD=0 and CD=1 representing no permitted ridesharing trips and permitted ridesharing trips to a single destination, respectively. In addition, any additional trip that may increase the distance of the direct trip by more than 20% was forbidden. For the rideshare methodology, it only allowed rideshare matching in the station where the first agent was at. That is, the rideshare trip can only be made in one station during the DD. Furthermore, this work considered the SAV oversupply situation, which meant that every station had adequate SAVs for services. Different scenarios of all combinations of CD = 0, 1, 2, 3, 4, 5 and DD = 0, 1, 2, 3, 4, 5. were simulated across New Jersey. The results showed that with increased CD and DD, the True Average Vehicle Occupancy (AVO) increased. With (DD, CD) =(5,5), the AVO increased to the maximum value 2.93, which indicated that the total miles traveled by the SAVs accounted for one thirds if all trips were served individually. Regions with high population density were estimated to have a significant impact on AVO and these particular regions would benefit more from SAVs.

Tachet et al. (2017) modeled shareability to reflect the fraction of individual trips that can be shared in New York City, San Francisco, Singapore, and Vienna. A shareability shadow was generated along with one arbitrary trip, by representing a cylinder consisted of circular areas with radius R at a specific time T. The trips with their origins and destinations in the shareability shadow were both considered as eligible sharing trips. An explanatory variable L consisted of an area of the city, the average traffic speed in the city, sharing delay and the average number of trips per hour. The used dataset included networks, information of taxi (positions, fleet size, average speed, etc.) and generated trips with the Poisson distribution and Uniform distribution in spatial terms. The results showed that if R was large enough, shareability would not be influenced by R, and the city's influence on the shareability only appeared through the quantity L. That means, with rigorous mathematical derivation, the authors created a single, universal shareability curve that could predict the potential for ridesharing in any city. The authors concluded that there was a unifying mathematical law that governed the potential for ridesharing in cities of diverse sizes and traffic characteristics. According to the results of shareability predictions, even for regions with low trip density, and with allowing delays no longer than Δ=5minutes, the potential for sharing was massive. A dramatic increase and a modest increase of shareability could be witnessed in regions with low trip density and regions with high trip density respectively, which meant that the growth rate of shareability was in negative proportion to the average trips per hour.

Gurumurthy et al (2019) concluded that the use of SAVs with DRS was beneficial to the system, especially to congestion pricing scenarios. They simulated travel patterns in Austin with the implementation of personal AVs, and shared AVs, by using MATSim. Compared with private AVs, SAVs' fixed costs were assumed to be 50% lower ($0.125 per trip) and distance-based costs of SAVs were also 50% lower ($0.1/mile). Since AVs, SAVs and DRS options could make motorized travel easier, extra VMT and congestion would exist. A no-toll scenario simulated was self-explanatory. The pricing scenario was set to $0.10/mile in morning peak hours (7–9 am) and $0.5/mile during the evening peak hours (5–7 pm) in all major network links. Except the base case (without AVs or SAVs), the no-toll scenario and the pricing scenario, with a 50% discount and 75% discount for these 2 scenarios, were tested as 6 sub-scenarios by applying SAVs' reference fare. The results showed that the base SAV fare level attracted a strong SAV use. The SAV use was predicted to decrease 25% of the regional VMT. The pricing was indeed estimated to improve shared-use uptake and reduce congestion. On an average, the VMT was reduced by about 15% compared with the no-toll scenario. The maximum AVO was 1.32 for small SAV fleet (500 SAVs), and with such small fleets, the VMT can be reduced by up to 40%. On an average, the fleet continued to generate more than $30 a day per SAV per day, which was relatively high, when compared with the revenue of existing fleets of transportation network companies (TNCs), since there were no costs incurred by drivers, which were a main part of operating costs of TNCs.

Martinez and Viegas (2017) examined an implementation of a shared and self-driving fleet of vehicles in Lisbon, Portugal. Two kinds of self-driving vehicle concepts were built. One was shared taxis, which can provide a real time ordering and door-to-door services with 8 seats per taxi and the waiting time of 5 to 10 minutes. The other was taxi buses with 8 or 16 seats, which required pre-booking at least 30 minutes in advance. They could provide a dynamic bus-like service, with designate stops no farther than 300 m away from the agents' doors. Four mode choices including walking, subway or rail, shared-taxies and taxi-buses were tested in Lisbon which was divided in a homogeneous grid of 200 m x 200m cells as the spatial resolution of the model for simulation. Unlike the other SAV simulation, instead of a logit model, the travel modes of agents in this paper was prepared in advance based on trip lengths and durations. In addition, all agents were forced to take taxi-buses if they did not select the previous 3 options. Shared trips were matched in the beginning rather than matching during the trips. The results showed that shared taxies had about 45% mode share throughout the day, with the efficiency of taxi-buses increasing during peak periods and subway and rail were replaced by taxi-buses significantly in night. Shared taxies could reduce approximately 55% costs per kilometer compared with taxi services. For taxi-buses services, their average costs per kilometer were estimated to be 40% lower than the current public transport fare. Shared and self-driving fleets could also bring considerable traffic improvements. Likewise, congestion would have a 30% reduction with much more vehicle uses. This would reduce increasing current private vehicles and could provide younger and environmentally friendly fleets.

Another study case of Lisbon was conducted by the International Transport Forum (ITF) (2015). It developed an agent-based model based on real trip-taking activity to simulate a shared mobility system with SAVs. As for the limitation of waiting time, this work had the standard that waiting time should be no more than additional 20% travel time of single trips or 10 minutes, which was the same as the rule in the study of Zachariah et al. (2014). Several scenarios were generated by 4 main parameters including the status of ridesharing (with or without ridesharing), the status of availabilities of public transit (with or without availabilities of public transit), penetration rates of SAVs and time periods. Like other study cases, the DRS method also required matching DRS trips in the beginning of the first trip rather than during a trip. The results showed that if public transit was not available, SAVs took 50% total trips in the simulation. However, the total travel volume (Car-km) in the city was estimated to be impacted negatively by the implementation of SAVs. The growth rate of the total travel volume in the scenario with DRS was 22.4%, while the value in the scenario without DRS was 89.4% if there was 100% SAV implementation. The paper indicated 100% SAV implementation would lead to a clear decrease of circulating cars in peak hours, but with 50% SAV implementation, the congestion was likely to become worse, compared with the base case without SAVs. SAV fleets also were estimated to have positive impacts on the parking space. Even in the worst scenario, 2.12 km^2 (2.5%) of Lisbon's urban area could be released by eliminating all 110,000 off-street and 24,379 (49%) on-street parking lots.

Chapter 2: Datasets

This chapter introduces 2 major input files to simulate SAV fleet operation across MSP and Minnesota: MSP network and a 24-hour trip table in a weekday.

NETWORK DATA

The network file comes from OpenStreetMap (OSM). OSM is an open source website which provides free editable maps of the world with detailed geographic information system (GIS) data including land uses, roads and other geographic information. It allows any user to extract large network files by using overpass application programming interface (API), which is a mirror of the OSM database. After being converted to a shape file, the extracted MSP network is shown as follows, in Figure 2.1.

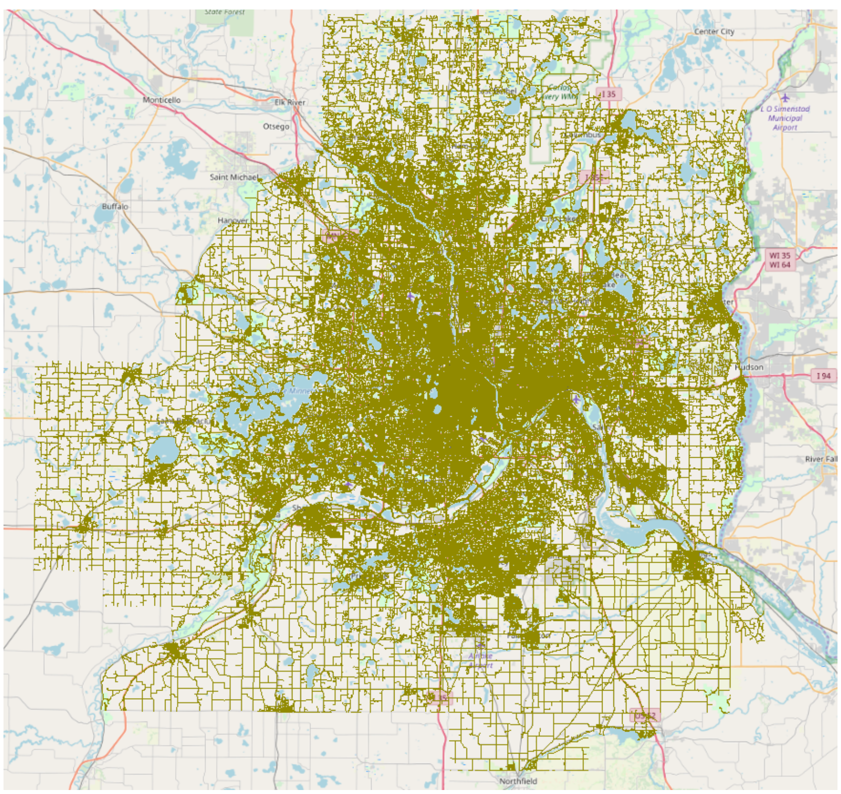


Figure 2.1: MSP's 7-County Network.

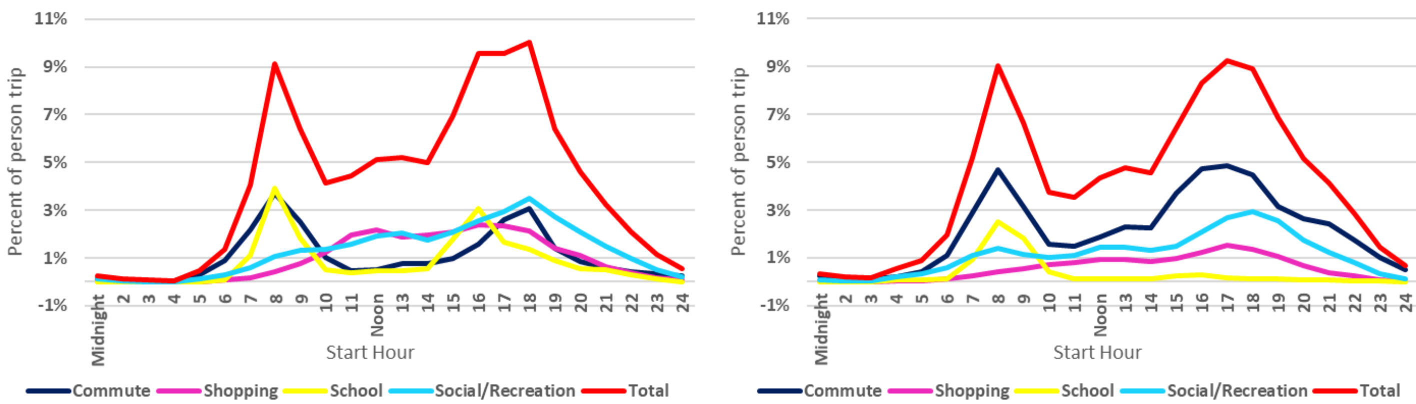
The network file includes 7 counties of the Minnesota state with 42,485 directed road links and 20,746 nodes. It also contains coordinates of nodes and basic information of each link, such as connected nodes, length, free speed, capacity, number of lanes and available travel modes.

TRIP DATA

Nearly 11 M person-trips for a typical weekday (when school is in session) were provided by Metropolitan Council, with each trip identified by a person ID, a household ID, the person type, trip mode, trip purpose, origin TAZ, destination TAZ, trip distance, departure time and arrival time. The person types are classified as a child, non-working adult, senior, part-time worker, full- time worker or an adult student. The trip purposes include school, work, university, meal, shopping, personal business and social recreation. There are 7 trip modes observed, including drive alone, shared rides, walk to transit, park-and-ride, bike and school bus. This study assumes that all demand is satisfied by using SAVs. Therefore, the dataset's selected modes were not used. External trips and truck trips are also not included in this dataset or in this work's SAV fleet assignments, since they come from far away or require large vehicles, and Metropolitan Council did not have departure times or tours for them. As a result, the congestion levels in this thesis' simulations lack some expected congestion that would lengthen travel times and perhaps extend many SAV response times.

Figure 2.2 shows departure time choices for person-trips by trip purpose in the MSP data set and in the U.S.'s 2017 NHTS. It appears that all trip types in Figure 2.2 have both AM and PM peaks excepting MSP's school and shop trips, which are low and flat (and perhaps too low and flat to be realistic) across all afternoon hours. The NHTS data set also suggests more shopping trip departures across most times of day, with more of a mid-day peaking pattern. NHTS social/recreation trips exhibit mid-day and PM peaks.

On average, MSP travelers make 4.36 person-trips per weekday versus just 3.37 trips/day in the NHTS data set. Such trip-generation differences are striking and suggest that NHTS respondents are under-reporting or MSP data are biased high. Daily person-miles traveled (PMT) in the MSP data for a typical weekday is around 34.3 miles, versus 39.0 miles/weekday/person in the NHTS data set, suggesting that MSP person-trips are relatively short.



a. Time of Day from 2017 NHTS

b. Time of Day from MSP Data Set

Figure 2.2: Distribution of Person-Trips by Trip Purpose and Trip Start Time (based on 1-hour bins).

Another important point of comparison, beyond trip rates by purpose and time of day, is trip distances, as shown in Figure 2.3's histograms. To equitably compare trip lengths in the NHTS vs MSP data sets, only trips of one-way distances under 70 miles were used in the NHTS 2017 (which has very long-distance trips that go between nations and reigons, rather than staying within a single region, like the MSP trips do), with the same modes available in the MSP data set. As evident in Figure 2.3, the biggest difference in the NHTS 2017 and MSP distance data sets occurs in person-trips between 0 and 20 miles (one-way). For the [0, 5 miles) distance range, NHTS data has 8.2% more short trips (as a share) than the MSP data set shows (i.e., 62% vs 5%??). To make up for this deficit in short trip distances, the MSP histogram bars are higher than those in the NHTS data set over the [5, 10 mi), [10, 15 mi) and [15, 20 mi) ranges. The reason for this may be that MSP does a better job surveying travelers, capturing many short trips neglected in the NHTS. Or the survey instructions differ, so that MSP collects more short trips. Or it may be that the agent-based modeling methods MSP modelers are using to create the itineraries used here have a bias toward creation of short trips.

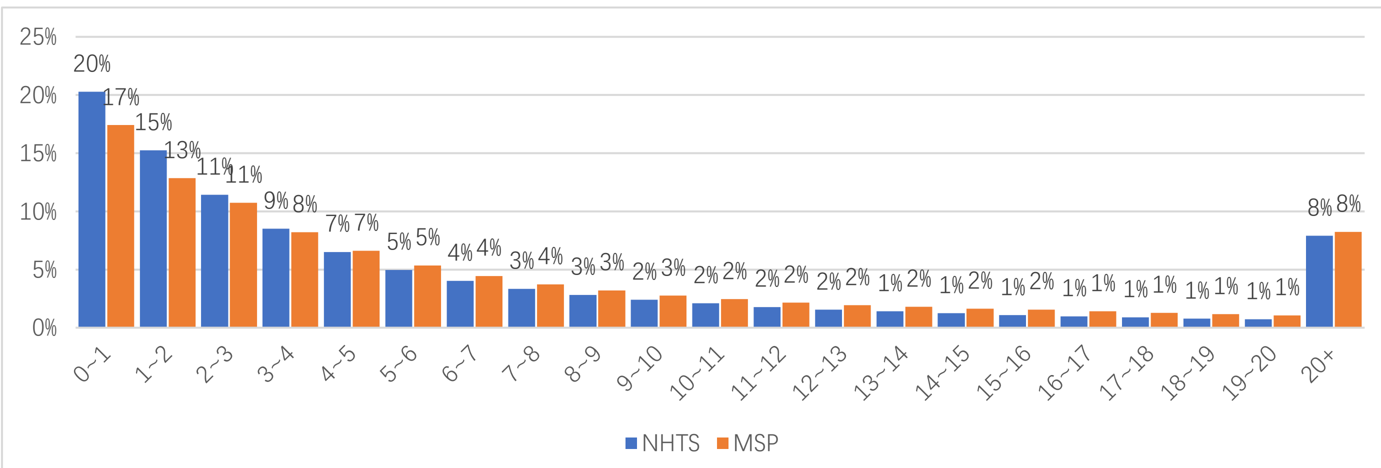


Figure 2.3: NHTS vs MSP Trip Length Histograms (based on 1-mile bins).

Chapter 3: Methodologies Used

This chapter contains all the steps to simulate SAV operations across the MSP region, starting from data inputs all the way through the various simulation scenarios. The first step is quite complex, and involves SAV generation and assignment to travelers, including some ride-sharing assignments, and then SAVs to routes, for endogenous estimates of travel times, then updating of departure times and mode choices. Parking lot generation for busy locations is also important code, once those busy locations are identified. Such JAVA codes for use with MATSim are illuminated in the Appendix to this thesis.

TRIP COORDINATES GENERATION

Metropolitan Council person-trip start and end times are provided in rather coarse 30-minute bins, and their origins and destinations are grouped into/aggregated by TAZ. There are just 48 half-hour bins in a day and 2485 TAZs across this 6364 square-mile region. For effective agent-based simulation of SAV fleet operations, across tens of thousands of roadway links, with updates every second on vehicle assignments and position, much higher temporal and spatial resolution are needed. To respect Texas Advanced Computing Center Wrangler supercomputer run-time restrictions (of 48 hours), only conditions in the MPO's 7 counties (Anoka, Carver, Dakota, Hennepin, Ramsey, Scott and Washington counties), located in the center of Minnesota State (as shown in Figure 3.1) were used, rather than trips that ended in another 12 MSP-area counties that the Metro Council also provided (since they like to keep track of 7-county residents' movements in a halo region around their model region).

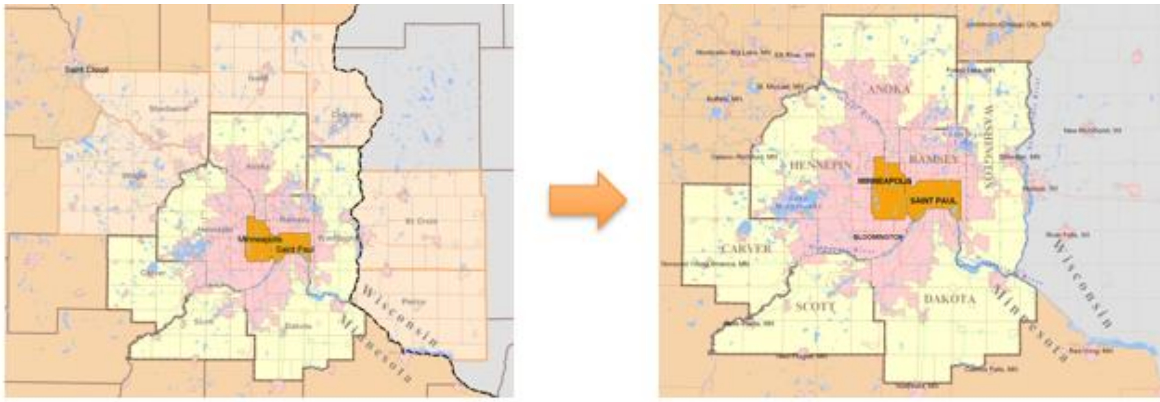


Figure 3.1: Moving from 19 County Trip Data to 7 Counties for the Modeled MSP Region.

One-minute bins were used here for obtaining the detailed departure and arrival times for each agent. The departure times were disaggregated by first spreading all binned trips' start times across the 30 minutes uniformly, and then adding a random number from a uniform distribution with a mean 0 and standard deviation of 15 minutes to help smooth trip-rate transitions across all the 30-minute bin endpoints, as discussed in Gurumurthy and Kockelman (2018). Once a departure time was adjusted for these spatially disaggregated trips, departure times could be estimated using the Met Council's highway travel-time "skim" values (for each of 4 broad times of day: AM peak, midday, PM peak and night).

Home-based trip origins and destinations were disaggregated using Python code and an ArcGIS package to generate specific coordinates for each person's home location, uniformly spread (in 2D space) across the associated TAZ, and then associated with the closest OSM roadway link, to ensure each home site is accessible.

Instead of spreading all non-home trip ends uniformly across TAZs, 5 types of non-home "facilities" or sites were created, to provide some natural within-TAZ aggregation of jobs and businesses. Sites for individual work, shopping, social contact, and school activities tend to be clustered in larger buildings, rather than smaller, often-separated dwelling units. To help avoid many unrealistic, crossed paths by travelers and unrealistic or wasted routings, and enhance opportunities for DRS, these 5 trip-end site types were created. Their locations are randomly generated in each TAZ. The numbers of sites in a TAZ are determined by the numbers of trip-ends (e.g. 1 new work location per 2000 work trip ends). Each TAZ has at least 5 trip-end sites for 5 types if the trip ends are less than 2000.

TRAFFIC ASSIGNMENT

MATSim is based on dynamic traffic assignment (DTA) code in tandem with person-journey optimization (over departure times, mode choices, and route selections) and equilibration to approach a Nash equilibrium or stochastic user equilibrium. There are 5 stages in the MATSim operation process: input files loading, mobility simulation, scoring, re-planning, and analyzing (Horni, 2016). After scoring, re-planning will be the end of an iteration, while mobility simulation and scoring will be executed for the subsequent iteration. Once user equilibrium or the max number of iterations is reached, this loop will be broken, and the results of final iteration will be analyzed. In this study, since there is no convergence criterion given by MATSim, the trip files for SAV simulations are obtained from the simulations with private vehicle and walk as modes. After 50 iterations, the trip tables can be close to an equilibrium. The first step is the loading of input files, including the network files and agents' travel plans or itineraries for the day to be simulated. MATSim requires an initial demand file (called "population" or "plan") for a day or more of travel choices. A tour contains a circle of activities (e.g., start at home, go to work, go to school, return home), coordinates of starting and destination locations, modes and departure time preferences, plus any useful but optional information (like detailed travel routes and exact travel times to use for some or all travelers).

MATSim updates all agent positions on one-second intervals, and the first network loading or "iteration 0" in this "mobility simulation" simply assigns the initial demands to the network with existing scheduled travel plans. Agents' choices start affecting one another in iteration 1. For example, if too many agents plan to leave at a same time or go to a same place, they might use same roads in the network, generating a traffic jam in the simulation. The simulated agents' plans may not align with the original agents' plans that describe their expected travel time. Once the mobility simulation of the plans is completed, scores are allocated for evaluating each agent's plan. The scoring function used in this study is the Charypar-Nagel (2005) scoring function, as shown here:

$$\mathrm{S} = \sum_i (U_i^{perf} + U_i^{late}) + \sum_j U_j^{leg} \tag{2-1}$$

where U_i^{perf} is the utility of performing activities i; U_i^{late} is the penalty of arriving late; and U_j^{leg} is the utility of mode j.

A more detailed function can be found in the MATSim user guide (available at https://www.matsim.org/docs/userguide/). The scoring process ensures that on-time departures and arrivals increase agent scores while late or early arrivals lower their scores. The parameters of the scoring function are customizable. The agents' travel choices are modeled in MATSim through an iterative learning mechanism based on a quantitative score, which is referred to as utility. For each iteration, agents choose from an existing set of daily plans according to a multinomial logit model. During the simulation, the 5 highest scores of agents' plans are always restored. For each iteration, if the newest score is greater than the lowest of the 5 highest scores, it will replace the lowest score. After the entire simulation, MATSim selects the plan with the highest score and converts it into an event file, recording the events (e.g. vehicles picking up and dropping off, vehicle entering or leaving a specific link) by each second. The event file is important for the analyzing step, as described in the subsequent section.

As mentioned above, agents may generate congestion in the network and can be influenced by the same. During the re-planning step that follows scoring, in order to avoid congestion that may decrease the scores and determine a more optimal plan, MATSim will improve their plans based on co-evolutionary algorithms by adjusting the routes and modes of each of these travelers. Then, the simulation will move on to the next iteration and run network loading and scoring processes again. With each iteration, the aim is to determine a greater average score of the executed plans for agents. Note that the scores can only be referred in one scenario to find the best solution. The comparison of different scores in different scenarios is meaningless, as scores depend on the parameters of the scenarios.

At the end of the simulation, the key performance values are mainly taken from the plan file and the event file. For instance, the mode shares, time of day, and trip duration from the plan file will be analyzed, while the event file contains VMT, eVMT, and response time of SAVs. Some analyses can be obtained automatically at the end of the simulation, and MATSim also provides a customizable analysis function for post-processing.

SAV CODE

Maciejewski and Bischoff (2017) coded the AV mode into MATSim. Compared to the operation of privately owned and operated AVs, SAVs are considered a new ride-hailing or TNC option (such as Lyft, Didi, and Uber) or a new type of taxi service. Unlike conventional services, SAV services are not affected by the drivers; it can satisfy demands as much as possible with a considerably shorter response time. SAVs also have the advantage of unlimited working time to reduce costs. In addition, excluding the costs incurred by drivers yield further savings, which can also benefit the users with lower service prices. This study ignores external commercial-vehicle trips and commercial-vehicle trips (about 16% of traffic) New, latent or induced demands by SAVs are also ignored. In the MATSim simulation, SAVs satisfy agents' demands according to their travel plans. New SAVs are randomly generated in the network. The SAV fleet size and SAVs' locations are saved after iteration 0 for all subsequent 24-hr-day simulations. In the next iteration, after serving the first group of travel demands, SAVs respond to agents' requests within a predefined service radius. If there is no SAV response, the agents will shift to other modes in the next iteration. If an agent's demand is responded to, the assigned SAV will move from its current location to the agent's location. Specifically, agents' requests can be presented a few minutes earlier or later than when the individual finishes their last trip. In short, SAVs and agents can wait for each other till both are ready. After trips are completed, SAVs will become available for the next request and will park on curb and wait for the next request so that they only cause congestion when they respond to a demand. This issue can be resolved by adding a parking function in the simulation code.

PARKING SIMULATION CODE

The underlying parking strategy of the SAV simulation is based on the Autonomous Mobility on Demand (AMoD) simulator, as developed by ETH Zurich and the Institute for Dynamic Systems and Control (Ruch et al., 2018). As discussed above, after passengers leave SAVs, these vehicles will be excluded from the simulation network and assigned to fake links in another dimension. This is not in accordance with real life, since vehicles are still on the roads when the agents arrive at their destinations. In other words, empty SAVs are not directly modeled in MATSim or loaded on the network even though they can affect the traffic in the network. Related to this idea of empty travel by SAVs, in between drop-offs and pickups, there is the idea of excessive curb use by idling/sitting-still SAVs that have not yet received their next service call or person-trip assignment. Many popular addresses in the region could have long queues of SAVs picking up or dropping off passengers, creating excessive curb space and lane-level congestion. On the links with at least 400 trip origins and destinations per curb per day, parking lots were created nearby to allow SAVs to exit the roadway until they have received a new trip assignment. Each SAV parking lot's capacity is assumed to simply be the SAV fleet size divided by the number of parking lots, minus 20 percent (to recognize that not all SAVs will need parking at any one time, thanks to uncongested curb spaces in much of the region and many SAVs during daytime hours serving trips, etc.), as follows:

$$Parking\ capacity\ = (\#SAVs/\#parking\ lots) * (1 - correction\ factor) \qquad (2\text{-}2)$$

During the simulation, parking requests are satisfied every 5 seconds. After a trip is completed, the empty SAV will locate the nearest and second nearest links with parking lots and check if the first nearest parking lot is available. If the first parking lot does not have any spaces, the second nearest parking lot will be checked. If both parking lots are full, entailing that the area has enough idle SAVs, the SAV will randomly choose an available parking lot in order to improve the SAV fleet operation of the entire simulation. Once a parking destination is decided, the SAV will follow the best route to the parking lot. After arriving at the parking lot, the SAV will stay on the parking lot and wait for the next request. One weakness of the parking strategy is evident here: The parking trip cannot be canceled during its execution, which means that the SAV that has initiated the parking protocol will not be available for agents before it arrives at the parking destination even if its parking trip conflicts with agents' requests. Future works should be executed in order to solve this problem and allow SAVs to cancel parking trips automatically when agents who are nearby have requests.

DYNAMIC RIDESHARING

The DRS code used in this study is adapted from Claudio et al. (2018) and Fagnant et al. (2015). In MATSim, the dynamic vehicle routing problem (DVRP) module (Maciejewski et al., 2017) is implemented for SAV simulation and allows for dynamic and demand-responsive vehicle dispatch, similar to taxi operation. Vehicle dispatch is generally initiated the moment an agent wishes to depart using such a mode (Simoni et al., 2019). Based on the module, all SAV trips are matched for DRS. A least-cost path algorithm in MATSim is used in the code for optimizing collocation and determining aggregated trips for SAVs within acceptable distances for pickup. Fagnant et al.'s (2015) DRS matching constraints are used here and can be summarized as follows: Constraint 1: Passengers' trip duration increases should less than 20%. Constraint 2: Passengers' remaining trip time increases should less than 40%. Constraint 3: The total trip time for second or subsequent trips increases by ≤ 20% of the total trip without ridesharing or by 3 minutes. Constraint 4: Second or subsequent travelers will wait up to 10 minutes. Constraint 5: Total planned trip time to serve all passengers ⩽ remaining time to serve the current trips + time to serve the new trip + drop-off time, if not pooled. Constraint 5: Total planned trip time to serve all passengers ⩽ remaining time to serve the current trips + time to serve the new trip + drop-off time, if not pooled.

Figure 3.3 shows the DRS flow diagram, including a couple parking module decision steps.

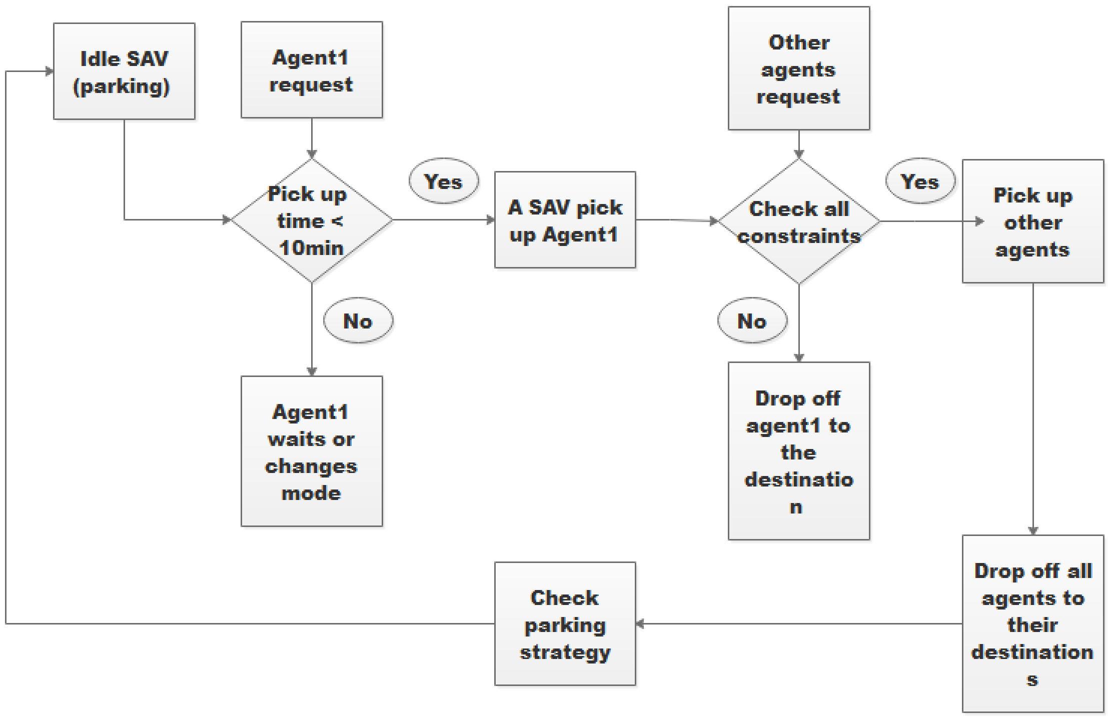


Figure 3.2: SAV DRS Trip Simulation Flow Diagram.

SCENARIO ASSUMPTIONS

7-county SAV-fleet simulations were run with various fleet sizes to appreciate the variation in system performance metrics. The 22 different scenarios' results are compared here. Using the full, 7-county region, this work simulated 456,800 person trips (5% of the region's total 9.5 million person trips) over a 24-hour period. For the Twin Cities scenario, about 487,000 person trips were simulated from the dataset. Different fleet sizes are used to understand wait time and mode preference. A base scenario was studied as the business-as-usual (BAU) case by simply simulating the travel demand obtained from the local metropolitan planning organization (MPO) without enabling SAV use. The agent itineraries, network, and activity locations were processed to obtain the BAU metrics for VMT samples of 2%, and 5% of the total trips are used rather than the full population given the long runtimes. The results of the BAU case were calibrated using the dataset travel times by modifying the flow and storage capacities of links for realistic sample simulation. Personal AVs and SAVs are 2 trends in AV usage. On the one hand, with the development of technologies, the costs of AVs will decrease and may become affordable one day. On the other hand, transportation network companies have already tested SAVs. SAVs are more suitable for these companies from a business model point of view, and the cost of operation of SAVs will be relatively lower than personal AVs. SAVs are implemented as the only transportation mode in the scenario used in this study. That is, regardless of trip modes information in the dataset, all trips were satisfied by using SAVs. Based on the results of 5% of the total trips simulation, some scenarios were simulated without DRS, which means each SAV can only service one agent a time. Fleet sizes were also altered for different scenarios to understand how fleet size affects trip patterns. Fleet sizes may have the greatest impact on VMT/eVMT, idle time, and travel delay, since SAVs need to spend more or less time to arrive at the start location. Furthermore, the simulation of the Twin Cities area (Minneapolis and Saint Paul), which has a higher population and trip density is more valuable for SAV operation in the foreseeable future compared to simulations of the large 7-county area. Since the simulations across the Twin Cities only consider the trips with both their origins and destinations within the Twin Cities area, extracting those trips from 5% of total trips will decrease the population and trip density compared to scenarios with the 7-county region. In order to balancing this influence, 20% of trips within the Twin Cities are simulated.

Chapter 4: Simulation Results

This section analyzes the SAV performances based on the results from various simulation scenarios.

Scenarios with Curb Parking Permitted Everywhere

In this study, 22 different scenarios are simulated here to be compared based on their performance metrics. The results suggest that an SAV in the MSP region can serve about 30 person trips per day, on average; thus, this replaces about 6 or 7 household vehicles (assuming no one needs to leave the region) but generates another 20% VMT per day and adds congestion to the network. Those using DRS spend time waiting for other passengers to enter or exit SAVs, which often go out of the way to pick up and drop off others, effectively increasing the average trip duration by 34% per day.

Different SAV fleet sizes affect the matching success rate that affects how many shared rides are observed. Furthermore, travel times of the networks and average wait time can also be impacted. Table 4.1 shows the results in terms of scenarios and fleet sizes. SAV fleet sizes are represented as the number of travelers per SAV per day in order to illustrate the influence of fleet sizes across scenarios with different populations. eVMT shows the negative effect of an SAV fleet. It is generated when an SAV receives a request and comes to the passenger who called the SAV. Unlike the conventional vehicle, eVMT cannot be avoided with SAV implementation. A low eVMT entails the high efficiency of using SAVs and can also help reduce congestion in the network, along with emissions. The SAV runtime represents the average working period of an SAV in 24 hours. AVO is the average vehicle occupancy for evaluating the effect of DRS in the network. The average wait time is another significant result of the efficiency of DRS from the traveler agents' point of view. The vehicle replacement rate represents the productivity of SAV implementation.

Each conventional vehicle performs 3.05 trips per day, on average, as per the NHTS data (Fagnant and Kockelman, 2016). The average number of served trips per conventional vehicle divided by the average value of served trips per SAV per day is the vehicle replacement rate. Revenue is the sum of the traveling expenses in a 24-hour simulation.

For 2% of the total trips scenarios without DRS, coupled with the growth of travelers per SAV per day (reduced fleet size), the average VMT and eVMT go up, causing a surge in the operation time of each SAV. The average waiting time for individuals in several scenarios' ranges from 2.5 minutes to 13.7 minutes, which is consistent with the actual waiting time of Uber or Lyft. For scenarios with DRS, 6–33% of the simulated trips are DRS ones. With smaller SAV fleets, the proportion of the DRS trips increases due to the fact that the lower availability of SAV prompts individuals to opt for DRS trips. The average VMT and eVMT decline sharply since the DRS can respond to multiple trips at the same time and choose the most economical route to pick up passengers. The values of AVO are relatively low, since there is a low trip density of 2% across the 7 counties. The average waiting time becomes slightly longer due to the decreased SAV fleet size. As the number of travelers per SAV per day rises from 10 to 15, there is an increase in the average waiting time per trip from 11 minutes to 40 minutes, as SAVs cannot satisfy all demands at the same time; consequently, some SAVs have to first finish some orders and then come back for the rest. However, since those scenarios involved 2% of the total trips across the 7 counties, the spatial dispersion resulted in 6% unserved trips per day. In order to avoid this impact, it is recommended that a higher population density be taken as a target parameter in a region.

Region and Trip #	DRS?	Travelers per SAV per Day	VMT per SAV per Day	Empty VMT (%)	SAV Run Time per Day	%Trips as DRS per Day	Trips per SAV per DAY	AVO	Avg Wait Time per Trip (min.)	Unmet trips (%)
7-counties, 2% of total trips	No DRS	5	175 mi/day	12.7%	9.4 hr	--	19.0trips	1 person	3.7 min	5.7%
		10	406	24.8%	11.4	--	37.9	1	11.0	5.7
		15	557	22.3%	18	--	55.4	1	39.9	6.9
	Yes DRS	5	170	11.7%	8.4	5.9%	19.0	1.03	4.0	5.7
		10	378	22.8%	11	15.7%	37.9	1.23	10.7	5.8
		15	526	22.2%	16.5	43.4%	34.3	1.80	34.5	6.5
7-counties, 5% of total trips	No DRS	5	173	14.1%	8.9	--	20.2	1	2.5	0.3
		7	277	18.1%	10.5	--	28.0	1	4.9	0.6
		10	432	25.2%	12.5	--	40.0	1	13.7	0.6
		15	559	23.0%	14.5	--	54.6	1	36.1	3
	Yes DRS	5	174	10%	8.4	12.4%	20.0	1.14	3.7	0.5
		7	254	14.5%	9.6	20.3%	28.0	1.23	4.6	0.5
		10	261	19.7%	10.9	26.3%	40.0	1.41	9.7	0.5
		15	514	20.0%	15.3	42.5%	59.3	1.84	32.3	1.8
Twin Cities 20% of total trips	No DRS	5	117	9.5%	4.3	--	15.9	1	2.5	1.5
		7	170	13.0%	6.1	--	22.3	1	3.2	1.5
		10	253	17.0%	6.1	--	31.8	1	3.9	1.5
		15	414	23.4%	7.9	--	47.8	1	11.9	1.6
	Yes DRS	5	109	7.2%	4	20.7%	15.9	1.28	2.9	0.1
		7	156	10.0%	4.6	25.2%	22.3	1.32	3.6	0.2
		10	227	13.3%	5.9	30.4%	31.8	1.56	3.6	0.5
		15	347	17.4%	7.2	38.8%	47.8	1.63	7.1	1.5

Table 4.1: Key Findings from 22 Simulation Scenarios.

The simulations with 5% trips have a larger trip density across 7 counties, leading to only 1.3% of simulated trips being unserved. Compared to the results from the same scenarios for 2% trip simulations, the VMT from the scenarios without DRS relatively increase, which is considered to be caused by the addition of 3% of trips with longer travel lengths; moreover, the eVMT decreases. This can be explained by the comparison of the

DRS-trip proportion between 2% and 5% sample simulations. For scenarios with travelers per SAV per day as 5, 10 and 15, the DRS trip proportions in 5% trip simulations increase by an average of 15%. Those increases are based on more opportunities for DRS trip matching, which lead to the decline in eVMT. The served trips in the scenarios using a 5% sample are more, and the average wait time is less. With the same travelers per SAV per day, individuals from the scenarios with different percentage of total trips face an equal probability of getting an SAV at the same time. However, with the increased fleet size in the scenario with 5% trips, the temporal travelers per SAV per day also increase. Each SAV would face more requests during a day, which will lead to more served trips and a shorter average wait time. With a smaller SAV fleet size, the values of AVO increase dramatically. The highest AVO achieved is 1.84 and is obtained with a small fleet serving 15 travelers per SAV per day. But it also yields the longest average wait time per trip of 32.3 minutes in the SAV-undersupplied setting. The SAV is expected to serve about 30 trips per day (Fagnant et al., 2015; Loeb and Kockelman, 2019; Loeb et al., 2018). Besides travelers per SAV per day as 5, 10 and 15, this study also simulates a scenario with 7 travelers per SAV per day, which represents 28 trips per day according to simulation results. Figure 4.1 shows the histogram of wait times of the scenario with and without DRS. About 62% of trip wait times are less than 5 minutes, mainly of 1–2 minutes. Compared to the 55% of trips with 0~5 minutes of wait time in the scenario without DRS, DRS can bring down the wait time of 68% of trips to less than 5 minutes. DRS reduces wait times significantly, especially for the trips with long wait times. The wait times for about 40% of trips that have a wait time of more than 11 minutes is reduced by DRS. Figure 4.2 and Figure 4.3 show average wait times during AM peak and PM peak across TAZs in 7 counties. 81% and 84% of TAZs with less than 6 minutes wait times are widely distributed during AM peak and PM peak, while only 1% of TAZs served by more than 10 minutes wait times. These figures show uniform wait times across the region and suggest residents of this region could get similar SAVs service level everywhere.

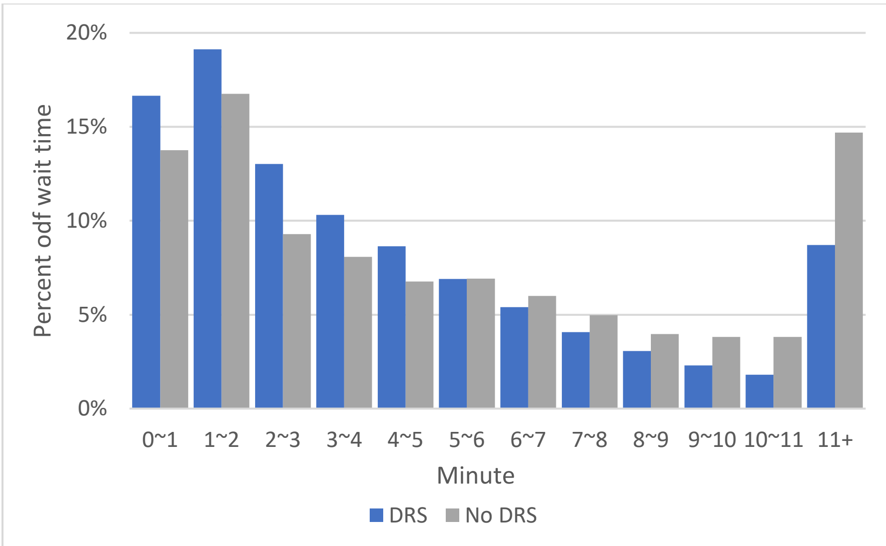


Figure 4.1: Wait Times Histogram across 7 Counties (assuming 7 travelers per SAV per day).

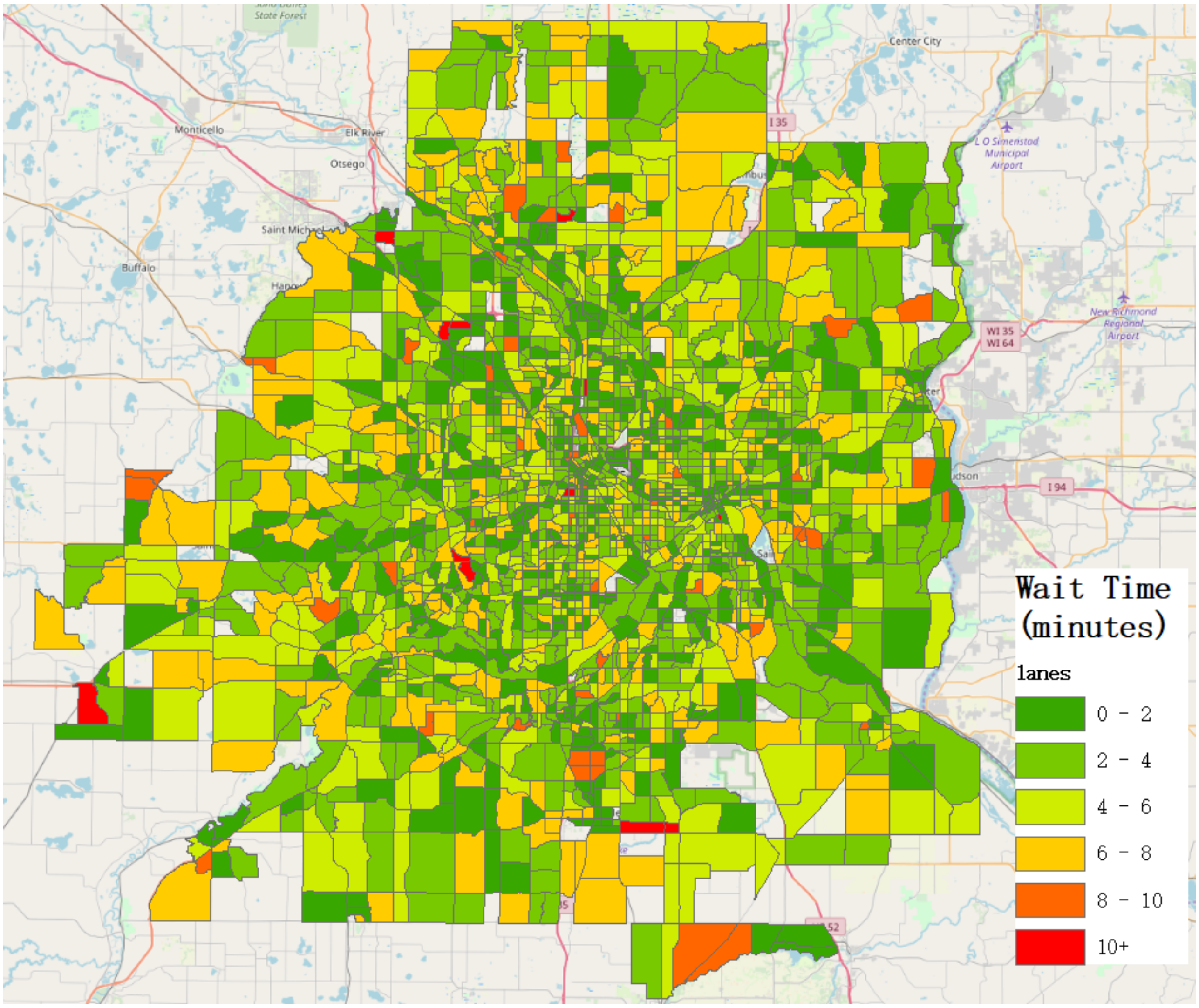


Figure 4.2: Wait Times during AM Peak across 7 Counties' TAZs (assuming 7 travelers per SAV per day).

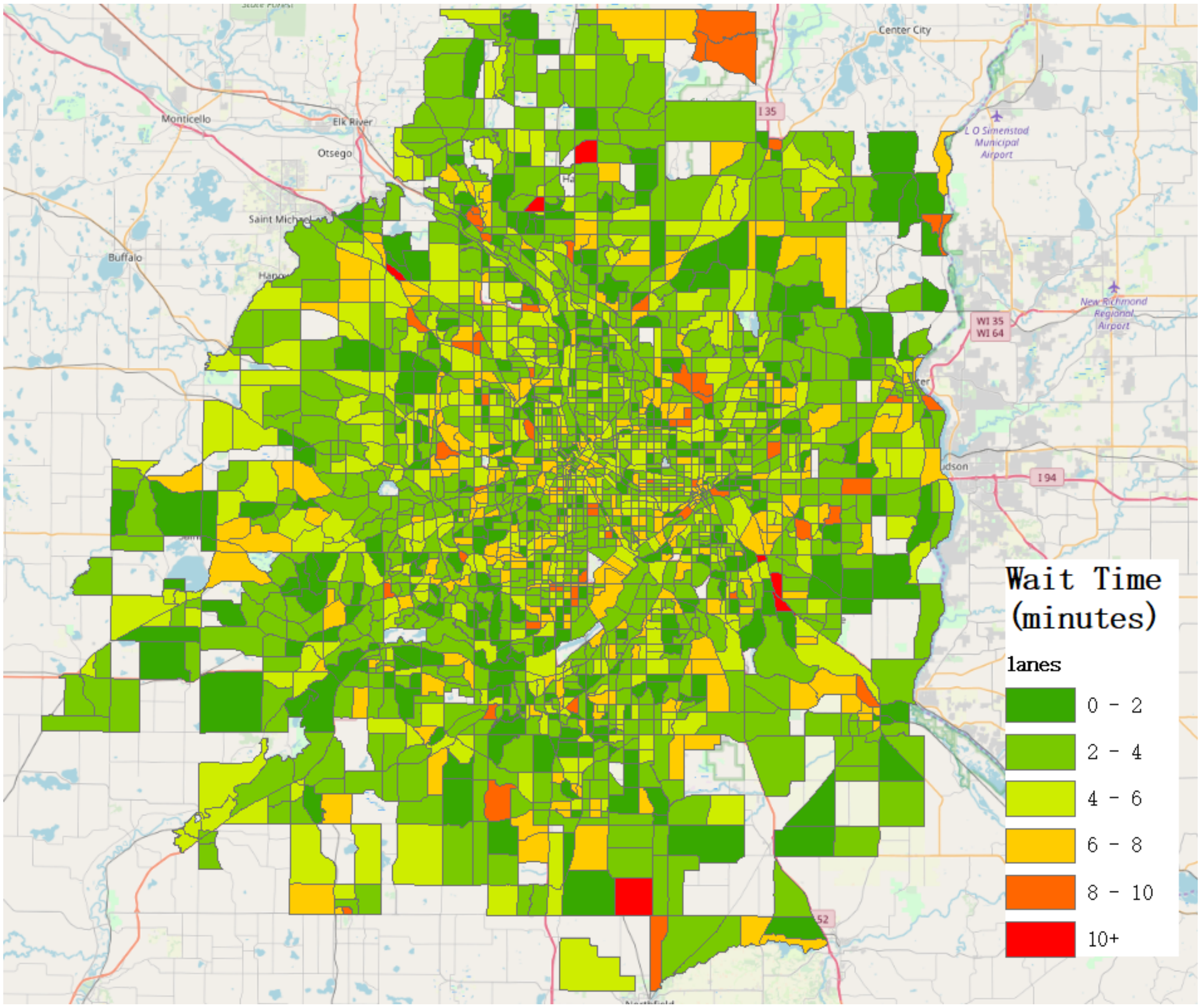


Figure 4.3: Wait Times during PM Peak across 7 Counties' TAZs (assuming 7 travelers per SAV per day).

The Twin Cities area in the MSP region is chosen to study the SAV operation. Compared with the results in the scenario of 7 counties, simulated VMT in the scenario of Twin Cities is significantly less, since the Twin Cities are much smaller than the 7 counties and the higher number of opportunities for DRS trips match the more concentrated trips of the area. The smaller area also yielded less percentage of eVMT per day. With similar numbers of simulated trips (456,800 trips in 7 counties and 487,000 trips in Twin Cities), the simulated trips in scenarios with 7 counties had longer travel distances and lower density. Lower density may result in a high percentage of eVMT because of the wide distribution of the trips. Thus, a high percentage of eVMT and long-distance trips may lead to more SAV run time per day. The proportion of DRS trips per day in Twin Cities increases from 20.7% to 38.8%, which is the highest average value among all the corresponding scenarios. It is noted that the number of trips per SAV per day is different in the 7-county scenarios and Twin Cities scenarios for the same number of travelers per SAV. For each traveler, the simulated trips in Twin Cities excluded those outside the Twin Cities. Although the same number of travelers per SAV and similar total simulated trips were used in these scenarios, the Twin Cities scenarios had more traveler agents and a larger SAV fleet. Therefore, the number of trips per SAV per day in Twin Cities scenarios is less than the number of trips per SAV per day in the 7-county scenarios. This could have a negative impact on AVO. However, according to the results of the scenarios with 5, 7, and 10 travelers per SAV per day, the values of AVO in Twin Cities are, on average, greater than the values of AVO in the 7 counties because more DRS trips may be generated in a small area. The average wait times of simulations with DRS are lower than those of simulations without DRS. This indicates that DRS reduces wait times. Agents find it difficult to find an idle SAV unless they are willing to wait until the SAVs drop other individuals and return for them, which will take up too much time. DRS can reduce the wait times in such situations. Among all scenarios, with decreased a SAV fleet size, there is a small impact on average wait times in scenarios across Twin Cities. This can justify why smaller areas relatively reduce the distance between SAVs and agents. With a larger trip density, the negative impact of a decreased SAV fleet size will incur more DRS trips. Thus, the values of AVO will also go up.

This study also considered the spatial and temporal analyses of VMT and eVMT across 7 counties. An extension code was created to extract VMT and eVMT, along with the links. In order to make the comparison more intuitive, Figures 4.4 and 4.5 show VMT and eVMT distributions across TAZs, respectively. As expected, most SAV VMT and eVMT occur on freeways and highways across the MSP region, especially the highway around the Twin Cities (of Minneapolis and Saint Paul). The TAZs with the most VMT were scattered around the downtown areas of the Twin Cities, since the two cities' central business districts (CBDs) have the highest trip-end densities. eVMT distribution is similar to VMT distribution. For example, Columbia Heights (in northern Minneapolis) and West Saint Paul (in southern Saint Paul) generate the most eVMT and VMT around Twin Cities, since they have a relatively few and dispersed trip ends. Since most VMT and eVMT are generated on freeways, it would be better if these SAVs arrived at the pick-up locations using secondary roads to reduce congestion on freeways. Like vehicles in reality, more drivers prefer to wait on highways rather than use idle secondary roads. But program controlled SAVs may perform better in this situation.

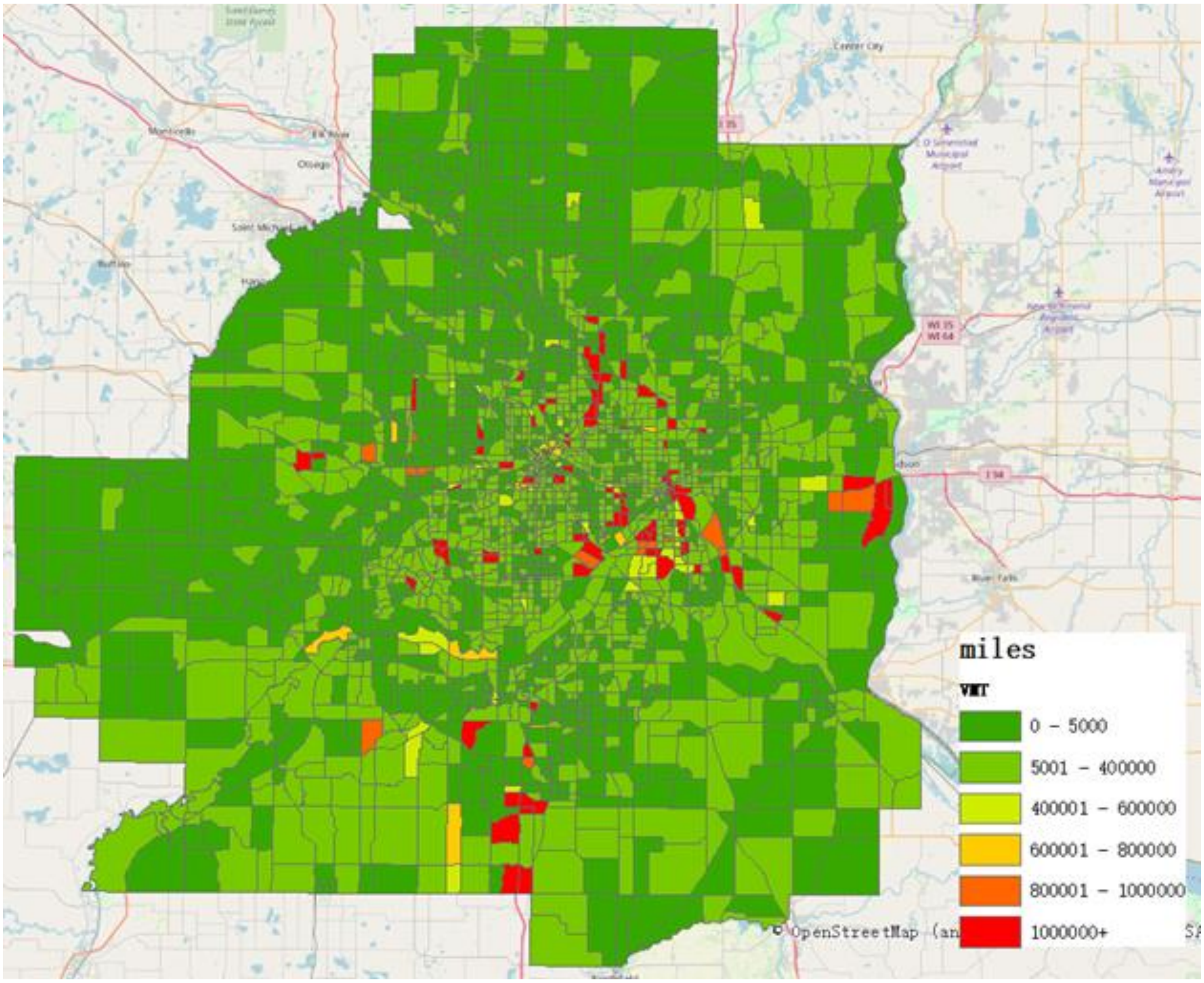


Figure 4.4: VMT Distribution across 7 Counties' TAZs.

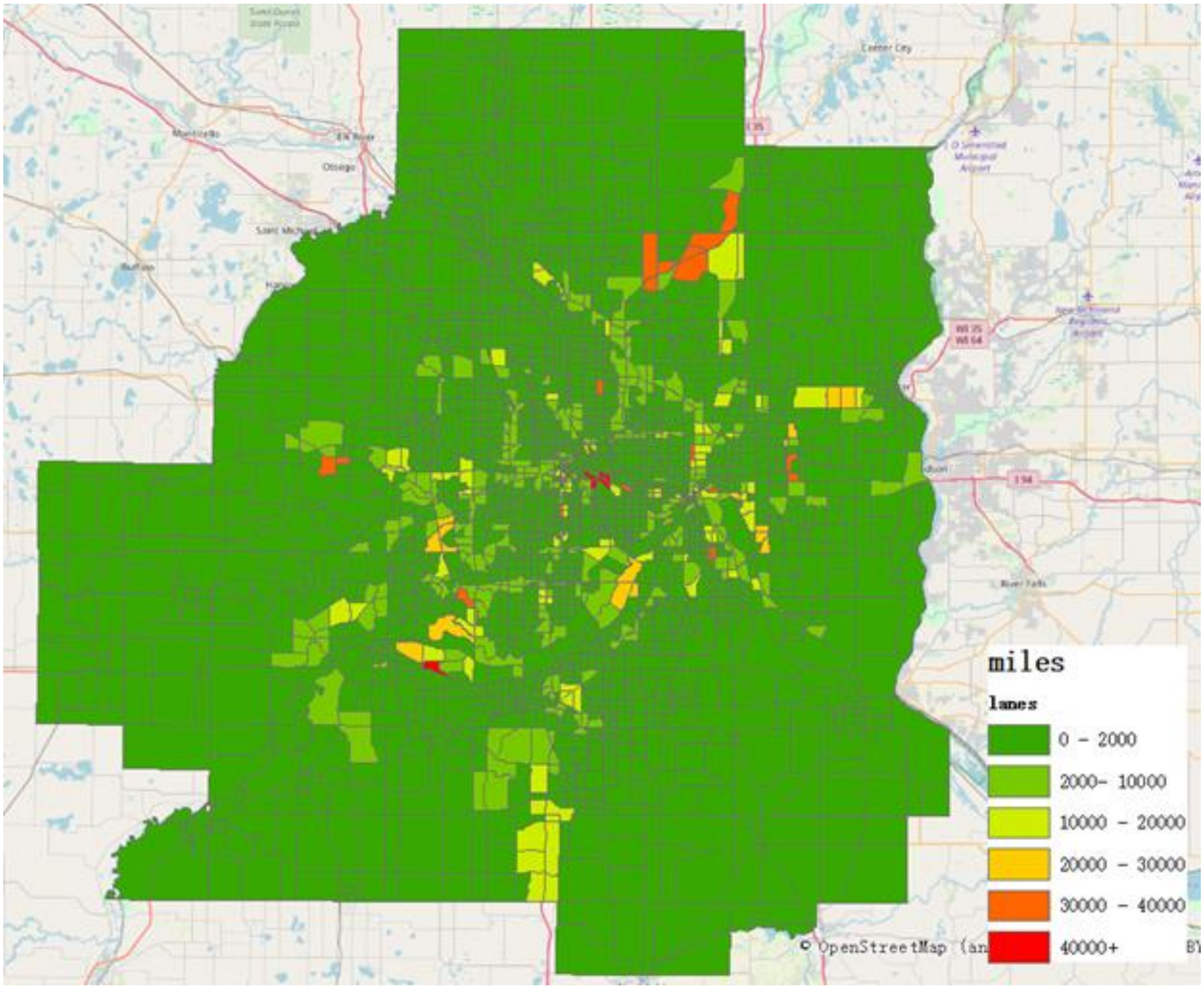


Figure 4.5: eVMT Distribution across 7 Counties' TAZs.

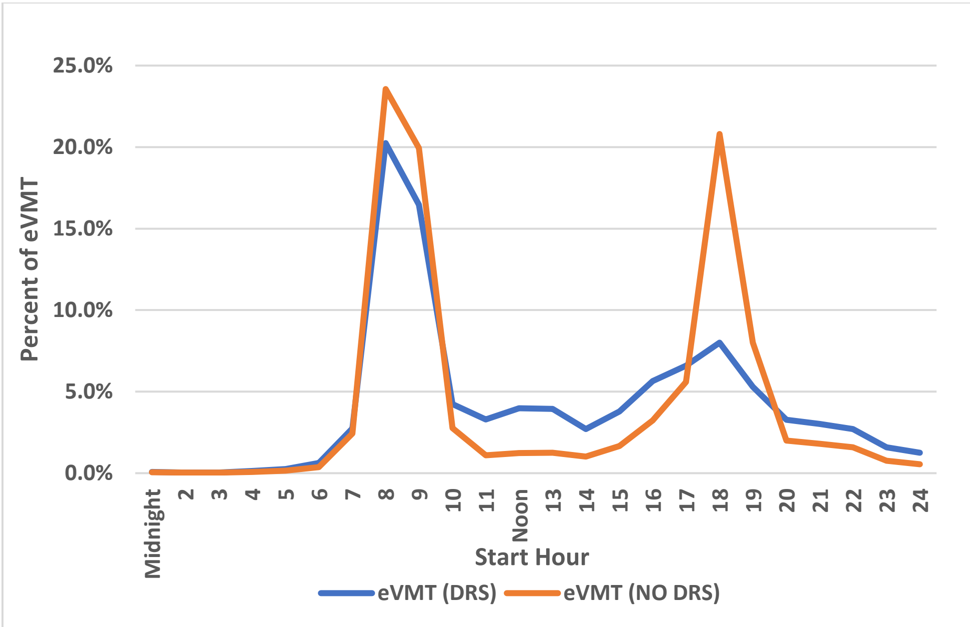


Figure 4.6: Distribution of eVMT with Trip Start Time (using1-hour bins, 7 travelers per SAV per day).

Figure 4.6 shows the eVMT of a day across Twin Cities. For the scenario without DRS, the AM peak and PM peak, which have numerous requests, are the main parts of eVMT distribution, and SAVs cannot satisfy those demands at the same time. Since DRS is not provided in this situation, SAVs can serve no more than one trip. As a result, more eVMT is generated due to the "coming back" processes. But when DRS is available, the eVMT will shrink because SAVs can serve multiple agents at one time, thus reducing the "coming back" processes. This obviously plays a role in the eVMT during the PM peak because more agents in this span share the same or close origins, especially in CBDs. Commuters need to go off work from their companies or other workplaces so there are more opportunities for DRS trip matching. Meanwhile, although eVMT also decreases during the AM peak with DRS, it only declines by 3% as compared to about 13% during the PM peak. The trips during the AM peak have opposite attributes. During the AM peak, more agents share the same or close destinations but with different origins. Due to widely distributed origins, SAVs cannot match many DRS trips, and centralized destinations can also centralize SAVs locations. This imbalance may lead to SAVs generating more eVMT to respond to subsequent requests. Figure 4.7 shows the distributions of response time, which have the similar trends with Figure 4.6 since the same reasons as discussed above.

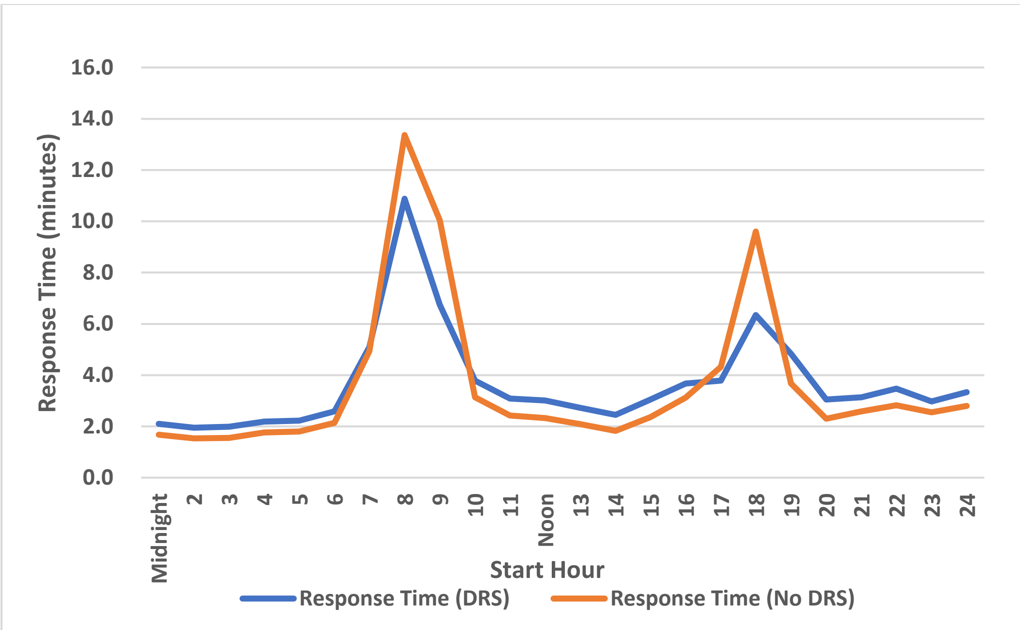


Figure 4.7: Distribution of Response Time with Trip Start Time (using1-hour bins, 7 travelers per SAV per day).

Added VMT means extra VMT for each agent from a DRS trip, which can cause congestion in the simulation network as compared to each agent's trip duration if the agent drives a private vehicle. This value also indicates the added VMT of SAVs. Figure 4.8 shows the added VMT of a day across the Twin Cities. As discussed above, DRS trips were mainly distributed during the PM peak. Hence, about 30% of the added VMT in a day was generated during this period, while only 10% added VMT was generated during the AM peak. But the average added VMT during the PM peak was 0.4 miles per trip, while the average added VMT during the AM peak was 0.7 miles per trip because the origins of agents were widely distributed during the AM peak.

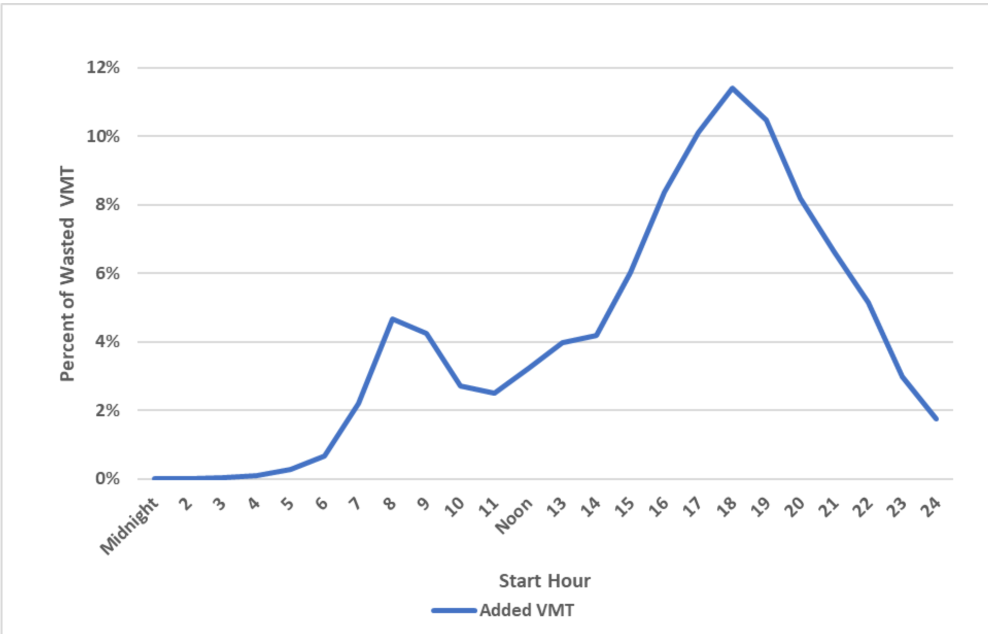


Figure 4.8: Distribution of Added VMT with Trip Start Time (using 1-hour bins, 7 travelers per SAV per day).

SCENARIOS WITH CURB PARKING RESTRICTED

In the original MATSim simulation, all modes will stay at their positions after requests are completed, but they are more likely to park on curb instead of parking lots in order to avoid extra congestion. They also can be activated when there are requests in their proximity and return to the simulation network. This process ignores the extra VMT and eVMT in real situations when SAVs need to drive themselves to nearby parking lots after they drop off agents. It may also have an influence over passenger wait times and SAV operation times. Parking lots were built in this study to analyze the impact of parking on the SAV simulation. As discussed above, different number of parking lots were used for scenarios with the 7 counties and the Twin Cities, since areas of simulation can significantly affect the available parking lots. For the aforementioned regions, 106 and 28 parking lots were created respectively, and the parking capacity were calculated by using eq. (2-2). Although the Twin Cities area is only 1/26th of the area of the 7 counties, the Twin Cities region contributes about 30% of the total trips. This is because the Twin Cities are 2 large cities: Minneapolis, the most populous city in the state, and Saint Paul, the state capital. It is obvious that the Twin Cities should have more parking lots to balance the SAVs. Figure 4.6 and 4.7 show the parking lots generation across the 7 counties and Twin Cities. The strategy of parking lots generation used here is relative to CBD areas and links with most origins and destinations. SAVs searched for the nearest parking lots if the distance between the SAVs and nearest parking lots was less than 0.5 miles.

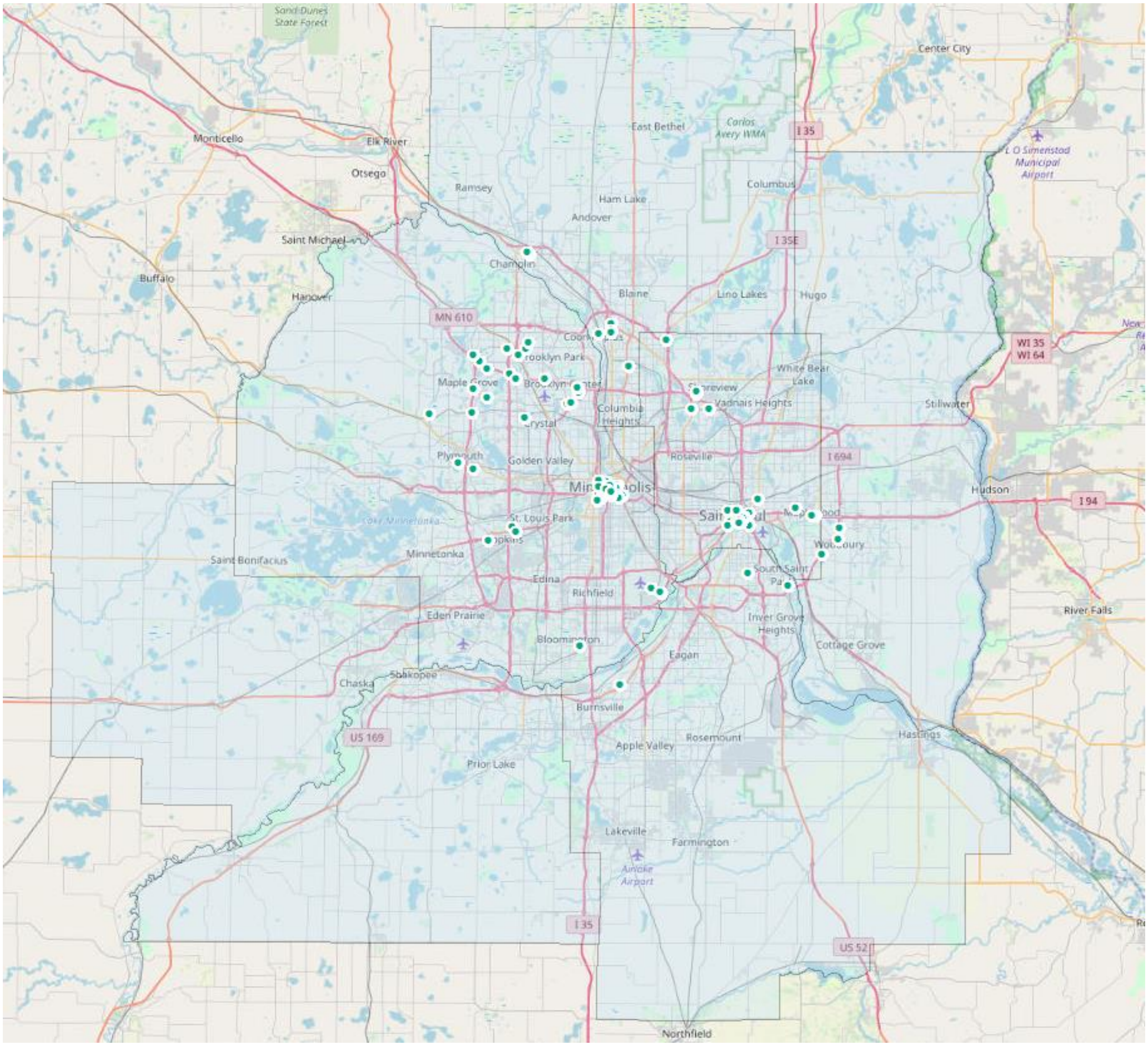


Figure 4.9: SAV Parking Lot Locations across the 7 Counties.

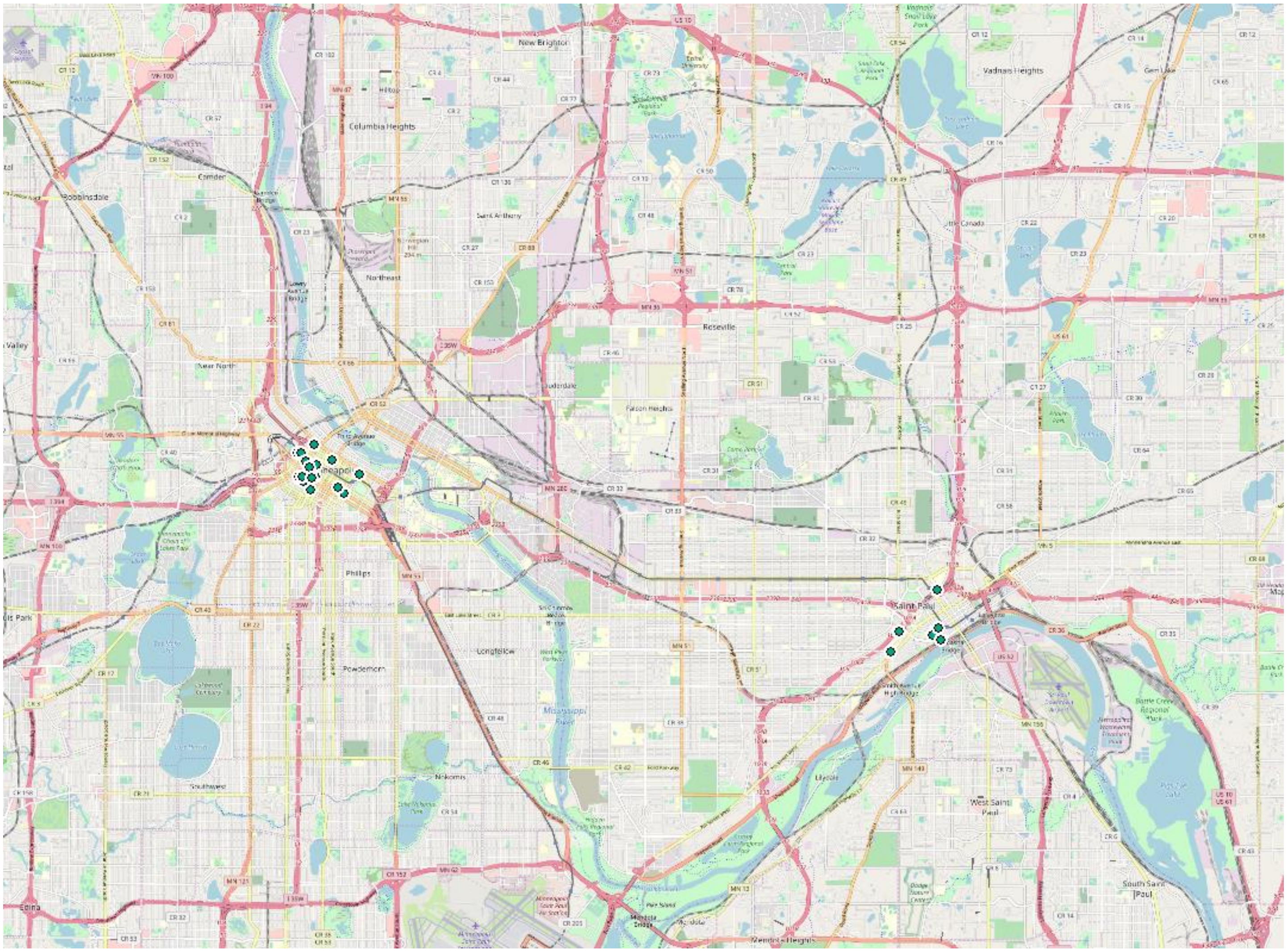


Figure 4.10: SAV Parking Lot Locations across the Twin Cities.

	Parking	Travelers per SAV per day	VMT per SAV per day per day	Empty VMT (%)	SAV Run Time per Day	%Trips as DRS	Trips per Day per SAV per day	AVO	Avg Wait Time (min.)
7-counties	Curb Parking Allowed	7	254 mi	14.5%	9.6	20.3%	28	1.23	4.6
	Curb Parking Constrained	7	272	20.1	10.1	19.6	28	1.24	5.2
Twin Cities	Curb Parking Allowed	7	156	10.0	4.6	25.3	22	1.32	3.6
	Curb Parking Constrained	7	170	18.2	5.8	24.9	22	1.32	4.0

Table 4.2: SAV Performance Comparison of Curb Parking Allowed and Curb Parking Constrained.

Table 4.2 showed the comparison of SAVs performances between scenarios with curb parking permitted everywhere and scenarios with curb parking restricted. The comparison of the parking strategies was based on 7 travelers per SAV per day after testing. For 7 counties, curb parking restricted implemented during simulation generated 8% more VMT on average, since SAVs always headed to the nearest parking lots after they dropped off the last passenger. According to the simulation results, the scenario with curb parking restricted increased, on average, 7% of the eVMT as compared to scenarios with curb parking permitted everywhere. Since those parking lots were created by links with the most origins and destinations, the parking trips for SAVs can be considered a relocation optimization. Such SAVs kept moving to parking lots with more trip densities, which entails that it may have more opportunities to respond to requests and may reduce the average wait time. However, since SAVs cannot respond to requests during parking trips, the average wait time was actually higher than the average wait time for the scenarios with curb parking permitted everywhere. Due to the times for parking, the SAV operation time in 24 hours increased by 15%. The number of DRS trips reduced by 5% in the scenario with curb parking restricted because, during the parking trip, SAVs could not respond to any trips requests until they arrived at the parking lots. Some potential DRS trips were ignored in this simulation, which could be improved in future work. There is a slight increase in the value of AVO, which seems to have a conflict with the decreased number of DRS trips. This might be explained by the fact that the relocation process may bring more potential DRS trips for more than 2 passengers in SAVs. As discussed above, those missed trips require more time to be satisfied, so the average wait time in the scenario with scenarios with curb parking restricted increased by about 10% as compared to the average wait time in the scenario without parking lots.

Areas of simulated regions could influence the performance of parking lots implementation. For the Twin Cities scenario, the performance of parking lots implementation had improved to varying degrees, including 47% less average SAV runtime per day, 5% more DRS trips, 7% more AVO, and 23% less average wait time. It indicated that considering a small area may decline the impacts of parking lot implementation on SAV performances, since it will be easier for SAVs to finish their parking trips and be ready for passenger requests. However, the average parking VMT per vehicle increased as compared to that from the 7 counties scenarios, as the Twin Cities had more parking lot density than the 7 counties.

ENERGY AND EMISSIONS ANALYSIS

An energy and emission analysis is warranted to determine the initial estimates for feasibility and consequences for the environment. Energy and emission coefficients were taken from a report on emission factors for greenhouse gas inventories (EPA, 2014) and Chester and Horvath's (2009) conventional gasoline vehicle inventory estimates, as shown in Table 4.3. The factors of emission species evaluated here are sulfur dioxide (SO_2), carbon monoxide (CO), oxides of nitrogen(NO_x), volatile organic compounds (VOC), particulate matter that is 10 micrometers or less in diameter (PM10), and greenhouse gases, (GHG) including carbon dioxide (CO_2) and methane (CH_4). The coefficients used here do not include pickup trucks and SUVs. This analysis assumed that SAV operation has no influence on trip demand, although people may make more and longer trips using SAVs if generalized costs fall in comparison with conventional vehicle travel. This analysis also included the energy and emissions of SAEVs if SAEVs operated in the MSP region instead of SAVs. But SAEVs were assumed to be charged at home and only once per day. Except GHG, all the coefficients of factors are based on vehicle operation (in use), startup emissions, manufacture, maintenance, and vehicle parking. Three specific vehicle kinds, namely internal combustion engine (ICE) vehicle, hybrid electric vehicle (HEV), and battery electric vehicle (BEV), were used here to represent sedans, SAVs, and SAEVs. Their miles per gallon (MPG) ratings, taken from EPA (2018), are shown in Table 4.4.

Energy and Emissions Species	Running Emissions per mile	Startup Emissions	Manufacture	Maintenance	Parking
Energy use (kj/mi)	4,800	0	550	210	79
SO_2(mg/mi)	21	0	110	45	19
CO(mg/mi)	11,000	7300	560	180	28
NO_x(mg/mi)	850	170	110	41	34
VOC(mg/mi)	310	350	110	52	27
PM10(mg/mi)	110	0	30	0	14
CO_2(mg/mi)	357,000				
CH_4(mg/mi)	173				

Table 4.3: Energy and Emission Assumptions for Conventional Gasoline Vehicle.

	ICE vehicle	SAV	SAEV
Type	Ford Focus	Toyota Prius (HEV)	Chevrolet Bolt (BEV)
MPG (Comb.)	31	52	106

Table 4.4: Fuel Economy of Three Vehicle Types.

From Kang and Recker (2009), this study assumed 1 hour for engines to cooldown. In total, 68% of U.S. vehicle trips (with internal combustion engines) are cold starts. From MSP simulation results, 10% are considered cold starts trips for the 7-county region and 7% are considered cold starts trips for the Twin Cities. In order to make comparisons intuitively, all the analyses were based on PMT. The average PMT of passenger by an ICE vehicle across the U.S. was taken from NHTS 2017. Using MSP simulation results, a comparison of the energy and emission for different vehicles are presented in Figure 4.11. Compared to an ICE vehicle, HEV SAVs could save 20% energy and reduce 30% of emissions, while BEV SAVs could decrease 64% of energy use and 68% of emissions. The costs of the emissions were also analyzed for advanced evaluation. In Table 4.5, Nichols et al. (2015) studied the emission species of battery electric vehicles and gasoline-powered vehicles. The authors considered emissions from powerplants to generate electricity (used by BEVs), while emissions released from the tailpipe of gasoline-powered vehicles are different. Figure 4.12 indicated that HEV SAVs and BEV SAVs may save 31% and 25% of emission costs respectively. However, there are many nuances in the emissions costs of BEVs (Nichols et al., 2015). The resources used for the various kinds of power plants are different; for example, thermal power plants may generate more emissions than wind or solar power plants. The emissions of SAEVs should be carefully studied in further studies. The results indicate that energy use and emissions per HEV SAV per day and BEV SAV were significantly less than those of conventional gasoline vehicles, since HEV SAVs and

BEV SAVs satisfy more demands. They, however, generated eVMT. The costs savings are not as much as the energy use and emissions savings, since the electricity generated in power plants may create more emissions.

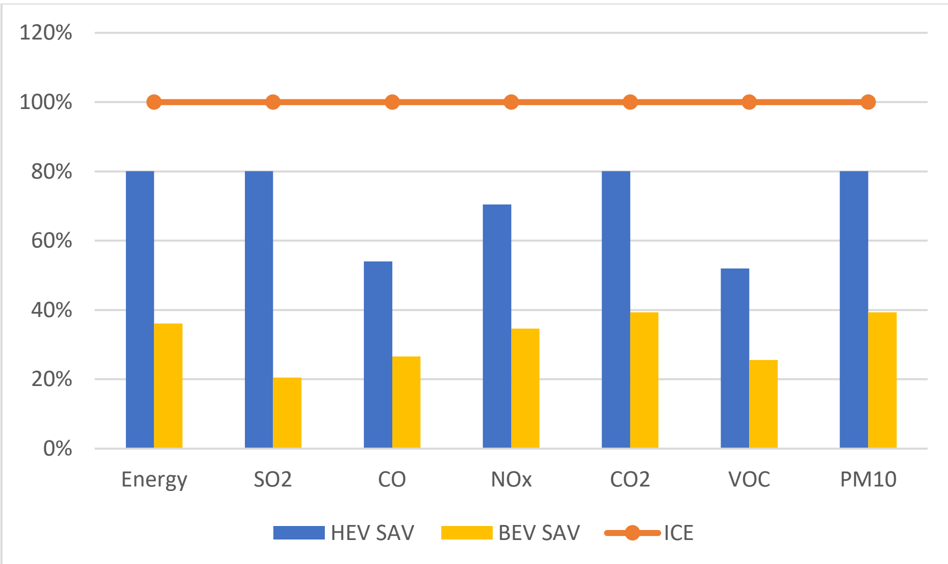


Figure 4.11: Energy and Emissions Per PMT, by Vehicle Type.

Emission Species	Battery Electric Vehicles	Gasoline-Powered Vehicles
	Costs per Vehicle-Mile	Costs per Vehicle-Mile
CO2	\$0.0016	\$0.0079
CO	\$0.0001	\$0.0008
CH4	\$0.0015	\$0.0001
NOx	\$0.0003	\$0.0005
SO2	\$0.0094	\$0.0002
VOC	\$0	\$0.0004
PM10	\$0.0012	\$0.0097
Total Costs	\$0.0141/VMT	\$0.0196/VMT

Table 4.5: Emission Costs of Battery Electric Vehicles and Gasoline-Powered Vehicles.

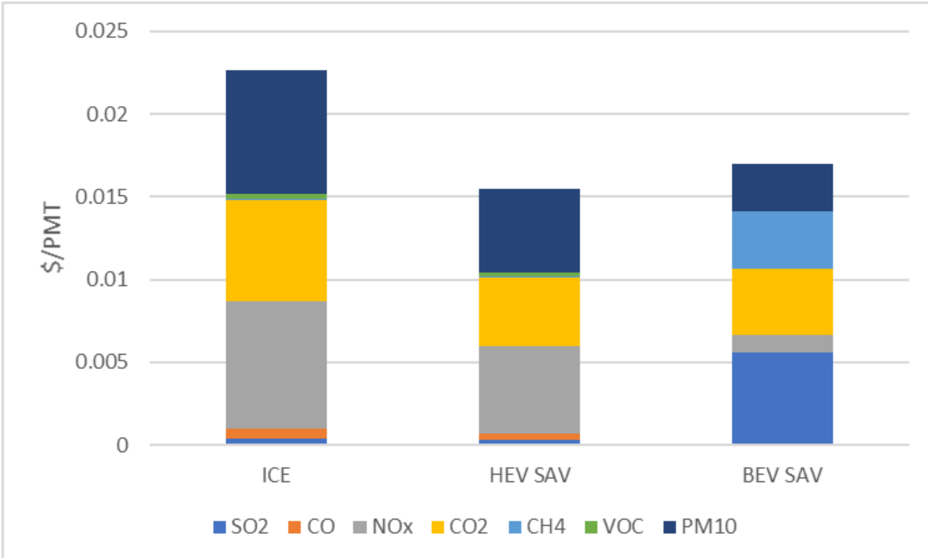


Figure 4.12: Emissions Costs Per PMT, by Vehicle Type.

Chapter 5: Conclusions

This work simulated and then evaluated the performance of an SAV fleet serving requests across the MSP region. The work uses MATSim code and compares the SAV fleet operations for different levels of trip demand and geofenced regions. Significant operational differences were found for different SAV fleet sizes (in terms of SAVs per traveler) serving different densities of demand (i.e., different percentage shares of all trips), with and without DRS enabled. With an average of 7 travelers per SAV per day across the region's 7 counties, vehicles served an average of 28 person trips per day with an average wait time of less than 5 minutes. Among all 22 simulation scenarios, eVMT averaged 7.2% to 25.2% of the SAV's fleet total VMT, with each SAV working 4 to 18 hours per day, with the DRS scenario with 5 travelers per SAV per day in Twin Cities and the no DRS scenario with 2% of total trip and 15 travelers per SAV per day in 7 counties, resulting in the most use & least use hours per day per SAV. Using the same fleet size and demand levels but allowing for DRS among strangers whose trips have meaningful overlap (in terms of routes or locations traveled and departure times), the same SAV fleets' average response times fell by 10% (from an average of 5 minutes to 4.5 minutes, for example).

This work also finds that SAVs may perform better in regions with a high population density and trip density with shorter trip lengths (i.e. 19% shorter trip lengths, on average) rather than a large region containing many suburban and rural areas. Relative to the large, 7-county service area, the Twin Cities-only geofenced fleet achieved, on average, 25% more DRS trips, and 19% shorter (average) wait times.

This study also evaluated variations in SAV VMT and eVMT values across the 7-county region's 2485 TAZs. Most SAV VMT and eVMT are generated on freeways and highways across the MSP region, especially the highway around the Twin Cities (of

Minneapolis and Saint Paul). eVMT distribution is similar to VMT distribution. The TAZs with the most VMT were scattered around the downtown areas of the Twin Cities, since the two cities' central business districts (CBDs) have the highest trip-end densities. SAVs' eVMT was mainly distributed in the areas that had a relatively low trip-end density and quite dispersed geographically while high-VMT TAZs were scattered around the downtown areas of the Twin Cities. 81% and 84% of TAZs with less than 6 minutes wait times are widely distributed during AM peak and PM peak. The study shows uniform wait times across the region and suggests residents of this region could get similar SAVs service level everywhere.

To avoid congesting busy streets across the region with SAVs idling in between drop-offs and pickup calls, this study created scenarios with parking lots near busy streets (i.e., links with at least 400 trip origins and destinations per curb per day) to compare to results for scenarios where curb parking is permitted everywhere. Scenarios that required off-street parking lot use for SAVs ending trips on busy links (roughly 0.02% of all street-miles in the MSP network) generated 8% more VMT and 7% more "busy" (SAV in-use) time on average, along with 5% fewer DRS (ride-sharing) trips for those scenarios. So, these curb-use constraints had a noticeable effect on operations. For the Twin Cities geofenced scenario (with just 273 square miles of SAV service area), compared to 7 counties scenario with curb parking restricted. parking lot use to protect busy streets actually resulted in 5% more DRS trips, 7% higher AVO values, and 23% less wait time, on average.

This study also analyzed the energy and emissions of hybrid-drivetrain (HEV) SAV fleets as compared to full-electric (BEV) SAVs and to conventional (ICE) drivetrain vehicles in the U.S., using the NHTS 2017 dataset. The BEV SAVs were assumed to be charged once per day, when the BEV SAVs are not in use, which is probably too optimistic for all SAVs in the fleet, given the long-distance trip making for some SAVs each day, and the high cost of larger batteries (for longer ranges). As compared to ICE vehicles, HEV SAVs are estimated to reduce the fleet's energy demands by 21% and 20% - 53% of different emission species, while BEV SAVs save 64% in energy use and 60% - 80% of different emission species. Thus, HEV SAVs and BEV SAVs may save 31% and 25% of emission costs.

Limitations of this study include the absence of external trips and commercial vehicle trips (i.e. about 16% of traffic), which contribute to VMT and congestion. Another limitation is that SAVs headed to parking lots go "offline" for a bit so that they cannot respond to any agents' requests until they finish parking themselves, thus resulting in overly conservative or pessimistic estimates of how SAV fleets facing curb-parking restrictions would actually fare. It also would be very useful for the simulations to equilibrate new destination and mode choices endogenously, when choosing departure times and routes, and to sample all travelers rather than subsets for the larger (county-wide and region-wide) services areas; but such behaviors complicated and/or slowed down the code too much to be used here (maxing out the UT Austin supercomputers' 48-hour run-time windows permitted). More optimization techniques can be used for vehicle assignments to travelers, fleet sizing, proactive SAV relocations, peak-hour SAV pricing, congestion pricing of all trips on congested links, and so forth.

Regardless, this study's results should prove helpful in anticipating future fleet operations across regions in the U.S. and elsewhere, enabling better decision-making by SAV fleet managers, regional policymakers, and the public at large. The metrics documented here can serve as a meaningful reference for decision-making during SAV implementation by local and federal authorities.

Appendices

Appendix A: Python Code

Coordinates generation

```
import csv
import arcpy
import glob
import os

#Allow the model to overwrite files
arcpy.env.overwriteOutput = True


##def get_all_shp(dir):
##      os.chdir(dir)
##      fcs = []
##      for file in glob.glob("*.shp"):
##              fcs.append(os.path.join(dir, file))
##
##      return fcs


req_csv = r'C:\Users\17307\Downloads\shp_trans_anlys_zones_offical_curent\origZone.csv'
file_dir = r'C:\Users\17307\Downloads\shp_trans_anlys_zones_offical_curent'
taz_gdb = r'C:\data\result_taz'
save_dir = r'C:\Users\17307\Downloads\shp_trans_anlys_zones_offical_curent\yhn'
xls_saved_dir =
r'C:\Users\17307\Downloads\shp_trans_anlys_zones_offical_curent\o_results'
# Make csv to dictionary, so to look up for the number by taz name
csv_dict = {}
with open(req_csv, 'rb') as csvfile:

    csv_content = csv.reader(csvfile, delimiter=' ', quotechar='|')
    header = next(csvfile)

    for row in csv_content:
        taz_info = row[0].split(',')
        taz_name = taz_info[0]
        number = int(taz_info[1])
        csv_dict[taz_name] = number
arcpy.env.workspace = taz_gdb
fcs = arcpy.ListFeatureClasses()
# print(fcs[0])
# arcpy.CreateRandomPoints_management(save_dir, "yhn.shp", "", "0 0 250 250", 10, "",
"POINT")
```

```
i = 0
for fc in fcs:
    number = csv_dict[fc.strip('.shp').lower()]
    print(i)
    arcpy.CreateRandomPoints_management(out_path=save_dir,
                                        out_name=fc,
                                        constraining_feature_class=fc,
                                        constraining_extent="0 0 250 250",
                                        number_of_points_or_field= number,
                                        minimum_allowed_distance="0 Meters",
                                        create_multipoint_output="POINT",
                                        multipoint_size="0")
    i = i+1

print "Generate random points done!"

arcpy.env.workspace = save_dir
featureclasses = arcpy.ListFeatureClasses()
i=1
for fc in featureclasses:
    print (i)
    arcpy.AddXY_management(in_features=fc)

    xls_name = fc + '.xls'
    xls_dir = os.path.join(xls_saved_dir, xls_name)
    arcpy.TableToExcel_conversion(Input_Table=fc,
                                  Output_Excel_File=xls_dir,
                                  Use_field_alias_as_column_header="NAME",
                                  Use_domain_and_subtype_description="CODE")
    i=i+1
print "Export to xls done!"
```

APPENDIX B: MATLAB CODE

Generate specific schedule

```
Highway_AM=csvread('Highway_AM.csv');
Highway_Midday=csvread('Highway_Midday.csv');
Highway_PM=csvread('Highway_PM.csv');
Highway_Night=csvread('Highway_Night.csv');
load('fivepercent-origin-sort.mat')
fivepercentadjust=tripsin7county;
for i = 1:9159833
%     if     fivepercentadjust(i,9)     ~=4andandfivepercentadjust(i,9)
~=6andandfivepercentadjust(i,13)~=fivepercentadjust(i,14)
        fivepercentadjust(i,16)=fivepercentadjust(i,16)-1/4+1/2*rand();
      if fivepercentadjust(i,16)<0
          fivepercentadjust(i,16)=tripsin7county(i,16)+abs(-1/4+1/2*rand());
      end
      if fivepercentadjust(i,16)>24
          fivepercentadjust(i,16)=24;
      end
%    end
end
for i = 1:9159833
%          if     fivepercentadjust(i,9)     ~=4andandfivepercentadjust(i,9)
~=6andandfivepercentadjust(i,13)~=fivepercentadjust(i,14)
    e=tripsin7county(i,13)*3061;
    for j = e-3060:e
        if
Highway_AM(j,1)==tripsin7county(i,13)andandHighway_AM(j,2)==tripsin7county(i,14)
            if tripsin7county(i,16)>=6andand tripsin7county(i,16)<9
                fivepercentadjust(i,17)=fivepercentadjust(i,16)+Highway_AM(j,4)/60;
            end
            if tripsin7county(i,16)>=9andandtripsin7county(i,16)<15.5

fivepercentadjust(i,17)=fivepercentadjust(i,16)+Highway_Midday(j,4)/60;
            end
            if tripsin7county(i,16)>=15.5andandtripsin7county(i,16)<18.5
                fivepercentadjust(i,17)=fivepercentadjust(i,16)+Highway_PM(j,4)/60;
            end
            if tripsin7county(i,16)>=18.5andandtripsin7county(i,16)<=24
                fivepercentadjust(i,17)=fivepercentadjust(i,16)+Highway_Night(j,4)/60;
            end
            if tripsin7county(i,16)<6
                fivepercentadjust(i,17)=fivepercentadjust(i,16)+Highway_Night(j,4)/60;
            end
%              if n(i,17)>n(i+1,16)andandn(i,3)==n(i+1,3)
%                  n(i+1,16) = n(i,17);
%              end
        end
```

```
    %       end
        end
    end


    save('fivepercentadjust','fivepercentadjust')

    x = 0:1/60:24;
    q = zeros(1441,1);
    p = zeros(1441,1);
    r = zeros(1441,1);
    h=0;
    b=[];
    c=1:1:1119629;
    t=0
     for i = 1:1119629
        if (fivepercentadjust(i,17)-fivepercentadjust(i,16))*60<15
            t=t+1;
        end
    %if                    fivepercentadjust(i,9)                    ~=4andandfivepercentadjust(i,9)
~=6andandfivepercentadjust(i,13)~=fivepercentadjust(i,14)
            a = floor((fivepercentadjust(i,17)-fivepercentadjust(i,16))*60/1);
            b(i)=a;
            if a == 0
                a = floor((fivepercentadjust(i,17)-fivepercentadjust(i,16))*60/1)+1;
            end
            if ceil(fivepercentadjust(i,17)*60/1)+1<=1441

r(ceil(fivepercentadjust(i,17)*60/1)+1)=r(ceil(fivepercentadjust(i,17)*60/1)+1)+1;
            end
            f = floor(fivepercentadjust(i,16)*60/1);
            q(f+1)=q(f+1)+1;
            if ((fivepercentadjust(i,17)-fivepercentadjust(i,16))*60/1)>=1

                if ceil(fivepercentadjust(i,17)*60/1)+1<=1441
                    for
j=ceil(fivepercentadjust(i,16)*60/1)+1:floor(fivepercentadjust(i,17)*60/1)
                        if ((fivepercentadjust(i,17)-fivepercentadjust(i,16))*60/1)>1
                            p(j)=p(j)+1/((fivepercentadjust(i,17)-
fivepercentadjust(i,16))*60/1);
                        else
                             p(j)=p(j)+1;
                        end
                    end

p(floor(fivepercentadjust(i,16)*60/1)+1)=p(floor(fivepercentadjust(i,16)*60/1)+1)+(fivepercentadjust(
i,16)*60/1-floor(fivepercentadjust(i,16)*60/1))/((fivepercentadjust(i,17)-
fivepercentadjust(i,16))*60/1);
```

```
p(ceil(fivepercentadjust(i,17)*60/1))=p(ceil(fivepercentadjust(i,17)*60/1))+(fivepercentadjust(i,17)*6
0/1-floor(fivepercentadjust(i,17)*60/1))/((fivepercentadjust(i,17)-fivepercentadjust(i,16))*60/1);
                    else
                        for j=ceil(fivepercentadjust(i,16)*60/1)+1:1441
                            if ((fivepercentadjust(i,17)-fivepercentadjust(i,16))*60/1)>1
                                p(j)=p(j)+1/((fivepercentadjust(i,17)-
fivepercentadjust(i,16))*60/1);
                            else
                                p(j)=p(j)+1;
                            end
                        end

p(floor(fivepercentadjust(i,16)*60/1)+1)=p(floor(fivepercentadjust(i,16)*60/1)+1)+(fivepercentadjust(
i,16)*60/1-floor(fivepercentadjust(i,16)*60/1))/((fivepercentadjust(i,17)-
fivepercentadjust(i,16))*60/1);
                    end
                else
                    p(ceil(fivepercentadjust(i,17)*60/1))=p(ceil(fivepercentadjust(i,17)*60/1))+1;
                end

    %         end
     end
```

Convert trip table (csv files) to xml files

```
    % load('fivepercentadjust.mat')
    tempname = 'MSP_five_AMoD2';

    docNode = com.mathworks.xml.XMLUtils.createDocument('population');
    docRootNode = docNode.getDocumentElement;
    %docRootNode.setAttribute('attr_name','attr_value');

    docNode.createComment('
======================================================================= ');

    for i =1:234245
        if i ==1||fivepercentadjust(i,4) ~= fivepercentadjust(i-1,4)
                personElement = docNode.createElement('person');

                personElement.setAttribute('id',num2str(fivepercentadjust(i,4)))
                docRootNode.appendChild(personElement);

                    planNode = docNode.createElement('plan');
                    planNode.setAttribute('selected','yes')
                    personElement.appendChild(planNode);
        end
        if i ==1||fivepercentadjust(i,4) ~= fivepercentadjust(i-1,4)||fivepercentadjust(i,4) ~=
fivepercentadjust(i+1,4)
```

```
                    if i ==1||fivepercentadjust(i,4) ~= fivepercentadjust(i-1,4)
                            activityNode = docNode.createElement('act');
                            activityNode.setAttribute('type','home');
                            activityNode.setAttribute('facility',num2str(fivepercentadjust(i,24)));
                            activityNode.setAttribute('x',num2str(fivepercentadjust(i,19)));
                            activityNode.setAttribute('y',num2str(fivepercentadjust(i,20)));

activityNode.setAttribute('end_time',datestr(fivepercentadjust(i,16)*3600*1.157407407407407/10000
0,'HH:MM:SS'));
                            planNode.appendChild(activityNode);

                            legNode = docNode.createElement('leg');
                        if
fivepercentadjust(i,18)==1||fivepercentadjust(i,18)==2||fivepercentadjust(i,18)==3||fivepercentadjust(i,
18)==5
                            if rand()<=1
                                legNode.setAttribute('mode','car');
                            else
                                legNode.setAttribute('mode','car');
                            end

legNode.setAttribute('dep_time',datestr(fivepercentadjust(i,16)*3600*1.157407407407407/100000,'H
H:MM:SS'))
                            planNode.appendChild(legNode);
                        end
                        if fivepercentadjust(i,18)==4||fivepercentadjust(i,18)==6
                            legNode.setAttribute('mode','walk');

legNode.setAttribute('dep_time',datestr(fivepercentadjust(i,16)*3600*1.157407407407407/100000,'H
H:MM:SS'))
                            planNode.appendChild(legNode);
                        end
                        if fivepercentadjust(i,18)==7||fivepercentadjust(i,18)==8
                            if rand()<=0.6
                                legNode.setAttribute('mode','pt');
                            else
                                legNode.setAttribute('mode','pt');
                            end

legNode.setAttribute('dep_time',datestr(fivepercentadjust(i,16)*3600*1.157407407407407/100000,'H
H:MM:SS'))
                            planNode.appendChild(legNode);
                        end
                            activityNode = docNode.createElement('act');
                        if fivepercentadjust(i,12) ==2||fivepercentadjust(i,12) ==514
                            activityNode.setAttribute('type','work');
                            activityNode.setAttribute('facility',num2str(fivepercentadjust(i,25)));
                        end
```

```
if fivepercentadjust(i,12) ==32||fivepercentadjust(i,12) ==16||fivepercentadjust(i,12) ==528||fivepercentadjust(i,12) ==544||fivepercentadjust(i,12) ==272||fivepercentadjust(i,12) ==288||fivepercentadjust(i,12) ==1040||fivepercentadjust(i,12) ==1056
    activityNode.setAttribute('type','shop');
    activityNode.setAttribute('facility',num2str(fivepercentadjust(i,25)));
end
if fivepercentadjust(i,12) ==1||fivepercentadjust(i,12) ==4||fivepercentadjust(i,12) ==129
    activityNode.setAttribute('type','education');

activityNode.setAttribute('facility',num2str(fivepercentadjust(i,25)));
end
if fivepercentadjust(i,12) ==8||fivepercentadjust(i,12) ==64||fivepercentadjust(i,12) ==128||fivepercentadjust(i,12) ==264||fivepercentadjust(i,12) ==320||fivepercentadjust(i,12) ==520||fivepercentadjust(i,12) ==544||fivepercentadjust(i,12) ==576||fivepercentadjust(i,12) ==640||fivepercentadjust(i,12) ==1032||fivepercentadjust(i,12) ==1088||fivepercentadjust(i,12) ==1152
    activityNode.setAttribute('type','leisure');

activityNode.setAttribute('facility',num2str(fivepercentadjust(i,25)));
end
    activityNode.setAttribute('x',num2str(fivepercentadjust(i,21)));
    activityNode.setAttribute('y',num2str(fivepercentadjust(i,22)));

activityNode.setAttribute('end_time',datestr(fivepercentadjust(i+1,17)*3600*1.157407407407407/100000,'HH:MM:SS'));
    planNode.appendChild(activityNode);

    legNode = docNode.createElement('leg');
if
fivepercentadjust(i,18)==1||fivepercentadjust(i,18)==2||fivepercentadjust(i,18)==3||fivepercentadjust(i,18)==5
    if rand()<=0.6
        legNode.setAttribute('mode','car');
    else
        legNode.setAttribute('mode','car');
    end

legNode.setAttribute('dep_time',datestr(fivepercentadjust(i+1,17)*3600*1.157407407407407/100000,'HH:MM:SS'))
    planNode.appendChild(legNode);
end
if fivepercentadjust(i,18)==4||fivepercentadjust(i,18)==6
    legNode.setAttribute('mode','walk');

legNode.setAttribute('dep_time',datestr(fivepercentadjust(i+1,17)*3600*1.157407407407407/100000,'HH:MM:SS'))
    planNode.appendChild(legNode);
end
```

```
                    if fivepercentadjust(i,18)==7||fivepercentadjust(i,18)==8
                        if rand()<=0.6
                            legNode.setAttribute('mode','pt');
                        else
                            legNode.setAttribute('mode','pt');
                        end

legNode.setAttribute('dep_time',datestr(fivepercentadjust(i+1,17)*3600*1.157407407407407/100000,'
HH:MM:SS'))
                        planNode.appendChild(legNode);
                    end
            end
            if fivepercentadjust(i,4) ~= fivepercentadjust(i+1,4)
                        activityNode = docNode.createElement('act');
                        activityNode.setAttribute('type','home');
                        activityNode.setAttribute('facility',num2str(fivepercentadjust(i,25)));
                        activityNode.setAttribute('x',num2str(fivepercentadjust(i,21)));
                        activityNode.setAttribute('y',num2str(fivepercentadjust(i,22)));
                        planNode.appendChild(activityNode);
            end
        else
                        activityNode = docNode.createElement('act');
                    if fivepercentadjust(i,12) ==2048
                        activityNode.setAttribute('type','home');
                        activityNode.setAttribute('facility',num2str(fivepercentadjust(i,25)));
                    end
                    if fivepercentadjust(i,12) ==2||fivepercentadjust(i,12) ==514
                        activityNode.setAttribute('type','work');
                        activityNode.setAttribute('facility',num2str(fivepercentadjust(i,25)));
                    end
                    if          fivepercentadjust(i,12)          ==32||fivepercentadjust(i,12)
==16||fivepercentadjust(i,12)      ==528||fivepercentadjust(i,12)      ==544||fivepercentadjust(i,12)
==272||fivepercentadjust(i,12) ==288||fivepercentadjust(i,12) ==1040||fivepercentadjust(i,12) ==1056
                        activityNode.setAttribute('type','shop');
                        activityNode.setAttribute('facility',num2str(fivepercentadjust(i,25)));
                    end
                    if          fivepercentadjust(i,12)          ==1||fivepercentadjust(i,12)
==4||fivepercentadjust(i,12) ==129
                         activityNode.setAttribute('type','education');

activityNode.setAttribute('facility',num2str(fivepercentadjust(i,25)));
                    end
                    if          fivepercentadjust(i,12)          ==8||fivepercentadjust(i,12)
==64||fivepercentadjust(i,12)      ==128||fivepercentadjust(i,12)      ==264||fivepercentadjust(i,12)
==320||fivepercentadjust(i,12)      ==520||fivepercentadjust(i,12)      ==544||fivepercentadjust(i,12)
==576||fivepercentadjust(i,12)      ==640||fivepercentadjust(i,12)      ==1032||fivepercentadjust(i,12)
==1088||fivepercentadjust(i,12) ==1152
                         activityNode.setAttribute('type','leisure');
```

```
activityNode.setAttribute('facility',num2str(fivepercentadjust(i,25)));
                    end
                        activityNode.setAttribute('x',num2str(fivepercentadjust(i,21)));
                        activityNode.setAttribute('y',num2str(fivepercentadjust(i,22)));

activityNode.setAttribute('end_time',datestr(fivepercentadjust(i+1,17)*3600*1.157407407407407/100
000,'HH:MM:SS'));
                        planNode.appendChild(activityNode);

                        legNode = docNode.createElement('leg');
                    if
fivepercentadjust(i,18)==1||fivepercentadjust(i,18)==2||fivepercentadjust(i,18)==3||fivepercentadjust(i,
18)==5
                        if rand()<=0.6
                            legNode.setAttribute('mode','car');
                        else
                            legNode.setAttribute('mode','av');
                        end

legNode.setAttribute('dep_time',datestr(fivepercentadjust(i+1,17)*3600*1.157407407407407/100000,'
HH:MM:SS'))
                        planNode.appendChild(legNode);
                    end
                    if fivepercentadjust(i,18)==4||fivepercentadjust(i,18)==6
                        legNode.setAttribute('mode','walk');

legNode.setAttribute('dep_time',datestr(fivepercentadjust(i+1,17)*3600*1.157407407407407/100000,'
HH:MM:SS'))
                        planNode.appendChild(legNode);
                    end
                    if fivepercentadjust(i,18)==7||fivepercentadjust(i,18)==8
                        if rand()<=0.6
                            legNode.setAttribute('mode','pt');
                        else
                            legNode.setAttribute('mode','pt');
                        end

legNode.setAttribute('dep_time',datestr(fivepercentadjust(i+1,17)*3600*1.157407407407407/100000,'
HH:MM:SS'))
                        planNode.appendChild(legNode);
                    end
          end

      end

      xmlFileName = [tempname,'.xml'];
      xmlwrite(xmlFileName,docNode);
      type(xmlFileName);
```

Appendix C: Java Code

VMT and eVMT in spatial and temporal extraction

```
import java.util.List;
import java.util.*;

import org.apache.log4j.Logger;
import org.matsim.api.core.v01.Id;
import org.matsim.api.core.v01.Scenario;
import org.matsim.api.core.v01.population.Activity;
import org.matsim.api.core.v01.population.Person;
import org.matsim.api.core.v01.network.Link;
import org.matsim.core.config.Config;
import org.matsim.core.config.ConfigUtils;
import org.matsim.core.network.NetworkUtils;
import org.matsim.core.network.io.NetworkReaderMatsimV2;
import org.matsim.core.population.PersonUtils;
import org.matsim.core.population.PopulationUtils;
import org.matsim.core.population.io.PopulationReader;
import org.matsim.core.router.EmptyStageActivityTypes;
import org.matsim.core.router.TripStructureUtils;
import org.matsim.core.scenario.ScenarioUtils;
import org.matsim.api.core.v01.events.ActivityEndEvent;
import org.matsim.api.core.v01.events.ActivityStartEvent;
import org.matsim.api.core.v01.events.LinkEnterEvent;
import org.matsim.api.core.v01.events.PersonEntersVehicleEvent;
import org.matsim.api.core.v01.events.PersonMoneyEvent;
import org.matsim.api.core.v01.events.handler.ActivityEndEventHandler;
import org.matsim.api.core.v01.events.handler.ActivityStartEventHandler;
import org.matsim.api.core.v01.events.handler.LinkEnterEventHandler;
import org.matsim.api.core.v01.events.handler.PersonEntersVehicleEventHandler;
import org.matsim.api.core.v01.events.handler.PersonMoneyEventHandler;
import org.matsim.api.core.v01.network.Network;
import org.matsim.contrib.util.CSVLineBuilder;
import org.matsim.contrib.util.CompactCSVWriter;
import org.matsim.core.api.experimental.events.EventsManager;
import org.matsim.core.controler.OutputDirectoryHierarchy;
import org.matsim.core.controler.events.IterationEndsEvent;
import org.matsim.core.controler.events.ShutdownEvent;
import org.matsim.core.controler.listener.IterationEndsListener;
import org.matsim.core.controler.listener.ShutdownListener;
import org.matsim.core.events.EventsReaderXMLv1;
import org.matsim.core.events.EventsUtils;
import org.matsim.core.events.MatsimEventsReader;
import org.matsim.core.utils.io.IOUtils;
import com.google.inject.Inject;
import ch.ethz.matsim.av.config.AVConfig;
import ch.ethz.matsim.av.config.AVOperatorConfig;
```

```
import ch.ethz.matsim.av.config.AVPriceStructureConfig;
import ch.ethz.matsim.av.dispatcher.AVVehicleAssignmentEvent;
import ch.ethz.matsim.av.dispatcher.multi_od_heuristic.aggregation.AggregationEvent;
import ch.ethz.matsim.av.schedule.AVTransitEvent;
import playground.gkmurthy.drs_pricing.AVTransitEventHandler;
import playground.gkmurthy.drs_pricing.AVVehicleAssignmentEventHandler;
import playground.gkmurthy.drs_pricing.AggregationEventHandler;
import playground.gkmurthy.drs_pricing.DRSAnalyzer;

public class ExtractEventInfo {

	final static private Logger log = Logger.getLogger(FixPopulationForDRS.class);
	private static int it;



	public static void main(String[] args)
	{
		String event_file = "C:\\Users\\17307\\git\\amod\\0.events.xml\\output_events.xml";
		String network_file = "C:\\Users\\17307\\git\\amod\\preparedNetwork.xml";


		EventsManager events = EventsUtils.createEventsManager(ConfigUtils.createConfig());
		Scenario scenario = ScenarioUtils.createScenario(ConfigUtils.createConfig());
		new NetworkReaderMatsimV2(scenario.getNetwork()).readFile(network_file);

		DRSAnalyzer xxx = new DRSAnalyzer(events,scenario.getNetwork());
		events.addHandler(xxx);
		new MatsimEventsReader(events).readFile(event_file);
		System.out.println(xxx.AV_VMT_in_m);

		for(Link l : scenario.getNetwork().getLinks().values()) {

			l.setNumberOfLanes(0);

		}

		for(Link l : scenario.getNetwork().getLinks().values()){

			if (xxx.AV_VMT_in_m.containsKey(l.getId().toString())){
	l.setNumberOfLanes(xxx.AV_VMT_in_m.get(l.getId().toString()));
			}

		}
```

```
			NetworkUtils.writeNetwork(scenario.getNetwork(), 
"C:\\\\Users\\\\17307\\\\git\\\\amod\\\\preparedNetwork_VMT.xml");

			log.info("File written to disk.");
	/*		NetworkUtils.writeNetwork(scenario.getNetwork(), 
"C:\\\\Users\\\\17307\\\\git\\\\amod\\\\revised_preparedNetwork.xml");

			log.info("File written to disk.");*/
		}

	}
```

Parking lots generation

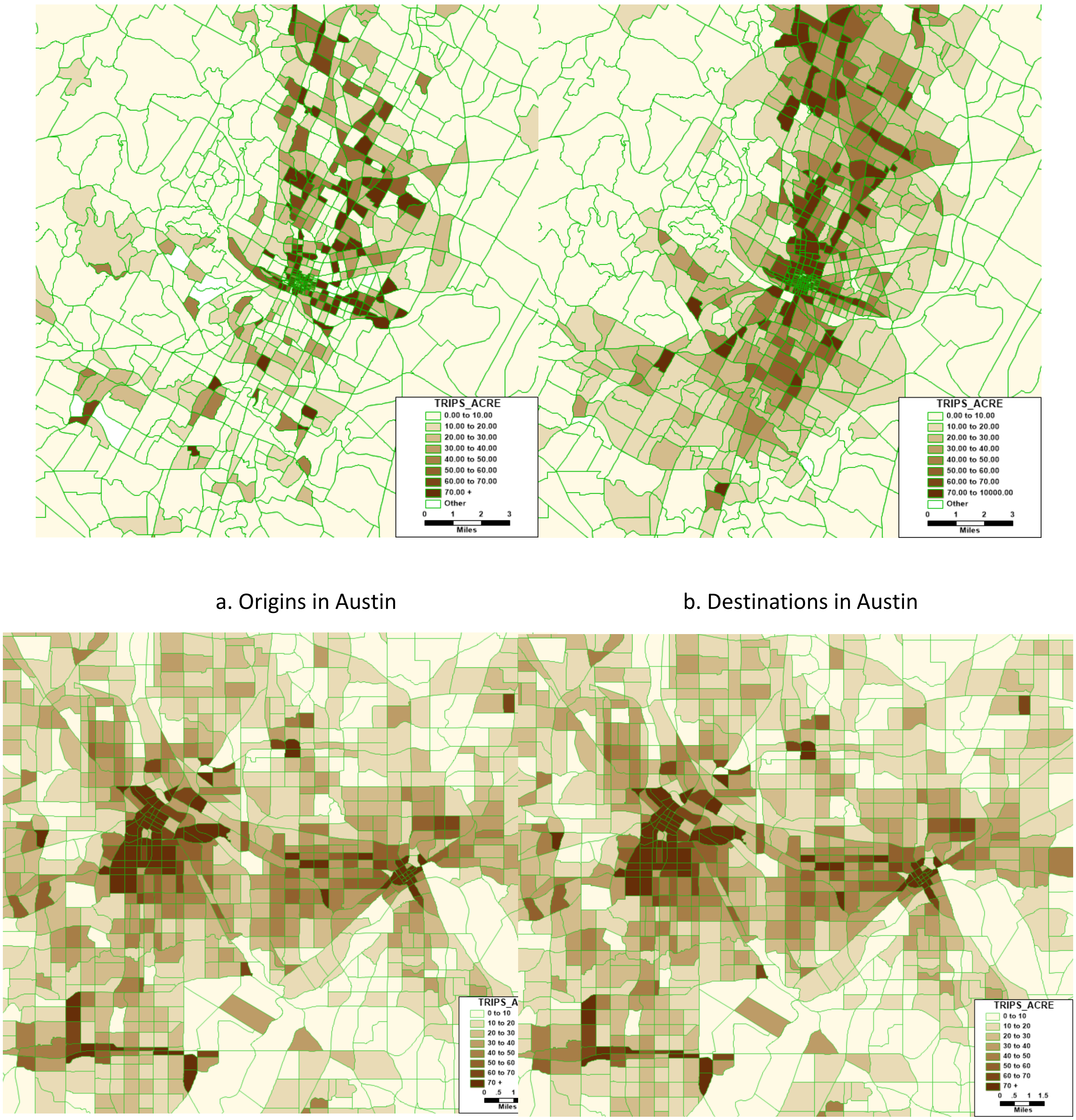
```
	import java.util.List;
	import java.util.*;

	import org.apache.log4j.Logger;
	import org.matsim.api.core.v01.Id;
	import org.matsim.api.core.v01.Scenario;
	import org.matsim.api.core.v01.population.Activity;
	import org.matsim.api.core.v01.population.Person;
	import org.matsim.api.core.v01.network.Link;
	import org.matsim.core.config.Config;
	import org.matsim.core.config.ConfigUtils;
	import org.matsim.core.network.NetworkUtils;
	import org.matsim.core.network.io.NetworkReaderMatsimV2;
	import org.matsim.core.population.PersonUtils;
	import org.matsim.core.population.PopulationUtils;
	import org.matsim.core.population.io.PopulationReader;
	import org.matsim.core.router.EmptyStageActivityTypes;
	import org.matsim.core.router.TripStructureUtils;
	import org.matsim.core.scenario.ScenarioUtils;

	public class AddParkingSpace {

		final static private Logger log = Logger.getLogger(FixPopulationForDRS.class);

		public static void main(String[] args)
		{
			String				plan_file				= 
"C:\\Users\\17307\\git\\amod\\revised_MSP_five_AMoD.xml.gz";
			String network_file = "C:\\Users\\17307\\git\\amod\\preparedNetwork.xml";
			int vehiclenumber=15000;
			int parkinglotsnumber=300;
			int[] array = new int[parkinglotsnumber];
			Map<Id<Link>, Integer> m1 = new HashMap<Id<Link>, Integer>();
```

```
		Scenario			scenario			=
ScenarioUtils.createScenario(ConfigUtils.createConfig());
		new PopulationReader(scenario).readFile(plan_file);
		new
NetworkReaderMatsimV2(scenario.getNetwork()).readFile(network_file);
		;

		for(Person p : scenario.getPopulation().getPersons().values())
		{
			List<Activity>			acts			=
TripStructureUtils.getActivities(p.getSelectedPlan(), EmptyStageActivityTypes.INSTANCE);

			for(Activity a : acts)
			{
				if (m1.get(a.getLinkId())==null){
				m1.put(a.getLinkId(), 1);
				}else{
					m1.put(a.getLinkId(),m1.get(a.getLinkId())+1);
				}
			}
		}
		m1 = sortByValueDescending(m1);


		for(Link l : scenario.getNetwork().getLinks().values())
		{
			int a=1;
			for (Map.Entry<Id<Link>, Integer> entry : m1.entrySet()) {
				if
(l.getId()==entry.getKey()andanda<=parkinglotsnumber){
					NetworkUtils.setSpatialAvCapacity(l	,	(long)
(vehiclenumber/parkinglotsnumber));
				}
				a=a+1;
			}
		}

		NetworkUtils.writeNetwork(scenario.getNetwork(),
"C:\\\\Users\\\\17307\\\\git\\\\amod\\\\revised_five_preparedNetwork.xml");

		log.info("File written to disk.");
	}


	public	static	<K,	V	extends	Comparable<?	super	V>>	Map<K,	V>
sortByValueDescending(Map<K, V> map)
	{
		List<Map.Entry<K, V>> list = new LinkedList<Map.Entry<K, V>>(map.entrySet());
```

```
        Collections.sort(list, new Comparator<Map.Entry<K, V>>()
        {
            @Override
            public int compare(Map.Entry<K, V> o1, Map.Entry<K, V> o2)
            {
                int compare = (o1.getValue()).compareTo(o2.getValue());
                return -compare;
            }
        });

        Map<K, V> result = new LinkedHashMap<K, V>();
        for (Map.Entry<K, V> entry : list) {
            result.put(entry.getKey(), entry.getValue());
        }
        return result;
    }
}
```

APPENDIX D: COMPARISONS BETWEEN AUSTIN AND MSP TRAVEL AND LAND USE PATTERNS

Figure 1A to Figure 3A shows the trip end density, population density and job density across the Austin and MSP. All those 3 pairs of figures are similar to each other. For Austin, most of trips, population and jobs are distributed along with Mopac and I-35. Besides that, main part of them are centralized in downtown area. On contrary, for MSP region, trips, population and jobs are distributed widely. Since MSP region has two important cities (Minneapolis and Saint Paul), these Twin Cities could cover a larger area thus people could live relatively further. But for people in Austin, they have small scope of activities around the downtown of Austin. That would be a reason why there is lower eVMT from Austin simulation. In a word, for Austin simulation, although 6-counties were simulated, the simulation actually contains a halo region with sparse trip distribution and a center region (Austin city area) with high trip density. More eVMT would be generated for those trips in halo region, but the number of these trips are extremely small compared to the number of trips in the center region. So simulation could produce a better results than the results from MSP region (not sure if Chicago also has the same situation as MSP region). In addition, according to Figure 4, since there is a wider river (than the river in Austin) across the MSP region, beside Minneapolis and Saint Paul, there is actually another center in southern part. It makes the trips, population and jobs across MSP disperse further.



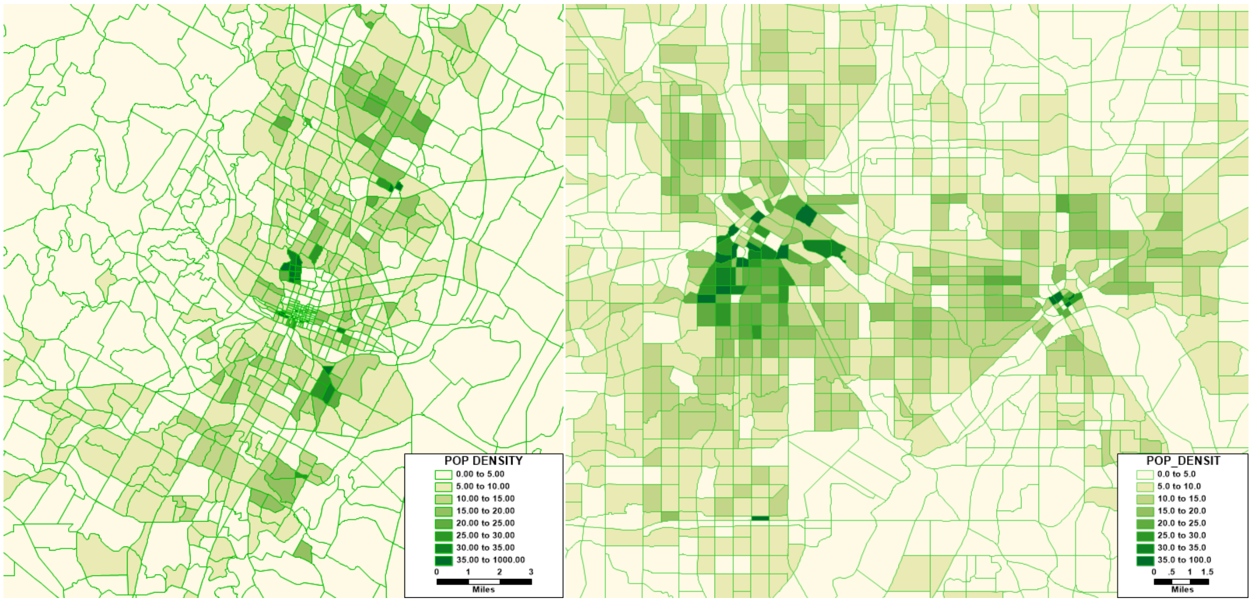
a. Origins in Austin

b. Destinations in Austin

c. Origins in MSP

d. Destinations in MSP

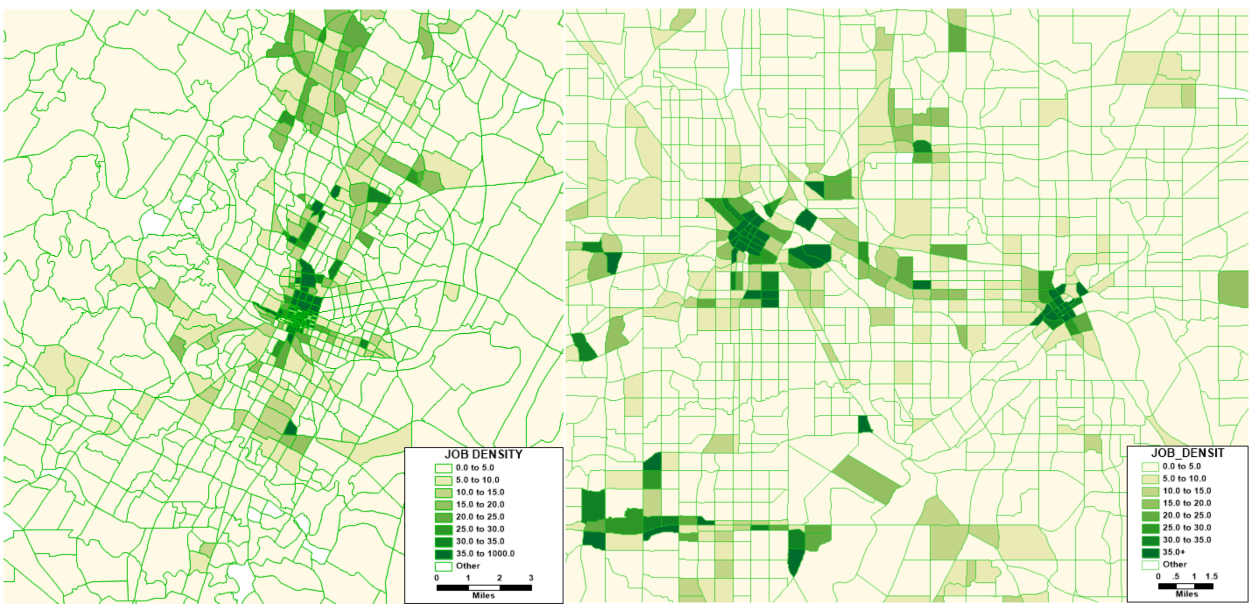
Figure D.1: Trip End Densities for Austin and MSP Regions (person-trip ends per acre).



a. Population Densities in MSP

b. Population Densities in MSP

Figure D.2: Population Densities for Austin and MSP Regions (residents per acre).



a. Job Densities in MSP

b. Job Densities in MSP

Figure D.3: Job Densities for Austin and MSP Regions (jobs per acre).

Glossary

AV	-	Autonomous Vehicle
AVO	-	Average Vehicle Occupancy
BEV		Battery Electric Vehicle
eVMT	-	Empty Vehicle-Miles Traveled
DRS	-	Dynamic Ridesharing
DTA	-	Dynamic Traffic Assignment
HEV	-	Hybrid Electric Vehicle (gasoline-electric)
SAEV	-	Shared Autonomous Electric Vehicle
SAV	-	Shared Autonomous Vehicle
TNC	-	Transportation Networking Company
VMT	-	Vehicle-Miles Traveled

References

Atasoy, Bilge; Ikeda, Takur; Ben-Akiva, Mosche E. (2014) Optimizing a Flexible Mobility on Demand System. *Transportation Research Record* 2536: 76-85.

Bösch, Patrick M., Ciari, Francesco, Axhausen, Kay W. (2016) Required Autonomous Vehicle Fleet Sizes to Serve Different Levels of Demand. *Transportation Research Record: Journal of the Transportation Research Board*, (2542), 111-119.

Burghout, Wilco; Rigole, Pierre-Jean; Andreasson, Ingmar (2015) Impacts of Shared Autonomous Taxis in a Metropolitan Area. Proceedings of the 94th Annual Meeting of the Transportation Research Board, Washington DC. Available at: https://www.researchgate.net/publication/298346251_Impacts_of_Shared_Autonomous_Taxis_in_a_Metropolitan_Area

Charypar, D., and Nagel, K. (2005). Q-Learning for Flexible Learning of Daily Activity Plans. *Transportation Research Record*, *1935*(1), 163-169.

Chester, M., and Horvath, A. (2009). Life-Cycle Energy and Emissions Inventories for Motorcycles, Diesel Automobiles, School Buses, Electric Buses, Chicago Rail, and New York City Rail. *University of California, Berkeley Report* No. UCB-ITS-VWP-2009-2

Chen, Donna and Kockelman, Kara and Hanna, Josiah (2016) Operations of a Shared, Autonomous, Electric Vehicle Fleet: Implications of Vehicle and Charging Infrastructure Decisions. *Transportation Research Part A* 94: 243-254.

Chen, Donna, and Kara Kockelman (2016) Management of a Shared, Autonomous Electric Vehicle Fleet: Implications of Pricing Schemes. *Transportation Research Record* No. 2572: 37-46.

Claudel, M., and Ratti, C., 2015. Full Speed Ahead: How the Driverless Car Could Transform Cities. Available at:

http://www.mckinsey.com/businessfunctions/sustainability-and-resource-productivity/our-insights/full-speed-ahead-howthe-driverless-car-could-transform-cities (accessed June 29, 2017), McKinsey and Company.

Ruch, C., Hörl, S., & Frazzoli, E. (2018). Amodeus, a Simulation-based Testbed for Autonomous Mobility-on-Demand Systems. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)* (pp. 3639-3644). IEEE.EPA (2014). Emission Factors for Greenhouse Gas Inventories. Stationary Combustion Emission Factors. US Environmental Protection Agency. Available at: https://www.epa.gov/sites/production/files/2018-03/documents/emission-factors_mar_2018_0.pdf

EPA (2019). The 2018 EPA Automotive Trends Report: Greenhouse Gas Emissions, Fuel Economy, and Technology since 1975. Available at: https://www.epa.gov/automotive-trends/download-automotive-trends-report

Eugensson, A., Brännström, M., Frasher, D., Rothoff, M., Solyom, S., and Robertsson, A. (2013). Environmental, Safety Legal and Societal Implications of Autonomous Driving Systems. Proceedings of the International Technical Conference on the Enhanced Safety of Vehicles (ESV), Seoul. Available at: https://www-esv.nhtsa.dot.gov/Proceedings/23/files/23ESV-000467.PDF

Fagnant, D. and Kockelman, K.M. (2014). The Travel and Environmental Implications of Shared Autonomous Vehicles, Using Agent-Based Model Scenarios. *Transportation Research Part C* 40: 1-13.

Fagnant, D. J., Kockelman, K. M., and Bansal, P. (2015). Operations of shared autonomous vehicle fleet for the Austin, Texas market. *Transportation Research Record: Journal of the Transportation Research Board* No. 2536: 98-106.

Fagnant, Daniel J.; Kara Kockelman (2016) Dynamic Ride-Sharing and Optimal Fleet Sizing for a System of Shared Autonomous Vehicles in Austin, Texas. *Transportation* 45: 1-16.

Fagnant, D. J., and Kockelman, K. M. (2018). Dynamic Ride-Sharing and Fleet Sizing for A System Of Shared Autonomous Vehicles in Austin, Texas. *Transportation*, 45(1), 143-158.

Federal Highway Administration (2017) National Household Travel Survey. U.S. Department of Transportation. Washington, D.C. Available at: https://nhts.ornl.gov/

Gurumurthy, K. M., and Kockelman, K. M. (2018). 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.

Gurumurthy, K. M., Kockelman, K. M., & Simoni, M. D. (2019). Benefits and Costs of Ride-Sharing in Shared Automated Vehicles across Austin, Texas: Opportunities for Congestion Pricing. *Transportation Research Record*, 0361198119850785.

Harper, C. D., Hendrickson, C. T., Mangones, S., and Samaras, C., 2016. Estimating potential increases in travel with autonomous vehicles for the non-driving, elderly and people with travel-restrictive medical conditions. *Transportation Research Part C: Emerging Technologies* 72, 1–9.

Hörl, S., (2017). Agent-based Simulation of Autonomous Taxi Services with Dynamic Demand Responses. *Procedia Computer Science*, 109: 899-904.

Horni, A., Nagel, K., and Axhausen, K. W. (Eds.). (2016). The Multi-Agent Transport Simulation Matsim. London: Ubiquity Press. Available at: https://oapen.org/search?identifier=613715

Jung, J., Jayakrishnan, R., and Park, J. Y. (2013). Design and Modeling of Real-Time Shared-Taxi Dispatch Algorithms. In Proc. *Transportation Research Board 92nd Annual Meeting*. Available at: https://pdfs.semanticscholar.org/65a1/9bf678e034a9987a5ba9b17c1561afa0c7b8.pdf

Kang, Jee E. and W. W. Recker (2009) An Activity-Based Assessment of the Potential Impacts of Plug-in Hybrid Electric Vehicles on Energy and Emissions using 1-Day Travel Data. *Transportation Research Part D*, 14 (8): 541-556.

Kockelman, K., Huang, Y., and Quarles, N. (2019). The Rise of Long-Distance Trips, in a World of Self-Driving Cars: Anticipating Trip Counts and Evolving Travel Patterns Across the Texas Triangle Megaregion. Available at: http://sites.utexas.edu/cm2/files/2019/04/Year2_Kockelman_AVs.pdf

Kornhauser, A. (2013) PRT Statewide Application: Conceptual Design of a Transit System 611 Capable of Serving Essentially All Daily Trips. *Urban Public Transportation Systems* 612: 357-368.

LaMondia, J. J., Fagnant, D. J., Qu, H., Barrett, J., and Kockelman, K. (2016). Shifts in Long-Distance Travel Mode Due To Automated Vehicles: Statewide Mode-Shift Simulation Experiment And Travel Survey Analysis. *Transportation Research Record, 2566*(1), 1-11.

Li, T., & Kockelman, K. M. (2018). Valuing the Safety Benefits of Connected and Automated Vehicle Technologies. Available at: http://www.caee.utexas.edu/prof/kockelman/public_html/CAV_Book2018.pdf

Liu, J., Kockelman, K. M., Boesch, P. M., and Ciari, F. (2017). Tracking a System of Shared Autonomous Vehicles across the Austin, Texas Network Using Agent-based Simulation. *Transportation*, 44(6), 1261-1278.

Litman, T., 2015. Autonomous Vehicle Implementation Predictions – Implications for Transport Planning. 2015 Transportation Research Board Annual Meeting, 15–3326, Washington.

Loeb, B. J. and Kockelman, Kara (2018). Shared Autonomous Electric Vehicle (SAEV) Operations across the Austin, Texas Region, with a Focus on Charging Infrastructure Provision and Cost Calculations. *Transportation Research Part C* 89: 222-233.

Loeb, Ben and Kockelman, Kara M. 2019. Fleet Performance and Cost Evaluation of a Shared Autonomous Electric Vehicle (SAEV) Fleet: A Case Study for Austin, Texas. *Transportation Research Part A* 121: 374-385.

Maciejewski, M., and Bischoff, J. (2017). Congestion Effects of Autonomous Taxi Fleets. *Transport*, 1-10.

Maciejewski, M., Bischoff, J., and Nagel, K. (2016). An Assignment-based Approach to Efficient Real-Time City-Scale Taxi Dispatching. *IEEE Intelligent Systems*, 31(1), 68-77.

Martinez, L., and Crist, P. (2015). Urban Mobility System Upgrade–How Shared Self-driving Cars Could Change City Traffic. Available at: https://www.itf-oecd.org/sites/default/files/docs/15cpb_self-drivingcars.pdf

Martinez, L. M., and Viegas, J. M. (2017). 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.

Nichols, B., Reiter M.S., and Kockelman K. (2015). Air Quality Impacts of Electric Vehicle Adoption in Texas. *Transportation Research Part D* 34: 208-218.

Perrine, Kenneth A., Kockelman, Kara M. and Huang Yantao. (2018). Anticipating Long-Distance Travel Shifts Due to Self-Driving Vehicles. *Transp Res Board*, 1-17.

Reiter, M. S., and Kockelman, K. M. (2017). Emissions and Exposure Costs of Electric Versus Conventional Vehicles: A Case Study in Texas. *International Journal of Sustainable Transportation*, *11*(7), 486-492.

Rodoulis S., 2014. The Impact of Autonomous Vehicles on Cities. JOURNEYS – Sharing Urban Transport Solutions, Issue 12, LTA Academy, Land Transport Authority, Singapore.

Simoni, M. D., Kockelman, K. M., Gurumurthy, K. M., and Bischoff, J. (2019). Congestion Pricing in a World of Self-Driving Vehicles: An Analysis of Different Strategies in Alternative Future Scenarios. *Transportation Research Part C: Emerging Technologies*, *98*, 167-185.

Spieser, K., Treleaven, K., Zhang, R., Frazzoli, E., Morton, D., and Pavone, M. (2014). Toward A Systematic Approach to the Design and Evaluation of Automated Mobility-on-Demand Systems: A Case Study in Singapore. *Road-Vehicle Automation* (pp. 229-245). Springer, Cham.

Tachet, R., Sagarra, O., Santi, P., Resta, G., Szell, M., Strogatz, S. H., and Ratti, C. (2017). Scaling Law of Urban Ride Sharing. *Scientific Seports*, 7, #42868.

Tesla Motors (2016c) Order a Tesla Model S. Available at: https://www.tesla.com/models/design

Technology Law and Policy Clinic (2015). Autonomous Vehicle Law Report and Recommendations to the ULC. University of Washington School of Law. Available at: https://www.law.washington.edu/ clinics/technology/Reports/AutonomousVehicle.pdf

Ticoll D., 2015. Driving Changes: Automated Vehicles in Toronto. Innovation Policy Lab, Munk School of Global Affairs, University of Toronto, Toronto. Available at: https://munkschool.utoronto.ca/ipl/files/2016/03/Driving-Changes-Ticoll-2015.pdf

Skinner, R., and Bidwell, N. (2016). Making Better Places: Autonomous Vehicles and Future Opportunities. *WSP-Parsons Brinckerhoff Engineering Services: London, UK*. Available at: http://www.wsp-pb.com/globaln/uk/wsppb-farrells-av-whitepaper.pdf

Zachariah, Jaison; Jingkang Gao, Alain Kornhauser, Talal Mufti (2014) Uncongested Mobility for All: A Proposal for an Area Wide Autonomous Taxi System in New Jersey. Proceedings of the 93rd Annual Meeting of the Transportation Research Board, Washington DC. Available at: https://orfe.princeton.edu/~alaink/SmartDrivingCars/Papers/ZachariahTRB_AutonomousTaxis.pdf