Eco-Mobility-on-Demand Service with Ride-Sharing

by

Xianan Huang

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Mechanical Engineering)
in The University of Michigan
2019

Doctoral Committee:

Professor Huei Peng, Chair
Dr. Joshua A. Auld, Argonne National Laboratory
Assistant Professor Alex Gorodetsky
Assistant Professor Xun Huan
Professor Henry Liu

Xianan Huang

xnhuang@umich.edu

ORCID iD: 0000-0002-1912-4295

ACKNOWLEDGMENTS

I would like to express my gratitude to my advisor Professor Huei Peng. His guidance helped me transition from an undergraduate student into an independent researcher. I am grateful for his light touch, which helped me to identify and develop a research direction. I'm indeed fortunate to have the opportunity to work with him. I would also like to thank my committee members, Dr. Joshua Auld, Professor Alex Gorodetsky, Professor Xun Huan, and Professor Henry Liu, for the constructive comments and suggestions.

I want to thank the Department of Energy for funding my research, and more importantly, giving me the chance to meet awesome people from Argonne National Laboratory such as Dr. Joshua Auld and Professor Vadim Sokolov. Collaborating with them truly opened my eyes to many interesting tools and topics in transportation engineering and data science.

I am also grateful to my labmates, Xiaowu, Tianyou, Ding, Ziheng, Steve, Yuxiao, Su-yang, Geunsob, Songan, Nauman, Boqi, Minghan, Xinpeng, Yuanxin, Shaobing, Pingping, and Yiqun. I am grateful for their help and assistance with the research. Especially, I would like to thank Qi Luo and Boqi Li for constructing the Ann Arbor road network from OpenStreetMap and converting it to SUMO format. I'd also like to thank my friends, who helped me enliven my life and helped me in both my academic and personal life.

Finally, my greatest appreciation goes to my parents, Jie Huang and Xiaojia Xia, as well as my grandfather, Chengren Xia, for their unconditional love and support through the process.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

BSFC	Brake Specific Fuel Consumption
CACC	Cooperative Adaptive Cruise Control
CAV	Connected Automated Vehicle
CRP	Chinese Restaurant Process
CVT	Continuously Variable Transmission
DP	Dynamic Programming
DPGMM	Dirichlet Process Gaussian Mixture Model
DSRC	Dedicated Short Range Communications
EAD	Eco-Approach/Departure
ELBO	Evidence Lower Bound
EM	Expectation-Maximization
GMM	Gaussian Mixture Model
GMR	Gaussian Mixture Regression
I2V	Infrastructure-to-Vehicle
ILP	Integer Linear Programming
KL Divergence	Kullback-Leibler Divergence
MARS	Multivariate Adaptive Regression Spline
MCMC	Markov Chain Monte Carlo
MDP	Markov Decision Process
MDS	Multi-Dimensional Scaling
MOD	Mobility-on-Demand
NHTSA	National Highway Traffic Safety Administration
NN	Neural Networks
RSE	Road Side Equipment
SAE	Society of Automotive Engineers

SAV	Shared Automated Vehicle
SPaT	Signal Phase and Timing
SPMD	Safety Pilot Model Deployment
SQP	Sequential Quadratic Programming
SVM	Support Vector Machines
TSP	Traveling Salesman Problem
VI	Variational Inference
VMT	Vehicle Miles Traveled
VRP	Vehicle Routing Problem

ABSTRACT

Connected Automated Vehicles (CAV) technologies are developing rapidly, and one of its more popular application is to provide mobility-on-demand (MOD) services. However, with CAVs on the road, the fuel consumption of surface transportation may increase significantly. Travel demands could increase due to more accessible travel provided by the flexible service compared with the current public transit system. Trips from current underserved population and mode shift from walking and public transit could also increase travel demands significantly. In this research, we explore opportunities for the fuel-saving of CAVs in an urban environment from different scales, including speed trajectory optimization at intersections, data-drive fuel consumption model and eco-routing algorithm development, and eco-MOD fleet assignment.

First, we proposed a speed trajectory optimization algorithm at signalized intersections. Although the optimal solution can be found through dynamic programming, the curse of dimensionality limits its computation speed and robustness. Thus, we propose the sequential approximation approach to solve a sequence of mixed integer optimization problems with quadratic objective and linear constraints. The number of integer states is the number of green windows of all traffic lights in the planning horizon, thus the number of integer variables is limited. The speed and acceleration constraints at intersections due to route choice are addressed using a barrier method. In this work, we limit the problem to a single intersection due to the route choice application and only consider free flow scenarios, but the algorithm can be extended to multiple intersections and congested scenarios where a leading vehicle is included as a constraint if an intersection driver model is available.

Next, we developed a fuel consumption model for route optimization. The mesoscopic fuel consumption model is developed through a data-driven approach considering the tradeoff between model complexity and accuracy. To develop the model,

a large quantity of naturalistic driving data is used. Since the selected dataset doesn't contain fuel consumption data, a microscopic fuel consumption simulator, Autonomie, is used to augment the information. Gaussian Mixture Regression is selected to build the model due to its ability to address nonlinearity. Instead of selected component number by cross-validation, we use the Bayesian formulation which models the indicator of components as a random variable which has Dirichlet distribution as prior. The model parameters are obtained through max-a-posterior inference from data, and the conditional expectation of fuel consumption on input variables can be obtained in closed form since the individual components follow the Gaussian distribution. The model is used to estimate fuel consumption cost for routing algorithm. In this part, we assume the traffic network is static.

Finally, the fuel consumption model and the eco-routing algorithm are integrated with the MOD fleet assignment. The MOD control framework models customers' travel time requirements are as constraints, thus provides flexibility for cost function design. At the current phase, we assume the traffic network is static and use offline calculated travel time and fuel consumption to assign the fleet. To rebalance the idling vehicles, we developed a traffic network partition algorithm which minimizing the expected travel time within each cluster. A Model Predictive Control (MPC) based algorithm is developed to match idling fleet distribution with the demand distribution. A traffic simulator using Simulation of Urban MObility (SUMO) and calibrated using data from the Safety Pilot Model Deployment (SPMD) database is used to evaluate the MOD system performance. It's argued from the literature that ride-sharing has the potential to reduce fuel consumption. However, this dissertation shows that if the objective function of fleet assignment is not designed properly, even if ride-sharing is allowed, the fleet fuel consumption could increase compared with the baseline where personal vehicles are used for travel.

CHAPTER 1
Introduction

1.1 Motivation

Connected Automated Vehicle (CAV) technologies have the potential to change the future of ground transportation significantly. CAVs can save fuel, reduce traffic accidents, ease congestion, and provide better mobility service to the elderly, physically challenged, and vision-challenged population [1]. The Society of Automotive Engineers (SAE) defined six levels of automated driving in the SAE J3016 Standard [2] as shown in Table 1.1.

Table 1.1 SAE 6 Levels of Automation Vehicles

Level	**Name**	**Execution of steering and acceleration/ deceleration**	**Monitoring of driving environment**	**Fallback performance of dynamic driving task**	**System capability (driving modes)**
Human driver monitors the driving environment					
0	No Automation	Human	Human	Human	N/A
1	Driver Assistance	Human/System	Human	Human	Some modes
2	Partial Automation	System	Human	Human	Some modes
Automated driving system monitors the driving environment					
3	Conditional Automation	System	System	Human	Some modes
4	High Automation	System	System	System	Some modes
5	Full Automation	System	System	System	All modes

The key distinction is between levels 3 and 4. A vehicle of levels 1-3 still requires a licensed driver to operate, while levels 4-5 vehicles allow driverless operations.

Currently, the industry is moving from level 1 and up, possibly to level 4 in the next few years, with Waymo started the autonomous mobility service in 2018 [3]. Since Electronic Stability Control (ESC) has been mandatory in the U.S. since 2012, and most vehicles are equipped with the conventional cruise control feature, most light-duty vehicles in the US can be said to be at least having some automated capabilities already.

Almost all major car companies have initiated researches and development programs for CAVs, and some new light-duty vehicles are equipped with automated driving functions such as Autopilot from Tesla [4], Supercruise from GM [5] and Pilot Assist from Volvo [6]. However, as the level of automation increases, robust perception and decision making require additional hardware such as LiDARs, Radars, high-resolution Cameras, and high-performance computers. Currently, one of the most popular sensors, LiDAR, costs $4,000 to $70,000 per unit [7]. IHS Automotive predicts that the self-driving technologies can lead to $7,000 to $10,000 increase in new vehicle price by 2025 [8]. The hardware and software need to be maintained routinely [9] to avoid the potentially fatal effects of system failure, which could further increase the operational cost and slow down the adoption of CAVs [10]. One potential solution is to use connected vehicle technologies [11] to reduce the necessity of high-performance sensors for individual vehicles. For example, instead of relying on advanced localization technologies such as Real-Time Kinematic (RTK) GPS, connected vehicles can exchange the relative location information with each other and the infrastructure to achieve better localization accuracy [12]. However, a high penetration ratio is required to achieve a reliable level of performance, which is likely to take a while [9]. Mobility-on-Demand (MOD) service using the shared automated vehicles (SAV) is proposed as a solution [13, 14] to make CAV technologies more accessible.

MOD service such as Uber and Lyft have brought significant changes, especially in densely populated urban areas. In 2015, the mobility service accounted for 4% of global mileage traveled, and by 2030, Morgan Stanley estimates that the number could reach 26% [15], as shown in Figure 1.1. Compared with the fixed-schedule and fixed-route public transit systems, it can provide a more flexible and convenient service. Compared with privately owned cars, the travel cost per mile is lower by sharing the cost with others having similar itineraries [16]. A recent study [17] found that 18,000 shared vehicles are enough

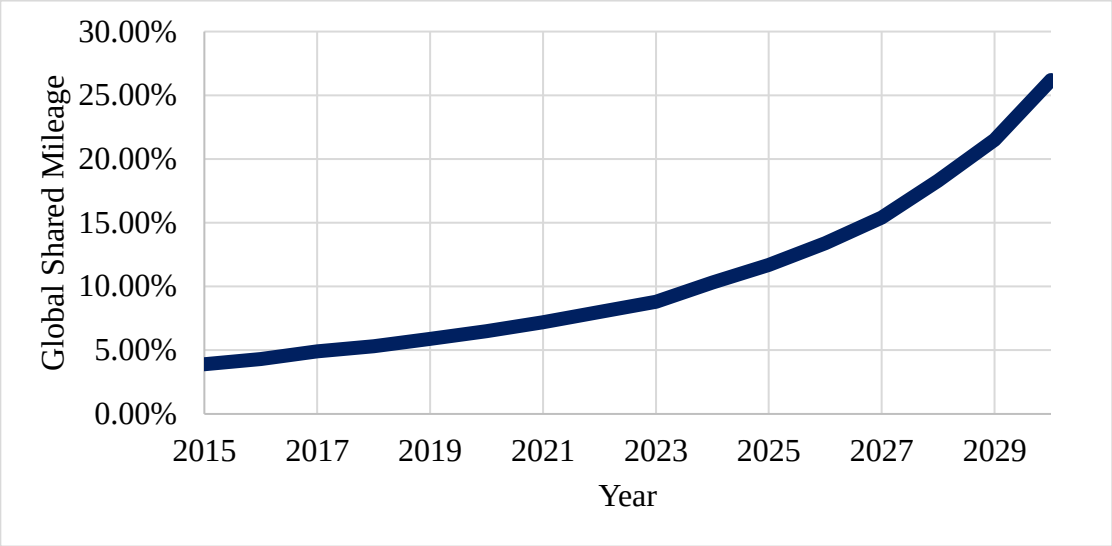


Figure 1.1 Global Shared Miles Forecast [15]

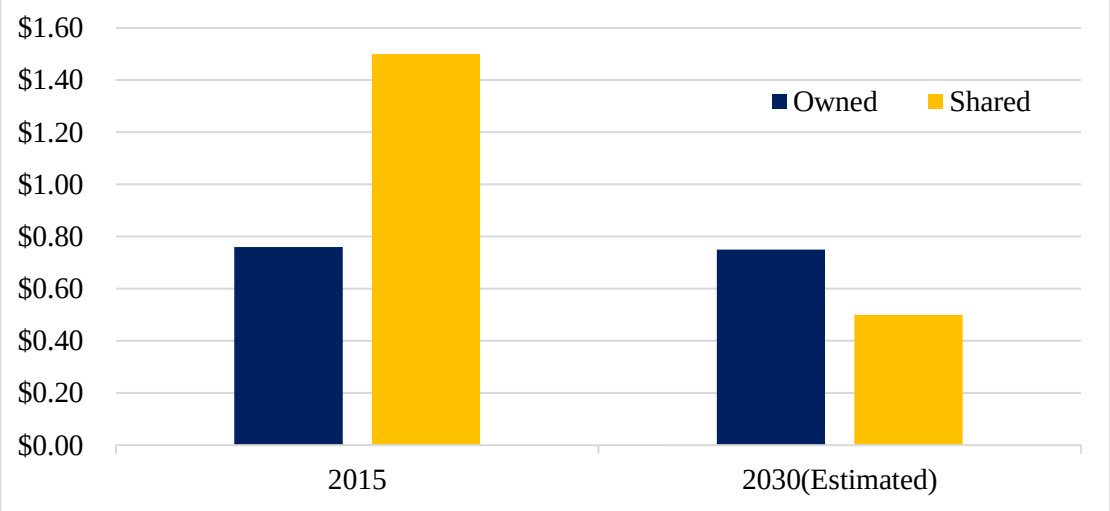


Figure 1.2 Cost per Mile: Shared v.s. Owned [15]

to serve 120,000 customers who use their cars less than 70 miles per day in Ann Arbor, which is 60% of all customers. Combined with highly automated vehicle features such as Cooperative Adaptive Cruise Control (CACC) and eco-routing as well as battery electric vehicles, [15] predicted that by 2030, the cost per mile for MOD service is 33% lower than personally owned non-autonomous vehicles. MOD service has the potential to change vehicle ownership and travel behavior dramatically. An analysis of the Car2go program [18] revealed that the shared-vehicle service could reduce vehicle ownership. Also, a nation-wide survey [19] found that the shared-vehicle service can reduce personal vehicle ownership by 49%.

The increasing market share of MOD service and cost reduction due to highly automated vehicles can change travel demand dramatically. On the one hand, vehicle travel could increase due to current non-drivers traveling, the empty vehicle travels for pick-up and drop-off, and reduced travel cost [18, 20–22]. On the other hand, the change in car ownership and reduction in activities such as hunting for parking could reduce vehicle travel. [23] predicted that the trips due to current non-drivers can contribute to an 11% increase in vehicle travel. Moreover, [9] predicted that with the current policy and vehicle pollution level, the emission could increase 10-30% and even more on major corridors. A recent study by several U.S. national labs [24] found that although the CAV technologies have the potential to reduce fuel consumption by 90%, the increased travel mileage can increase emission by 200%. As one of the most significant segments for energy consumption, ground transportation consumes 26.5% of the world energy in 2016 [25]. With increased travel demand due to automated vehicles and MOD service, fuel consumption needs to be considered by the fleet operators.

To mitigate the potential impact on emission and reduced operation cost for MOD fleet operator, researches [26–29] have explored opportunities to reduce fuel consumption and emission using the CAV technologies. In highway driving, the concept Eco Cooperative Adaptive Cruise Control (ECACC) is proposed. While the primary goal of traditional CACC is to maintain string-stability to improve safety and road capacity, the shorter inter-vehicle distance can reduce wind resistance, and thus the fuel consumption by more than 6% [30, 31]. Besides fuel savings through reduced drag, the concept of eco-driving is applied to the car-following scenario in ECACC by switching between two efficient engine operating points [32–34], showing that fuel consumption can be reduced by up to 8.9% [32]. The “pulse-and-glide” eco-driving strategy has also been applied to the mixed automated and human-driven vehicle platoon [35] and can reduce fuel consumption by 10%, but the smoothness of traffic flow can suffer. [34] designed a car-following strategy based on bounded stability to vehicle platoon, which achieved more than 20 % fuel saving without sacrificing string stability. To achieve the full potential of platooning, [36] developed a routing strategy for the truck fleet to maximize the probability of platoon formulation, resulting in 1.2% fuel reduction.

In city driving, where MOD service can have a transformative impact [37], stop-and-go and idling due to congestion and signalized intersections wasted a significant amount of fuel [38]. Recently, an experimental study [39] showed that in some cities, fuel consumption at signalized intersections is more than 50% of the whole trip on average. Currently, the primary technique to address this issue is through adaptive traffic signal controls such as SCOOT and SCATS [40]. These infrastructure-centric solutions have limitations due to the delayed response to traffic flow and low effectiveness when the number of vehicles is low. Broadcasted by the road-side equipment (RSE), signal phase and timing (SPaT) contains the current and future signal phase and timing information, which enables predictive control and smooth driving at signalized intersections. The National Highway Traffic Safety Administration (NHTSA) performed a preliminary analysis of the benefits of broadcasting SPaT, which showed a 90% reduction in red-light violations and up to 35% energy saving [41]. The infrastructure-to-vehicle (I2V) communication enables the vehicle-centric solutions for fuel-saving at signalized intersections [42], which advises drivers to anticipate traffic signals to avoid unnecessary acceleration, deceleration, and stops. The drivers who follow eco-driving advice on average consume 12.9% less fuel, but the travel time increases by 12.7%. With the information of traffic signals and surrounding vehicles available, the vehicle speed trajectory can be planned to reduce fuel consumption at the signalized intersections as

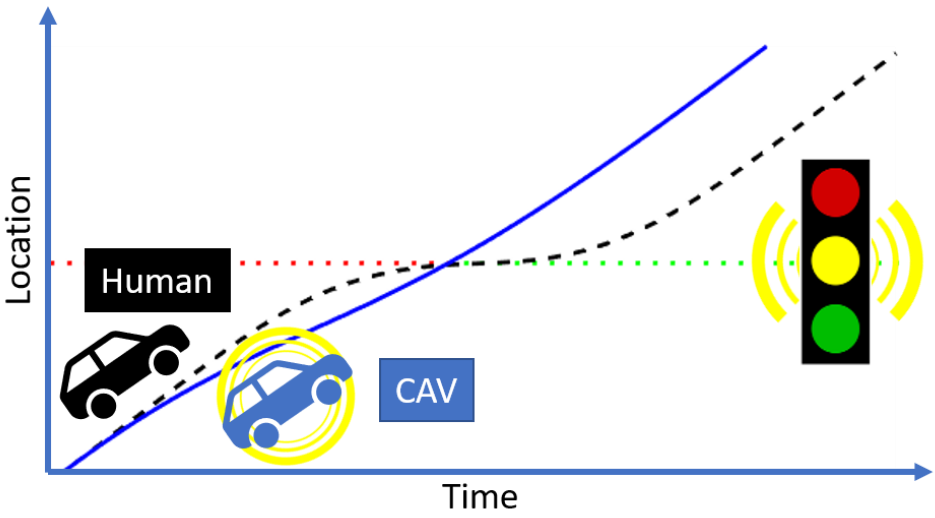


Figure 1.3 Location trajectory comparison of a human driver and a CAV at a signalized intersection

shown in Figure 1.3. This concept is known as Eco-Approach/Departure (EAD). Several pioneering works proposed speed planning algorithms based on the idea of idling minimization [28, 43, 44], finding that the fuel consumption can be reduced by 10% to 15% without explicitly optimizing the powertrain operations. The speed trajectory can be optimized to achieve higher powertrain efficiency using trajectory optimization algorithms such as Dynamic Programming (DP) [45], Pseudospectral [46] and Pontryagin's Maximum Principle (PMP)[47], showing that the full potential for fuel saving is about 40%. However, current studies on EAD are focusing on longitudinal speed optimization, and the extra fuel consumptions due to left and right turns are not addressed.

The fuel consumption benefits of CAV technologies are due to two factors. The I2V communication can provide the future schedule of the traffic lights, and the effective range can be as long as 500 meters using DSRC [11], which enables longer optimization horizon compared with non-connected vehicles. As illustrated in Figure 1.3, with knowledge of future traffic light status, the vehicle can avoid unnecessary decelerations and accelerations. Since the speed trajectory can be controlled optimally, fuel consumption, travel time, and longitudinal jerk can be minimized. Traffic conditions at intersections is also an important factor for vehicles' route choice. The vehicles' route choice at intersections can be formulated as a Markov Decision Process (MDP) as follows

$$x_i^* = \text{argmin}_{x_i \in adj_{out}(x_{i-1})} g(x_i, x_{i-1}) + \mathbb{E}\left(f^*(x_i)\right) \tag{1.1}$$

where x_i is the optimal next link, x_{i-1} is the current link. The next link should be in the adjacent set of current link. $\mathbb{E}\left(f^*(x_i)\right)$ is the expected optimal value function from the next link to destination, $g(x_i, x_{i-1})$ is the transitional cost from the current link to the next link. Due to the high variance in travel speed prediction [48], the value function can only be evaluated as an expected value. However, as the traffic condition including surrounding vehicles and traffic light status are known to the CAV on the current link, the transitional cost can be evaluated deterministically by solving the EAD problem. Since the motivation of our EAD algorithm is to assist route choice decision making, speed and acceleration limit at intersections due to left and right turns are included as constraints in the optimization problem.

In chapter 2 of this dissertation, we present a speed trajectory optimization algorithm with turning motion constraints using the sequential convex optimization method [49]. Sequential convex optimization is a method to obtain local optimal solutions by forming convex sub-problems sequentially. It finds local optimal solutions which do not suffer from the curse of dimensionality. We assume that the traffic signal is known within the problem horizon, and we do not consider the influence of the surrounding vehicles. Also, we do consider the effect of turning at intersections. The turning speed is determined by considering the characteristics of the intersection. The proposed algorithm is flexible in problem formulation: it can consider multiple objectives and can be applied to multiple-vehicle and multiple-intersection cases. In addition to having a flexible problem formulation, it is also important to use a robust numerical solver. We use Gurobi [50] in this research.

While eco-driving can save fuel at the microscopic level, vehicle trip planning and routing based on traffic information and predicted fuel consumption could save fuel and travel time at the trip level, the potential of which has not been deeply explored. An early study of eco-routing using average-speed-based fuel consumption model was conducted, which shows 25% fuel saving compared with a fastest-time routing strategy [51] without detailed microscopic eco-driving behavior. The user equilibrium and the system-optimal behavior were analyzed [52] to understand the network-wide benefits. The authors concluded that the potential of fuel-saving is 7.7% for user equilibrium. Other factors such as signalized intersections [53] and penetration ratio [54] were also studied.

A core piece of eco-routing algorithm development is a robust fuel consumption model. Microscopic fuel consumption models have been studied extensively [55], but for eco-routing, the fuel consumption of a large number of road sections needs to be evaluated, thus fast computation is also required. Macroscopic models [56] have also been studied to estimate fuel consumption without considering heterogeneity in driving, resulting in the same fuel consumption for the same average speed, thus not appropriate for eco-routing. Mesoscopic models using road link average speed and grade are widely used for eco-routing. By considering link-based variables, they can address driving heterogeneity, thus are more accurate than macroscopic models. However, most of the existing mesoscopic models for eco-routing are achieved with parametric regression-based models [51] or

power balance models [57] and are not accurate enough due to the complexity of traffic scenario and nonlinearity of vehicle powertrains. Advanced data-driven methods such as support vector machines (SVM) [58] and neural networks (NN) [59] were also studied, and many outperformed the traditional methods with an increment in model complexity. Recently, a nonparametric model called multivariate adaptive regression spline (MARS) was studied [60]. MARS partitions the feature space into hypercubes with boundaries perpendicular to the axes of the feature space thus can model nonlinear functions.

The main idea of our method is that the fuel consumption model should: (i) use credible physics-driven simulation model (such as Autonomie [55] that we choose); (ii) the driving speed should be from real vehicle data to reflect real-world operating condition of the road links; and (iii) instead of fitting individual trips, the model should aim to match the expected value from many trips. We use the Gaussian Mixture Regression (GMR) to build our model [61]. The GMR technique models the joint density of model input and output then derives the conditional expectation of the output from joint density function of the inputs and output, thus the model is invariant under any coordinate system. After the fuel consumption model is developed, we use it to evaluate the expected fuel consumption of different routing strategies, including shortest-distance, shortest-time, eco-routing, and travel-time-constrained eco-routing. The framework of our model development approach is summarized in Figure 1.4.

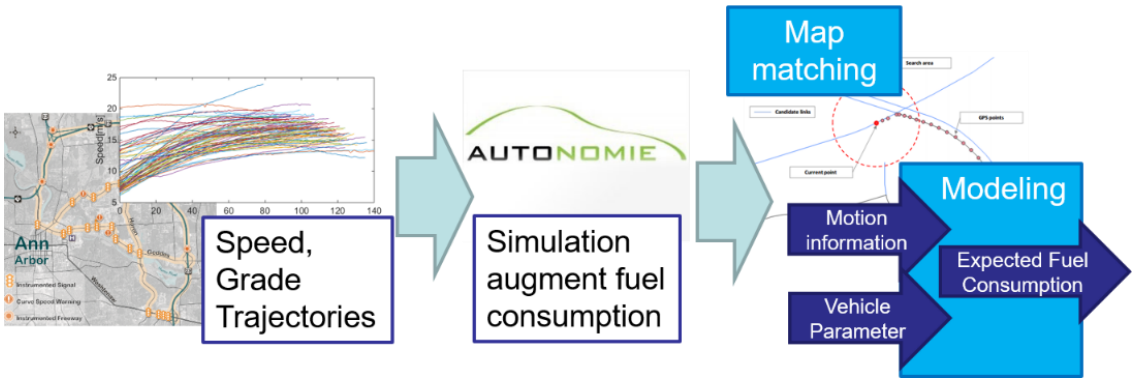


Figure 1.4 Fuel consumption modeling framework utilizing connected vehicle data

The main contributions on eco-routing presented in chapter 3 include: 1) a nonparametric data-driven fuel consumption model based on real-world driving data and Autonomie fuel consumption simulations; 2) a constrained eco-routing strategy addressing

trade-off between travel time and fuel consumption; and 3) numerical simulation study of the fuel consumption and travel time trade-off of different routing strategies.

Although eco-driving and eco-routing concepts have been proposed to reduce fuel consumption and emission at the operation level, as pointed out by a recent study on potential impact on fuel consumption of CAV technologies [24], the major cause for fuel consumption increase is the additional travel demand such as currently underserved population (2% ~ 40%), travel mode shift (~3.7%), and empty vehicle mileage (0%~11%). Thus, ride-sharing is proposed to reduce fuel consumption directly at the travel demand level [62] and has the potential to reduce vehicle mileage traveled by 12% [63]. However, currently the fleet assignment of MOD are either travel time oriented [64–69] or fleet sizing oriented [70–73], and the effect of fuel-saving is mainly due to reduced trips [18]. The full potential in fuel-saving by including trip-level techniques such as eco-routing or minimizing total fleet fuel consumption was not addressed in the literature.

To include fuel consumption in the objective and integrate MOD fleet control with the recent eco-routing [74] concept, we developed a fleet control algorithm based on the work in [65] where the customers' wait time and travel delay time are modeled as constraints. We propose a MOD fleet control algorithm, Eco-MOD, to minimize the fleet operation cost (fuel consumption) while satisfying the customers' travel time constraints. In our numerical study, travel demands generated by POLARIS [75], a mesoscopic agent-based transportation model, are calibrated with data from the Safety Pilot Model Deployment (SPMD) project [76] and used to generate the origins and the destinations of the customers. To evaluate the performance of Eco-MOD under realistic transportation environment, we developed a microscopic traffic simulator using Simulation of Urban Mobility (SUMO) [77] and performed a case study in Ann Arbor using the integrated model.

The main contributions of our work on MOD fleet assignment are: 1) a MOD fleet control algorithm which minimizes fleet fuel consumption directly while satisfying customer travel time constraints; 2) a MOD simulation framework using SUMO and Matlab; 3) findings showing the importance of including fuel consumption in optimization to reduce fleet operating cost; and 4) a traffic network partition method minimizing expected travel time for fleet rebalancing control.

In this dissertation, we focus on fuel consumption optimization of highly automated vehicles (level 4-5) in an urban environment, including three closely related subjects: eco-driving at signalized intersections, link-level fuel consumption model for route optimization, and MOD fleet assignment. The main contributions of this dissertation include: a fast algorithm to optimize vehicle speed trajectory approximately at signalized intersection including turns due to route choice; a data-driven fuel consumption model and apply it to vehicle route optimization; and a framework to combine energy-efficient route optimization and MOD fleet assignment.

1.2 Literature Review

1.2.1 Speed Trajectory Optimization at Signalized Intersections

With the information of traffic signals available, the vehicle speed trajectory can be planned to reduce fuel consumption at the signalized intersections, and this concept is known as eco-approach/departure (EAD). Multi-stage optimization methods have been used in several prior works [43, 78, 79]. The vehicle speed is designed to be the maximum allowable speed if there is enough green time to pass or minimum allowable speed to arrive at the next green window without stopping to avoid idling at the signalized intersections. Subsequently, with the smoothed speed profile designed using the simplified rules, a variety of optimal trajectory following methods are used. Asadi et al. [43] used a model predictive control algorithm with the objective function defined as a weighted sum of trajectory following error and fuel consumption. The work was extended to vehicle platoons [80] and hybrid electric vehicles [81]. Xia et al. [28] experimentally studied the effect of speed advisory with rule-based speed planning and found a 14% reduction in fuel consumption and a 1% reduction in travel time. However, the planned speed trajectory is mainly based on avoiding idling at the intersections, and the powertrain nonlinearity is not considered, thus the potential in fuel saving is not fully addressed.

Trajectory optimization techniques such as dynamic programming were frequently used [45, 82] to realize the full potential of fuel-saving. Instead of precisely known traffic signal states, [45] also studied the cases with inaccurate traffic signal states. An analytical solution was obtained for a single-vehicle case using the Pontryagin's minimum principle

[83]. [46] included the queuing vehicles at an intersection in the analysis. They estimated the queue clearing time and used the pseudospectral method to solve the optimal speed trajectory. In [84], a discretized solution was obtained. They assumed the vehicles only cross the intersection at specific time point such as at the beginning, the middle and at the end of the green phase window. The discrete choices are modeled as nodes in a graph, and a minimum cost path problem was solved using the Dijkstra's algorithm. In many of the works cited above, additional assumptions are made to reduce the computation load, e.g., constant traveling speed along each road section. Also, in urban driving, turning happens frequently. During a turn, the vehicle may incur significant penalty both in fuel economy and time. However, to our best knowledge, the effect of turning has not been considered in the literature.

1.2.2 Data-Driven Fuel Consumption Model and Route Optimization

A series of pioneering works have been conducted to study the energy-saving impact of eco-routing. [51] built a macroscopic fuel consumption model with average speed and road grade as input variables to perform eco-routing on a large scale, resulting in 13% fuel saving and 21% increase in travel time. [85] extended the analysis and included multiple vehicle classes, including heavy-duty diesel trucks, medium-duty diesel trucks, and light-duty gasoline vehicles. The trade-off of fuel consumption and travel time for route choice by travelers is further explored experimentally [86] by providing emission information to households and examining their daily commute decisions. Guo et al. [54] studied the influence of market penetration ratio of CAV on fuel-saving, showing that with increased penetration ratio, fuel-saving can be up to 12% and travel time can be reduced by up to 8%. Recently, [53] took the time window effect of signalized intersections into consideration and designed a routing algorithm based on Markov Decision Process (MDP), and the fuel usage was reduced by 10%. [87] developed an eco-reliable routing algorithm to minimize fuel consumption and late arrival probability in a network with dynamic stochastic travel time. The user equilibrium and system-optimal behavior were analyzed in [52] to understand the network-wide benefits, and the authors concluded that the potential of fuel-saving is 7.7%. Besides the decentralized strategies, centralized traffic assignment is also studied to understand the full potential of energy saving for the whole traffic network

[88–90], which showed that the energy consumption and the travel time could be reduced simultaneously at the system level.

The fuel consumption model used in the routing algorithm plays a central role in the system design. Previously, power balance model based on the vehicles' longitudinal speed was used to develop the macroscopic fuel consumption MOVES [56] and was used by [91–93] to develop eco-routing algorithms., Data-driven approaches such as exponential-polynomial model [51], Support Vector Machines (SVM) [58], neural networks (NN) [59], and Multivariate Adaptive Regression Splines (MARS) [60] were also applied for fuel consumption estimation considering the nonlinearity of powertrain and complexity of traffic scenarios. The performance of data-driven exponential-polynomial models and power balance models were compared in [57], which concluded that the power balance model is not complex enough to simulate mesoscopic link-level fuel consumption. [94] modeled the speed profile instead of fuel consumption and used the synthetic speed profiles to estimate the link fuel consumption to take advantage of accurate microscopic models. Instead of using fuel consumption models, Rakha et al. [52, 95] used probed vehicles in the same class to update the fuel consumption information and studied the fuel-saving impact of eco-routing for the entire network under high connected vehicle penetration ratio. A simulation study showed that the benefit could be from 3.3% to 9.3% compared with typical routing strategies that minimize travel time [96]. The model needs to be simple enough to evaluate the fuel consumption for the city-wide network and accurate enough to address the nonlinearity of fuel consumption and complexity of traffic scenarios. We choose the data-driven approach to address the trade-off between model complexity and accuracy. To address the trade-off between travel time and fuel consumption, we developed a travel-time constrained eco-routing algorithm, which only considers eco-routing solutions with travel time that are longer than the fastest route by no more than a few percentage points.

1.2.3 MOD Fleet Optimization

Control of MOD fleet has been studied extensively to minimize customers' travel time. The fleet assignment problem falls in the category of dynamic Vehicle Routing Problem (VRP) [97] in the demand-vehicle network, which is a generalization of Traveling

Salesman Problem (TSP) by allowing multiple vehicles to serve multiple customers. The problem is typically formulated as an integer programming problem. Several studies developed algorithms to find the exact solution [98–100]. However, considering the NP-hardness of VRP [101] and potentially large problem size, the centralized matching problem is hard to solve directly [102]. Thus, heuristic algorithms such as Genetic/Evolutionary algorithms combined insertion algorithm [103] and bee colony optimization [104] are applied to find a suboptimal solution faster compared with the exact approach. Decomposition-based algorithms focus on reducing the problem size either spatially [105] or use Lagrange relaxation [106] to combine multiple smaller TSP into the master VRP, thus the solution process is accelerated due to the reduction in problem size and parallelization.

Recently, a graph decomposition [107] methods demonstrated that current travel demand for taxis in New York City could be fulfilled with 15% of the existing fleet [65]. A data-driven approach is also used to improve the quality of the solution by considering future demands [64]. [108] developed a simulation optimization (SO) framework using continuous approximation as a metamodel to improve computational efficiency. Other aspects of MOD systems were also explored. A privacy-preserving algorithm was developed [109] to protect the location information of passengers without incurring significant performance drop. Continuous approximation [110] is used to study the dynamics of the fleet and the influence of large fleet to congestion as well as fleet routing problem in a congested network [67, 111]. However, none of the existing work considers fuel consumption when designing the controller, which is a core element in reducing the operation cost of the MOD service provider.

Knowledge of travel demand distribution plays a vital role in the control of MOD fleet. For carpool service with private cars, travel data can be used to identify optimally combined trips for carpooling and can reduce daily car mileage by 44% [112]. Intelligent transportation techniques such as connected automated vehicles provide richer information about travel demand and enable centralized coordination for the MOD fleet. Han et al. [113] showed that with a driverless MOD fleet, the direct control approach is 29% more efficient compared with current price-based indirect control. For service provided by a

commercial fleet, travel demand distribution can be used to control the idling vehicles for rebalancing [113–115] to better meet future trip requests when carpooling is not allowed.

A sampling-based algorithm is also proposed to control ride-sharing fleet using predicted future trip request information [64]. When solving the rebalancing problem, the traffic network needs to be partitioned so the travel demand and vehicle distribution can be characterized as a discrete random variable defined by the partitions and formulated as a linear optimization problem [113]. Currently, the partition is achieved through grid-based approximation [66, 116] and clustering analysis in the spatial coordinate [67, 112]. In dense cities, an integer programming can be formulated [117] to make sure every link is reachable within the time constraint. Thus, to apply the rebalancing algorithm to real traffic network, we developed a traffic network partition algorithm minimizing expected travel time from each link to the closest cluster centers and combined it with our eco-MOD framework.

1.3 Objective, Approaches, and Scope of the Study

The objective of this research is to optimize fuel consumption of connected automated vehicles in an urban environment, including speed trajectory optimization at intersections, data-driven fuel consumption model and eco-routing algorithm development, and eco-MOD fleet assignment. The sub-objectives are connected by the route choice of connected vehicle: EAD at intersections provides transit cost estimation at current intersection when traffic status including surrounding vehicles and traffic signal states are revealed to the vehicle; data-driven fuel consumption and route optimization provides expected value function for future links where the expected value of traffic information is available; using the expected fuel consumption cost, eco-MOD can assign vehicles and customers not only minimizing travel time but also fleet fuel consumption, which can reduce operation cost and reduce emission at the same time.

To use eco-approach/departure (EAD) as transit cost for route choices, one need the algorithm to be fast and robust enough for online computation. Although the optimal solution can be found through dynamic programming, the curse of dimensionality limits its computation speed and robustness. Thus, we propose the sequential approximation approach to solve a sequence of mixed-integer optimization problems with quadratic

objective and linear constraints. The number of integer states is the number of green windows of all traffic lights in the planning horizon, thus the number of integer variables is limited. The speed and acceleration constraints at intersections due to route choice are addressed using a barrier method. In this part of the dissertation, we limit the problem to a single intersection due to the route choice application and only consider free flow scenarios, but the algorithm can be extended to multiple intersections and congested scenarios where a leading vehicle is included as a constraint if an intersection driver model is available.

The mesoscopic fuel consumption model is developed through a data-driven approach considering the tradeoff between model complexity and accuracy. A large quantity of naturalistic driving data is used to develop the model. Since the selected dataset does not contain fuel consumption data, a microscopic fuel consumption simulator, Autonomie, is used to augment the information. Gaussian Mixture Regression [61] is chosen to build the model due to its ability to address nonlinearity. Instead of selected component number by cross-validation, we use the Bayesian formulation, which models the indicator of components as a random variable which has Dirichlet distribution as prior. The model parameters are obtained through max-a-posterior inference from data, and the conditional expectation of fuel consumption on input variables can be obtained in closed form since the individual components follow the Gaussian distribution. The model is used to estimate fuel consumption cost for routing algorithm. In this part, we assume the traffic network is static.

The fuel consumption model and the eco-routing algorithm are integrated with MOD fleet assignment. The MOD control framework is inspired by [65], where customers' travel time requirements are modeled as constraints, thus provides flexibility for cost function design. At the current phase, we assume the traffic network is static and use offline calculated travel time and fuel consumption to assign the fleet. To rebalance the idling vehicles, we developed a traffic network partition algorithm which minimizing the expected travel time within each cluster. The demand matching algorithm [113] is used to assign the rebalancing fleet. It is argued in [9] that ride-sharing has the potential to reduce fuel consumption. However, this dissertation shows that if the objective function of fleet assignment is only travel time, even if ride-sharing is allowed, the fleet fuel consumption could increase compared with the baseline where personal vehicles are used for travel.

1.4 Research Contribution

The contributions of this dissertation are listed below:

- A speed trajectory optimization algorithm at signalized intersections with speed and acceleration limits due to left and right turns is presented. The algorithm can be extended to multiple intersections and multiple vehicles.
- A data-driven fuel consumption model based on real-world driving data and Autonomie fuel consumption simulation and analysis of trade-off between travel time and fuel consumption of different routing strategies including fastest route, shortest route, eco-route, and travel time-constrained eco-route.
- A framework for eco-MOD combining eco-routing strategy and MOD fleet assignment with ride-sharing is developed, showing the importance of including fuel consumption in the assignment algorithm.
- A traffic network partition algorithm minimizing expected in-cluster travel time for MOD idling fleet rebalancing is proposed.
- A traffic simulation framework is developed using SUMO and calibrated using the SPMD database.

1.5 Outline of the Dissertation

This dissertation is organized as follows: in Chapter 2, the speed trajectory optimization algorithm is presented. In Chapter 3, the data-driven fuel consumption model is presented, and two versions of eco-routing algorithms using the model, with and without travel time constraints are discussed. In Chapter 4, the eco-MOD framework is presented. The performance is compared with personal vehicles traveling and MOD that minimizing travel time. In Chapter 5, the traffic network partition algorithm and the idling fleet rebalancing algorithm is presented. In Chapter 6, the conclusion and future works are presented.

CHAPTER 2
Speed Trajectory Optimization at Signalized Intersections

2.1 Introduction

In this chapter, we present a speed trajectory optimization algorithm considering turning motion constraints using the sequential convex optimization method. Sequential convex optimization is a method to obtain a local optimal solution by forming convex sub-problems sequentially. It finds a local optimal solution in a computationally efficient manner and scales better for higher-dimensional problems. We assume that the traffic signal state (red/green) is known within the problem horizon, and we do not consider the influence of other road users. The problem can be solved over the whole problem horizon, by manipulating the speed profile over multiple road sections. The second advantage is that we do consider turning at intersections. The driving speed during turning is determined by considering the characteristics of the intersection. The third advantage is the flexibility of the proposed method: it can combine multiple objectives and can be applied to multiple-vehicle and multiple-intersection cases. In addition to having a flexible problem formulation, it is also important to use a robust numerical solver. We use Gurobi [50] in this research.

The rest of the chapter is organized as follows: The model of a passenger car is constructed in Section 2.2. Section 2.3 presents the optimization problem. Section 2.4 shows the optimization results and their analysis. Finally, conclusions are presented in Section 2.5.

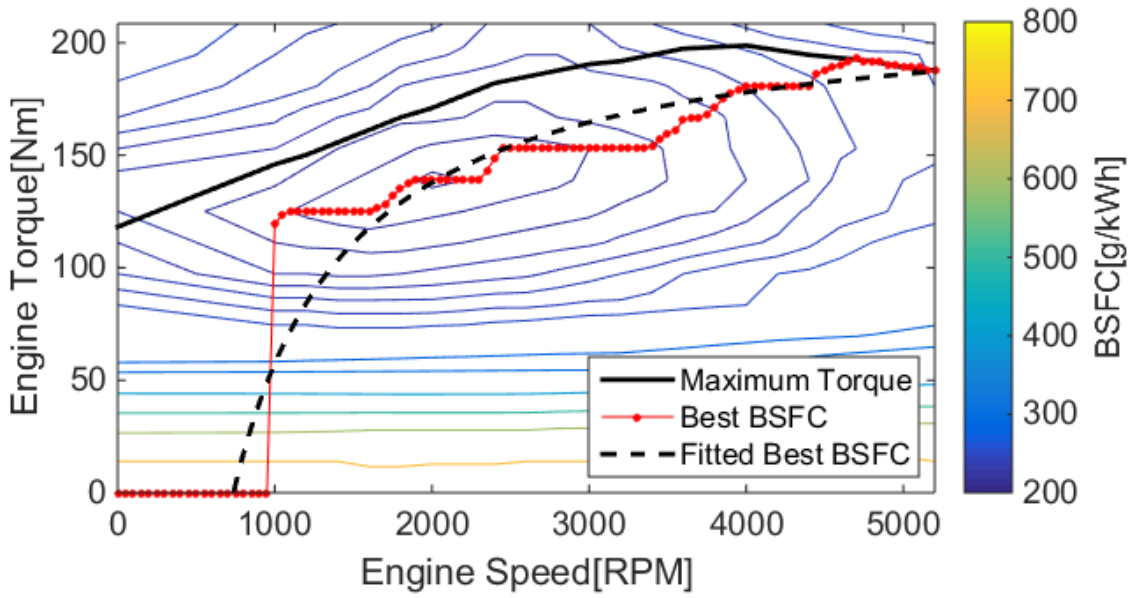


Figure 2.1 BSFC map of an engine

2.2 Vehicle Model

2.2.1 Fuel Consumption Model

In this study, we consider a passenger car equipped with a 4-cylinder 2.5-liter internal combustion engine and a continuously variable transmission (CVT). A simplified powertrain model is used with the following assumptions: (1) the powertrain efficiency is described by a static look-up table; (2) the CVT keeps the engine operating along the minimum brake specific fuel consumption (BSFC) line; (3) a simple longitudinal dynamics of the vehicle [118] is used.

$$M\dot{v} = F - Mg\sin\theta - Mgf\cos\theta - 0.5\rho C_d A(v + v_w) \tag{2.1}$$

where M is the vehicle mass, v is the vehicle speed, F is the longitudinal force, g is the gravity coefficient, θ is the road grade, f is the rolling resistance coefficient, ρ is the air density, C_d is the drag coefficient, A is the vehicle cross-sectional area, and v_w is the wind speed. In the following, we assume a flat road and zero wind speed. The driving force is a function of gear ratio and engine torque

$$F = i_g i_f \eta_T T_e / r_w \tag{2.2}$$

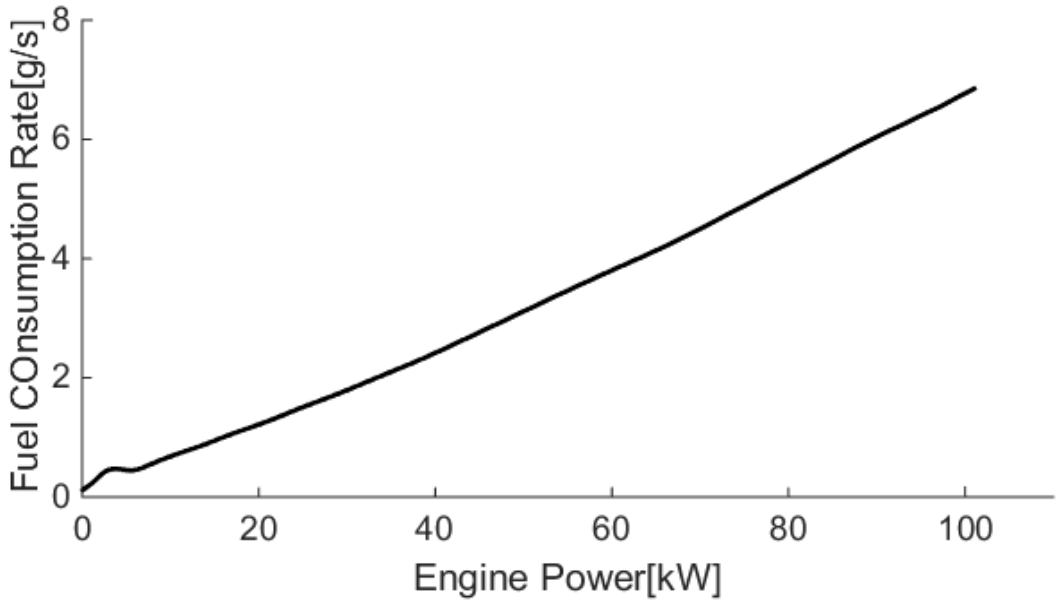


Figure 2.2 Fitted fuel rate of the engine as a function of the engine power

where i_g is the transmission gear ratio, i_f is the final drive ratio, η_T is the transmission efficiency, r_w is the wheel radius, T_e is the engine torque. The fuel consumption is estimated from the static fuel consumption map, as shown in Figure 2.1.

The idling engine speed is 800 RPM, and the idling torque is assumed to be 0 Nm. We assume engine stop-start [119] is not available, thus there is idling fuel consumption. The optimal BSFC point is around 2000 RPM and 140 Nm. To incorporate the transient effect of engine operation on fuel consumption, we follow the methods of Li et al. [120] by adding a modification term to the static fuel consumption map. The total fuel consumption Q is

$$Q = Q_{static} + k_e \dot{T}_e \tag{2.3}$$

where Q_{static} is the fuel consumption rate from the static lookup table, k_e is the coefficient for transient engine operations. The coefficient k_e is obtained from the drive cycle FTP-72, based on which the transient engine operation was found to increase the fuel consumption by 4~5% [120, 121]. The transmission is assumed to be controlled optimally so that the engine stays on the best BSFC line

$$\omega_{opt} = \frac{b}{1 - kT_{opt}} \tag{2.4}$$

where ω_{opt} is the engine speed along the best BSFC line, T_{opt} is the engine torque, k and b are parameters to be identified. With this ideal CVT, the fuel consumption rate is a function of the engine power. The fitted function of fuel consumption is shown in Figure 2.2.

2.2.2 Effect of Turning

We assume turning imposes speed and acceleration limits at intersections. For the speed constraint, we consider the simplified unbanked turning model [122], which computes the speed limit from the friction limit

$$v_{max1} = \sqrt{Rg\mu} \tag{2.5}$$

where R is the turning radius, μ is the friction coefficient. Also, we assume there is a limit on vehicle speed due to ride comfort. When the maximum allowed lateral acceleration is a_y, then the maximum speed is limited by

$$v_{max2} = \sqrt{Ra_y} \tag{2.6}$$

2.2.3 Baseline Driver Deceleration/Acceleration Model

A human behavior model at intersections is used as the benchmark to evaluate the effectiveness of the algorithm. The human driver deceleration/acceleration behavior model [123] is shown in Eq. (2.7). The model was evaluated in [124] and confirmed to match experimental data very well.

$$a = ra_m\theta(1-\theta^m)^2 \tag{2.7}$$

In Eq. (2.7), ra_m is a function of m, θ is the normalized acceleration/deceleration time, defined as time divided by desired acceleration/deceleration time. The model parameters are all adopted from [123]. The reaction distance is defined as the maximum distance to the intersection where the driver starts to decelerate if the light state is red. The desired acceleration/deceleration time and distance are calculated from [123].

$$t_a = \frac{v_f - v_i}{0.5778 + 0.0669\left(v_f - v_i\right)^{\frac{1}{2}} - 0.0182v_i} \tag{2.8}$$

$$x_a = \left(0.467 + 0.0072v_f - 0.0076v_i\right)\left(v_f + v_i\right)t_a \tag{2.9}$$

$$t_d = \frac{x_d}{(0.473 + 0.0056v_i - 0.0049v_f)(v_f + v_i)} \tag{2.10}$$

$$ra_m = \frac{\left[\frac{2(m+1)(m+2)}{m^2}\right]|v_f - v_i|}{t_{a(d)}} \tag{2.11}$$

where x_a and x_d are desired acceleration/deceleration distance, t_a and t_d are desired acceleration/deceleration time, v_f and v_i are desired final and initial speeds. The reaction distance is 150 m to the intersection, which is the mean distance to start deceleration from [125]. We assume that the desired deceleration distance is the distance to the intersection when the light is red and the driver is within the reaction distance. If the speed limit is 17.88m/s, parameter m for deceleration is -0.7193, and 9.1244 for acceleration from

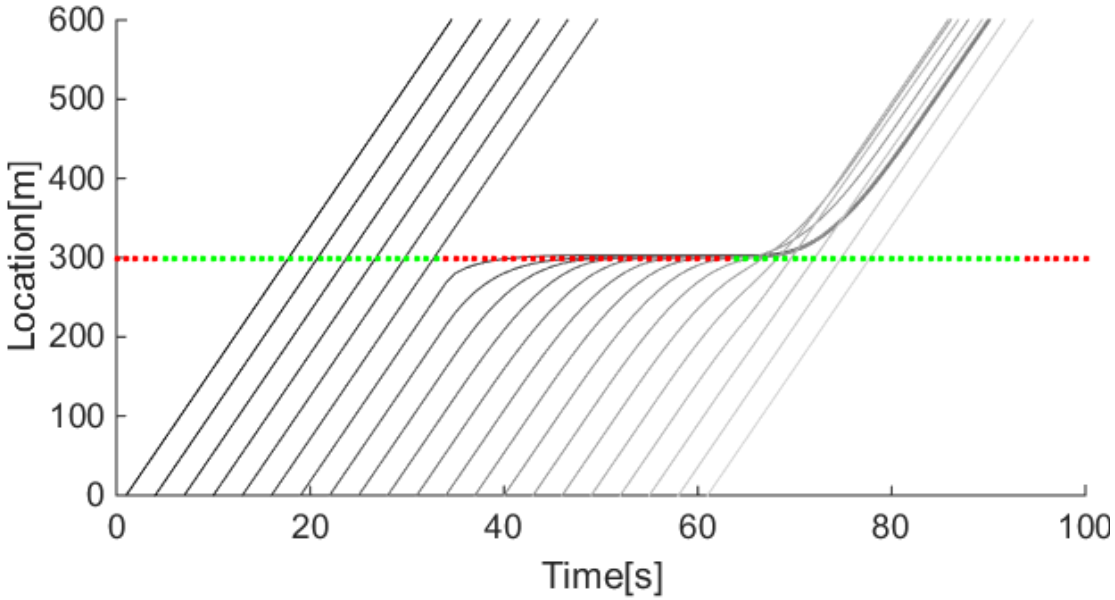


Figure 2.3 Intersection Motion Trajectory From Human Driver Model

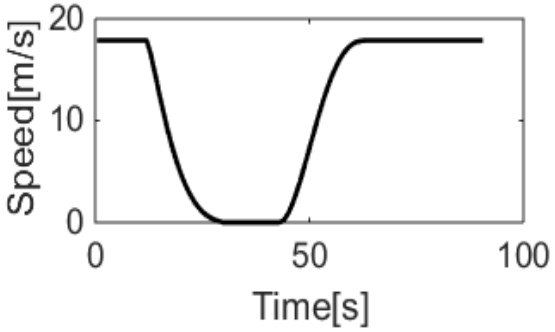


Figure 2.4 Intersection Speed Trajectory from Human Driver Model

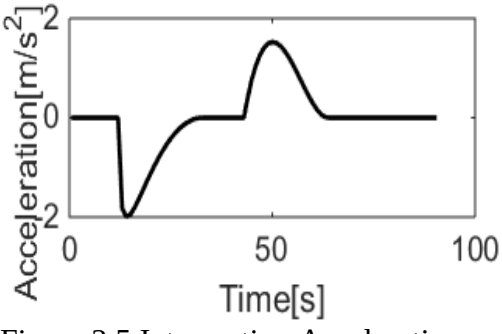


Figure 2.5 Intersection Acceleration Trajectory from Human Driver Model

(2.8)(2.9)(2.10) and [123]. The sample motion trajectories with different traffic signal phase are shown in Figure 2.3.

2.3 The Eco-Driving Problem and Solution Methodology

2.3.1 Mixed-Integer Problem Formulation

The speed trajectory optimization problem is formulated as a non-convex optimization problem. The objective is to minimize the fuel consumption, traveling time and meeting ride comfort requirement over the planning horizon, and the constraints including speed limits, acceleration limit, and red light violation. The vehicle motion is discretized with a sampling time, and during each sampling time, the acceleration is assumed to be constant. In the discrete-time, speed and displacement are

$$v(k+1) = v(k) + a\Delta t \tag{2.12}$$

$$d(k+1) = d(k) + \frac{(v(k+1) + v(k))}{2}\Delta t \tag{2.13}$$

The traction power at each time step is derived from the longitudinal vehicle model (2.1).

$$P(k) = Ma(k)v(k) + Mgfv(k) + 0.5\rho C_d A v(k)^3 \tag{2.14}$$

As discussed in Section 2, the fuel consumption is only a function of the engine power along the BSFC line. Therefore, fuel consumption FC(k) is

$$FC(k) = C_f(P_{eng})P_{eng}(k) = C_f(P_{eng})P(k)/\eta_T \tag{2.15}$$

where $C_f(P_{eng})$ is the fuel consumption coefficient.

A travel time penalty is imposed through a negative vehicle speed term over the planning horizon, and a penalty on acceleration and jerk represents the desire for better ride comfort.

$$J_{comfort}(k) = a(k)^2 + w_j(a(k) - a(k-1))^2 \tag{2.16}$$

The final objective function is defined as a weighted sum of fuel consumption, traveling time, and ride comfort.

$$J = w_{fc}\sum_{k=1}^{T} J_{fc}(k) - w_t\frac{1}{T}\sum_{k=1}^{T} v(t) + w_c\sum_{k=1}^{T} J_{comfort}(k) \tag{2.17}$$

where T is the horizon time, w_{fc}, w_t, w_c are weighting parameters for fuel consumption, traveling time and ride comfort, respectively.

To ensure the vehicle crosses the intersection without violating the red light, we define the constraints to address the green phase windows. $t_{r2g}^{(i)}$ is defined as the time the light changes from red to green for the i^{th} green phase window of the subject intersection, and $t_{g2r}^{(i)}$ is defined as the time the light changes from green to red. These time steps are critical for speed trajectory optimization at signalized intersections. To put the constraints into a matrix form, we define the vehicle location at the critical times and the indicator of crossing windows as follows

$$k = [k_1, \ldots, k_N]^T \tag{2.18}$$

$$d_{r2g} = [d_{r2g}^{(1)}, \ldots, d_{r2g}^{(N)}]^T \tag{2.19}$$

$$d_{g2r} = [d_{g2r}^{(1)}, \ldots, d_{g2r}^{(N)}]^T \tag{2.20}$$

where k is a singleton vector with only one of the indicators equals to 1, and the other elements are all 0. N is the total number of green phase windows in the planning horizon at the subject intersection. d_{r2g} and d_{g2r} are vectors of vehicle locations at critical times. With the variables defined in the vector form, the constraint for valid intersection crossing can be defined as

$$\sum_{i=1}^{N} k_i = 1, k_i \in \{0,1\} \tag{2.21}$$

$$k^T d_{r2g} < 0 \tag{2.22}$$

$$k^T d_{g2r} > 0 \tag{2.23}$$

Other constraints include the speed limit constraint, the acceleration limit constraint, and jerk constraint. Unlike the study in [7], we do not allow the vehicle to exceed the speed limit to catch a green light.

$$v(k) \in [0, a_{max}], a(k) \in [a_{min}, a_{max}] \tag{2.24}$$

$$a(k) - a(k-1) \in [Jerk_{min}, Jerk_{max}] \tag{2.25}$$

As discussed above, the problem is formulated as a non-convex optimization problem, with speed and position as the state variables, and acceleration and the crossing green phase window indicator as the input variables. The crossing green phase window

indicator is an integer variable. The constraints are either linear or quadratic. However, the objective function is non-convex, with nonlinear fuel consumption and aerodynamic drag. The sequential convex optimization technique is applied to solve the problem. Sequential convex optimization finds a local optimal solution by forming a convex sub-problem of the original problem sequentially. The method has been used to solve trajectory planning for aircraft, manipulators, and humanoid robots [49, 126]. To make the approximation at each iteration valid, the trust region method is applied, that is, an additional constraint is applied to make the step size small. At each iteration, the two non-convex terms are approximated by the values from the previous iteration. At iteration $j+1$ the objective function of fuel consumption is shown below.

$$\begin{aligned} J_{fc}^{j+1} = diag(fc_k^j/p_k^j)(a^T(MD \\ +0.5\rho C_d A D^T diag(v_k^j)D)a + {v_0}^T Ma \\ +\rho C_d A {v_0}^T diag(v_k^j)Da + MgfDa + K) \end{aligned} \tag{2.26}$$

where K is a constant term related only to the initial speed, p_k^j is the traction power at iteration j and time step k, fc_k^j is the fuel consumption rate at iteration j and time step k, a is the vector form of the acceleration in the planning horizon, D is an N×N lower triangle matrix representing the kinematic model (2.12).

$$D = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{pmatrix}, v = Da + v_0 \tag{2.27}$$

The assumption here is that in the trust region, the fuel consumption and the speed of the last iteration are close approximations of the actual value. The trust region method would impose additional linear constraints on speed and acceleration

$$v(k)^{j+1} \in [v(k)^j - \rho_v, v(k)^j + \rho_v] \tag{2.28}$$

$$a(k)^{j+1} \in [a(k)^j - \rho_a, a(k)^j + \rho_a] \tag{2.29}$$

where ρ_v and ρ_a are the trust-region radius of speed and acceleration, respectively. It is also noted from the solver that since breach-and-bound is used to solve the mixed-integer problem, the application of the trust region at each iteration would reduce the size of the search tree.

Since the multi-objective optimization problem is solved by the weighted sum method, the objective function is not guaranteed to be positive-semidefinite. Therefore, during each iteration, standard sequential quadratic programming (SQP) is used to obtain the solution.

The initial cost function for fuel consumption is set to minimize the traction power rather than the fuel consumption. The initialization helps to achieve a close starting point to a local-optimal solution considering the application of the traction power as the fuel consumption cost from the literature [43].

$$J_{fc}^{0} = a^{T} M D a + v_{0}^{T} M a \tag{2.30}$$

2.3.2 Incorporation of Turning Motion

As discussed in the previous section, we assume the geometry of the intersection can be neglected when incorporating the turning motion, which is modeled as speed and acceleration limits as follows

$$0 \leq v(t_{cross}) \leq v_{turn} \tag{2.31}$$

$$a_{turn\_min} \leq a(t_{cross}) \leq a_{turn\_max} \tag{2.32}$$

where t_{cross} is the crossing time at the intersection, v_{turn} is the maximum speed during turning, a_{turn_min} and a_{turn_max} are acceleration limits. Since the intersection crossing time is unknown even when the crossing window is determined, the crossing speed and acceleration constraints are achieved through soft constraints, as shown in Figure 2.6 and Figure 2.7.

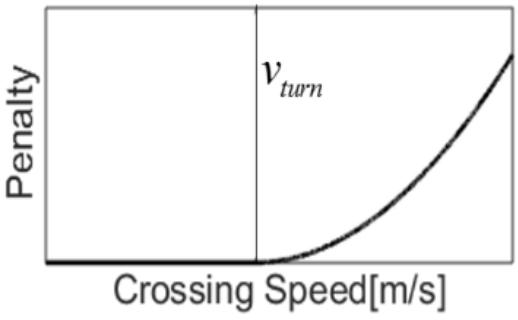


Figure 2.6 Crossing Speed Soft Constraint

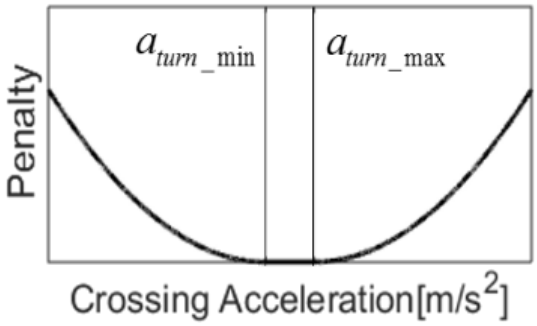


Figure 2.7 Crossing Acceleration Soft Constraint

The soft constraint is implemented with a piecewise linear objective in Gurobi [50]. With the convex nature of the quadratic and the SQP approximation of the original problem, using soft constraints will preserve the convexity of the sub-problem at each iteration. However, due to the use of the trust region, at each iteration, the converging step size is small. Also, using soft constraints increases computation time for mixed-integer programming. When the crossing time change between two consecutive iterations is larger than a specific threshold, we reinitialize the sequential convex optimization by resetting the cost function as propulsion power, removing the trust-region constraint and adding linear constraints for crossing speed and acceleration.

To define the stopping criteria for the sequential optimization, the distance of improvement between iterations is defined. The criterial iteration variables are fuel consumption rate, vehicle speed, and crossing speed. The distance of improvement is defined as

$$\Delta fc^j = \max_k\left(|fc(k)^j - fc(k)^{j-1}|\right) \tag{2.33}$$

$$\Delta v^j = \max_k\left(|v(k)^j - v(k)^{j-1}|\right) \tag{2.34}$$

$$\Delta {v_{cross}}^j = \left(v(t_{cross})^j - v_{turn}\right)_{v(t_{cross})^j > v_{turn}} \tag{2.35}$$

$$\Delta G^j = \sqrt{\Delta fc^{j2} + \Delta v^{j2} + \Delta {v_{cross}}^{j2}} \tag{2.36}$$

where ΔG^j is the difference between two consecutive iterations evaluated at iteration j, defined as the square root of the sum of the squares of the difference in fuel consumption rate, vehicle speed, and crossing speed. The iterative algorithm stops when ΔG^j becomes less than the selected threshold.

2.4 Optimization Results and Discussion

We first start from a single vehicle, single intersection case. The problem horizon is set to be 90 seconds. The speed limit is 17.9 m/s, or 40 mph. The acceleration limits are ±3 m/s^2, as used in [41]. The jerk limits are set to be ±0.5 m/s^3. Mixed integer programming is known to be NP-hard, and the computation time depends on the number of integer states and the problem size [127]. For our case, the integer variable is the crossing window

indicator, and the number is small in the problem horizon. The problem is solved with a computer with Intel i7-4710MQ CPU and 16 G RAM. When the turning motion is not considered, the computation time varies between 0.4 s and 1.9 s depending on the traffic light status. When the turning motion is considered, the computation time increases dramatically, varying between 6.6 s and 8.4 s depending on the traffic light status and the gap between initial speed and the desired turning speed.

2.4.1 Optimality Analysis

Sequential convex optimization (SCP) is a method to obtain local optimal solutions for non-convex problems. To verify the optimality of the solution, the speed trajectory is compared with solutions from dynamic programming (DP). Although dynamic programming achieves the global optimal solutions, the algorithm is computationally expensive and suffers from the phenomenon known as the curse of dimensionality. With pre-computed cost-to-go, DP would take 628 s to obtain the optimal solution. The speed trajectories for different traffic light phase are shown in Figure 2.8, with red dots representing the red phase of the traffic light and green dots representing the green phase. The change in the signal phase is achieved by fixing the traffic signal and changing the vehicle departure time.

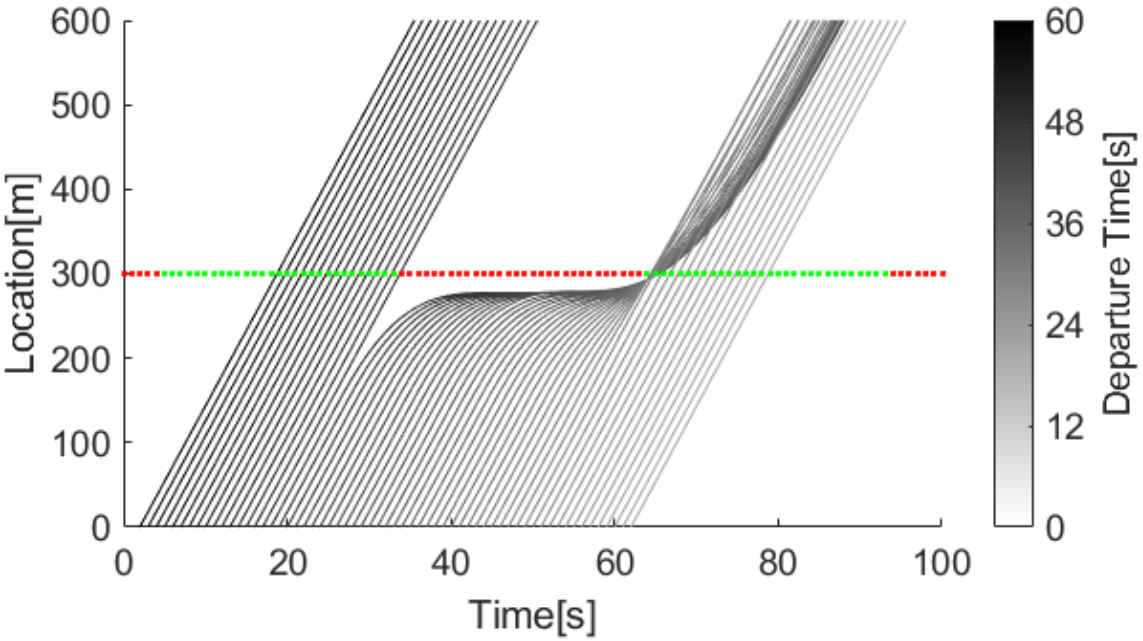


Figure 2.8 Vehicle Trajectories for Different Signal Phase

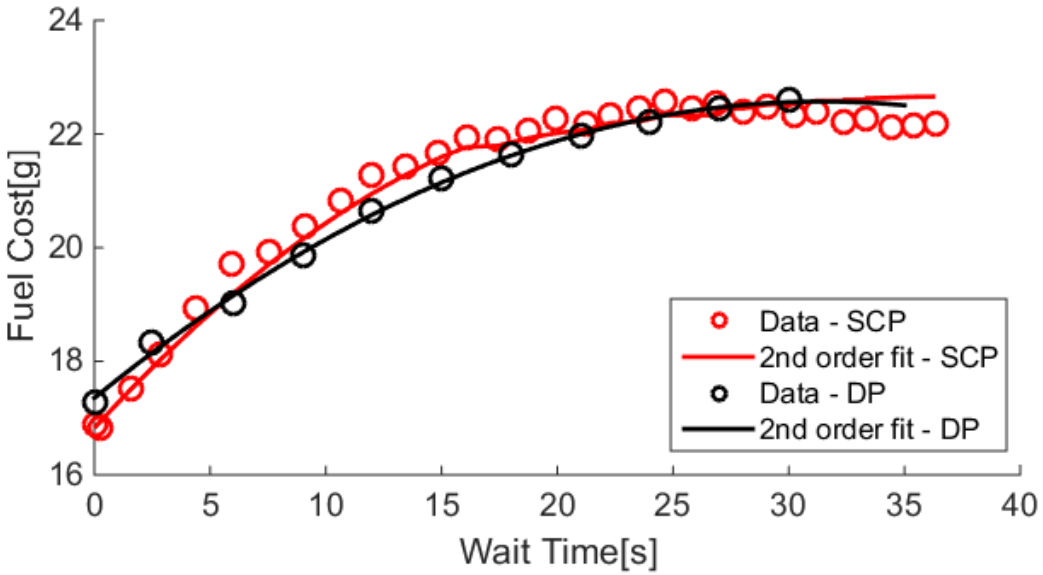


Figure 2.9 Comparison between DP and SCP solutions

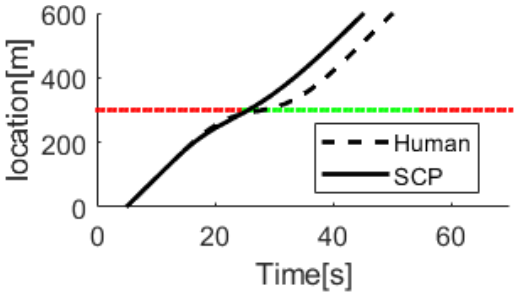


Figure 2.10 Example results of vehicle position

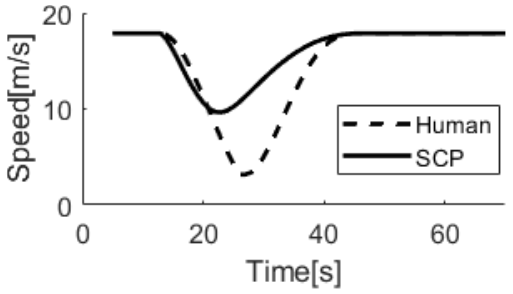


Figure 2.11 Example results of vehicle speed

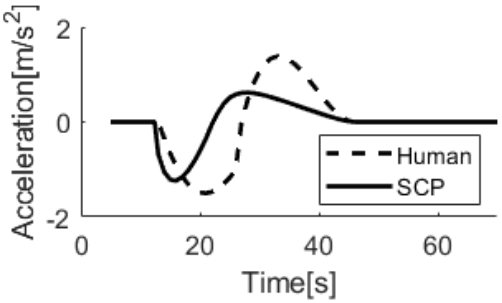


Figure 2.12 Example results of vehicle

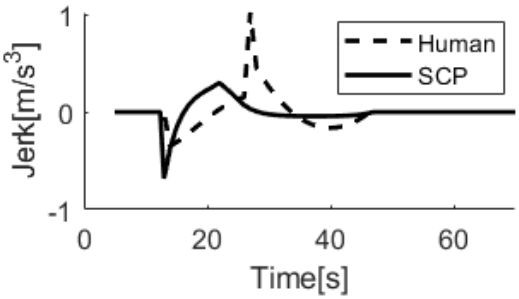


Figure 2.13 Example results of vehicle

Figure 2.9 shows the results from both DP and SCP. The waiting time is defined as the time difference between the actual travel time and free-flow travel time; fuel cost is the fuel consumed from 300 m before the intersection to 300 m after the intersection and reaching the original speed. In this section, the reaction distance of all optimal controllers are set to be 150 m, which is the mean distance of human drivers to start deceleration from

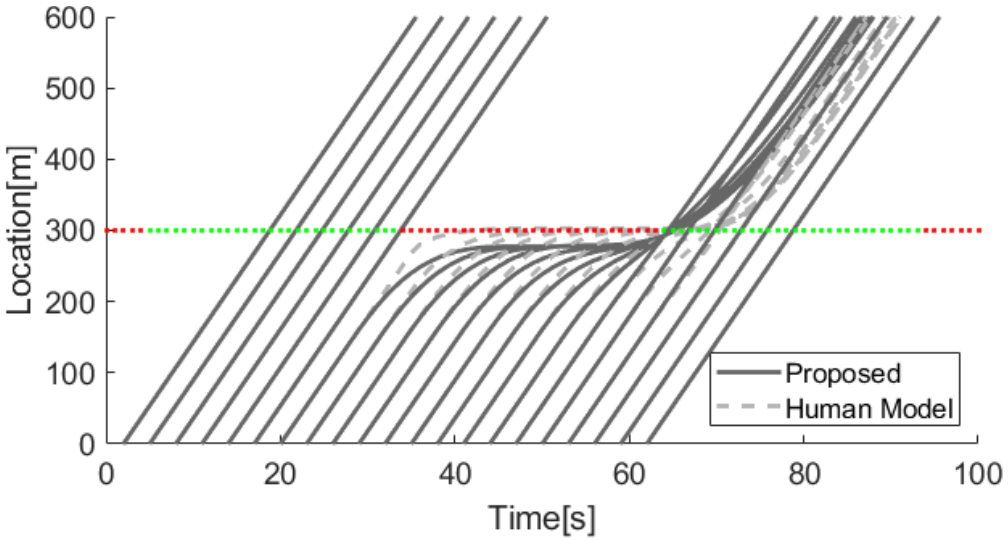


Figure 2.14 Comparison of vehicle position from proposed approach and human driver model under different traffic phase

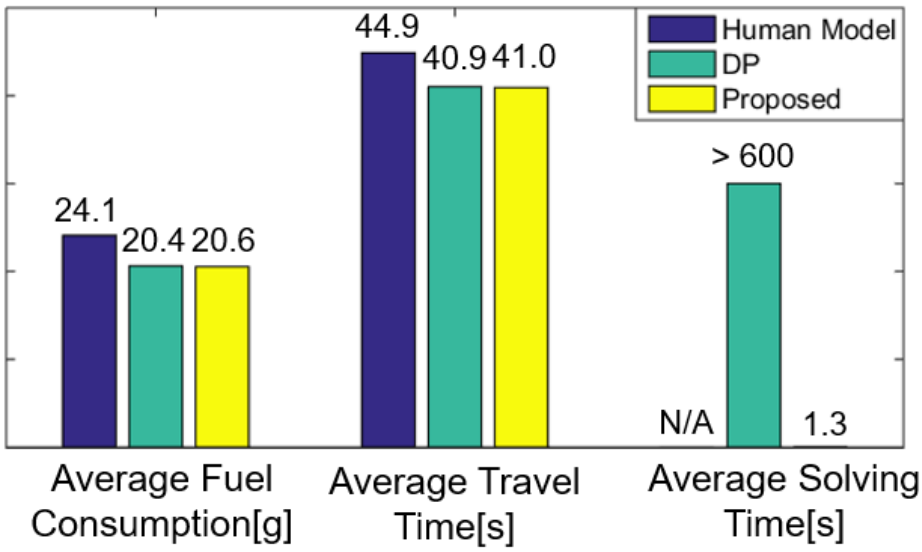


Figure 2.15 Performance comparison of DP, human driver model and proposed method

[125]. The relation between fuel cost and wait time can be fitted with a 2nd order curve. The maximum difference between DP and SCP results is 4.28%. The optimal results are obtained for different traffic signal phases. The average fuel consumption reduction is 12.1%, and time reduction is 7.5% for single intersection cases compared with the human driver model results. The reduction in fuel is as high as 35.6%, and the reduction in time is as high as 16.4%. A sample trajectory comparison of location, speed, acceleration, and jerk are shown in Figure 2.10 to Figure 2.13. The position trajectories obtained using the

proposed approach and a human driver model are summarized in Figure 2.14. The optimization results achieve smoother driving compared with the results from a driver model because future traffic light status is known and used. The comparison of average fuel consumption, average travel time and average solving time are summarized in Figure 2.15

2.4.2 Turning Motion Consideration

To understand the benefits of including turning motion in the optimization, we consider a left turn at an intersection selected from Ann Arbor road network as shown in Figure 2.20. The turning radius is set to be 35 m, the comfort lateral acceleration level is set to be 3 m/s^2, and the road friction coefficient is 0.7. The longitudinal acceleration during turning is assumed to be 0. The maximum speed to pass through the intersection is 13.1 m/s, and the comfortable maximum speed to pass through the intersection is 10.2 m/s. If the speed limit of the road is higher than the maximum safe passing speed, the method without turning motion constraints cannot obtain a feasible solution for free flow since the

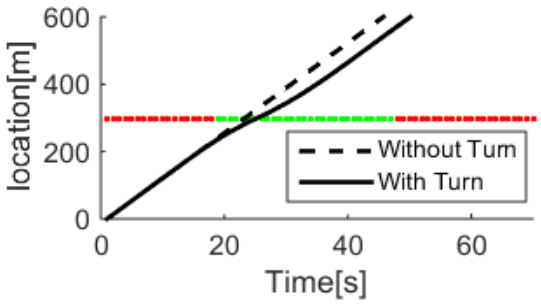


Figure 2.16 Example results of vehicle location during a turn

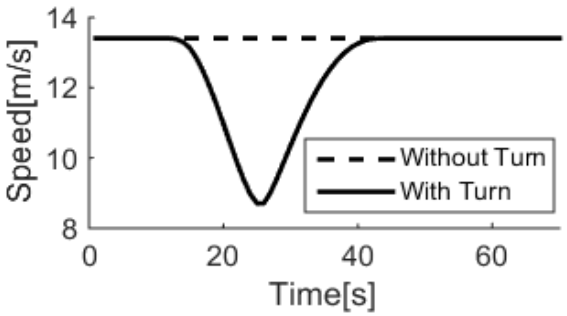


Figure 2.17 Example results of vehicle speed during a turn

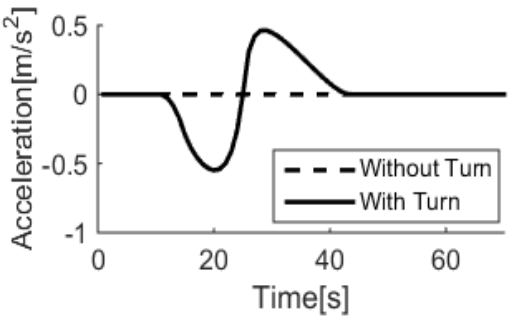


Figure 2.18 Example results of vehicle acceleration during a turn

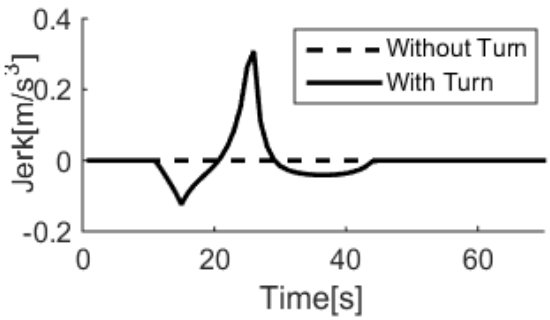


Figure 2.19 Example results of vehicle jerk during a turn

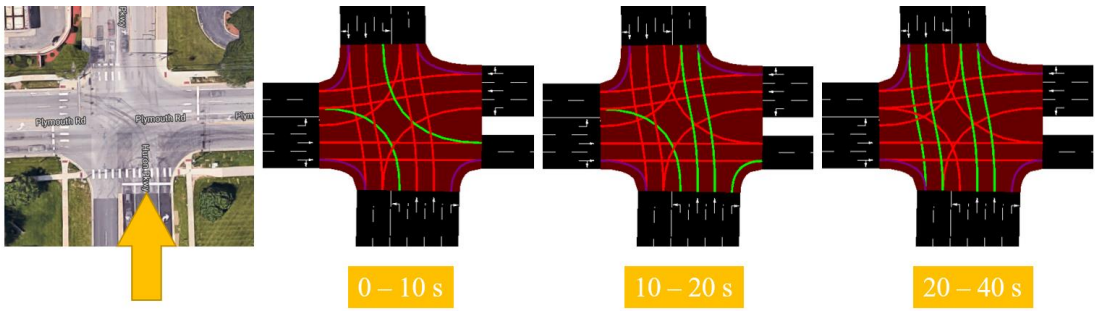


Figure 2.20 Traffic signal phase and timing of south leg at Plymouth Road - Huron Parkway intersection

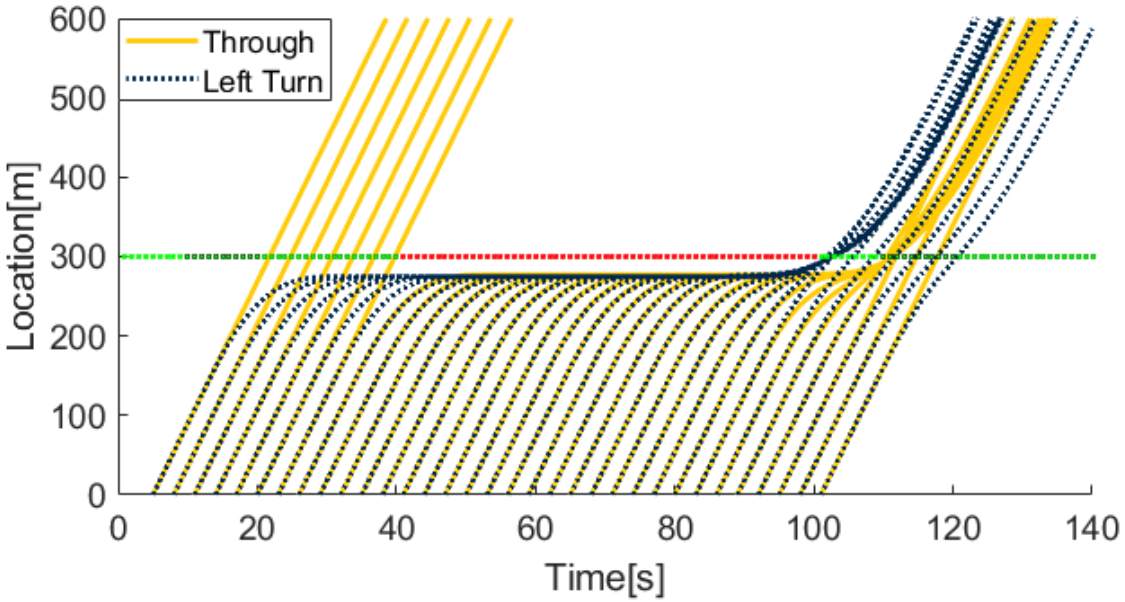


Figure 2.21 Comparison of the vehicle trajectories under different traffic phase

optimal solution is passing the intersection at a constant speed. We set the speed limit to be 13 m/s for a fair comparison. Results with and without turning motion consideration are shown in Figure 2.16 to Figure 2.19. As shown in the trajectories, when turning motion is involved, extra deceleration and acceleration are required to satisfy the speed and acceleration constraints at the intersection. To demonstrate the effect of the traffic signal on route cost, we calculate the optimal speed trajectory using the traffic signal phase and timing shown in Figure 2.20. In this study, we focus on vehicles approaching the intersection from the south leg of the intersection. The signal cycle length is set to be 100 s. The resultant location trajectories with initial time varying from 1 s to 100 s are shown in Figure 2.21.

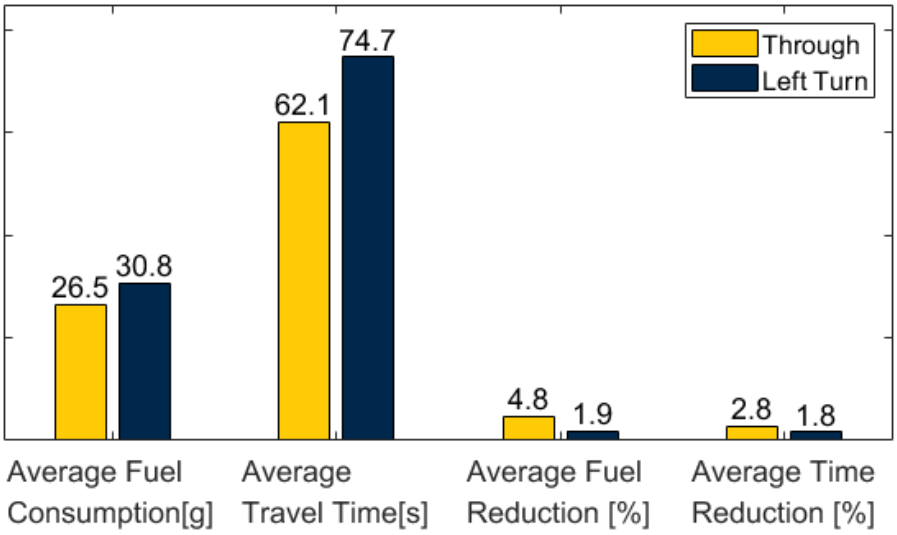


Figure 2.22 Comparison of fuel consumption, traveling time, and fuel and time reduction compared with human model for Eco-Approach and Departure for through and left turn traffic given signal phase and timing from Figure 2.20

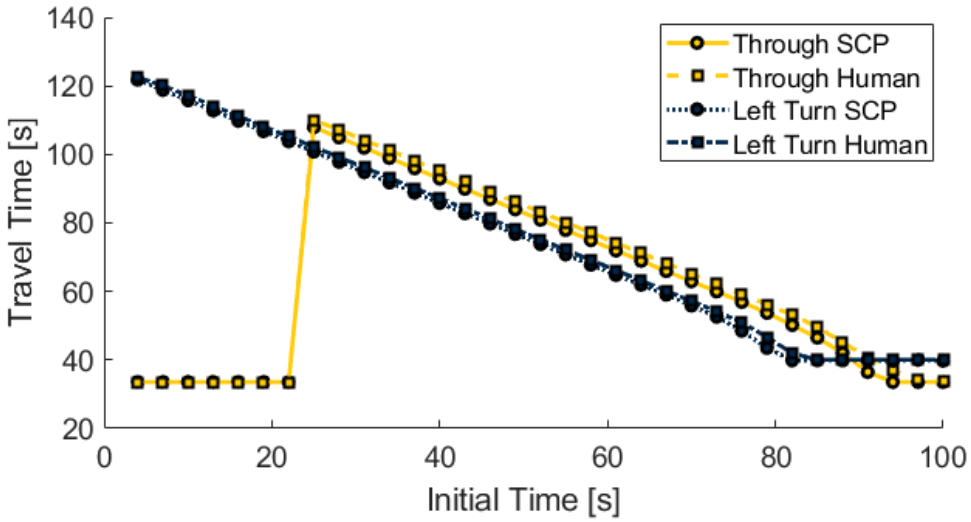


Figure 2.23 Travel time for through and left turn traffic with different initial time

To show the influence of turn motion, fuel consumption, traveling time, and the fuel and time saving compared with human driver baseline are shown in Figure 2.22. Fuel consumption and traveling time are defined as fuel and time-lapse from 300 m before the intersection to 300 meters after the intersection and reaching the original speed. In this comparison, we only consider the single intersection, and future routing cost is not included here. Also, we assume the initial time follows a uniform distribution. The expected fuel consumption and travel time of through traffic are both lower compared with the left-turn traffic. On average, the fuel consumption is 12.9% lower, and the traveling time is 16.7%

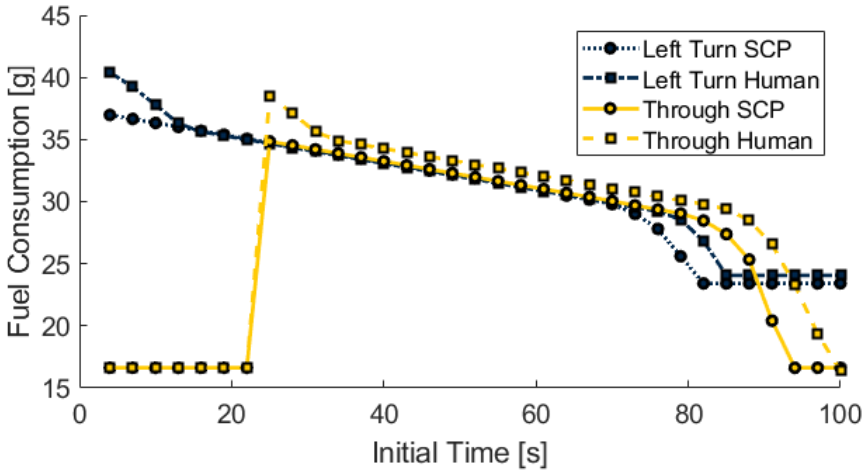


Figure 2.24 Fuel consumption for through and left turn traffic with different initial time

less. For the given traffic signal phase and timing, the average fuel reduction compared with human driver is 4.8% for through traffic and 1.9% for the left-turn traffic, while the maximum reduction is 28.6% for through movement and 12.7% for the left-turn traffic. The average travel time reduction compared with the human driver is 2.8% for through traffic and 1.8% for the left-turn traffic. Due to the speed and acceleration constraints of the left turn, the fuel and time reduction are both lower compared with the through traffic case. However, with the SPaT information, both fuel and travel time are reduced compared with the human driver baseline. The comparison of fuel consumption and travel time for one traffic signal cycle are shown in Figure 2.23 and Figure 2.24.

2.4.3 Parametric Study of the Weighting Parameters

The simulations are carried out for both single-intersection and multiple-intersection cases with a randomly generated traffic signal profile. The single intersection

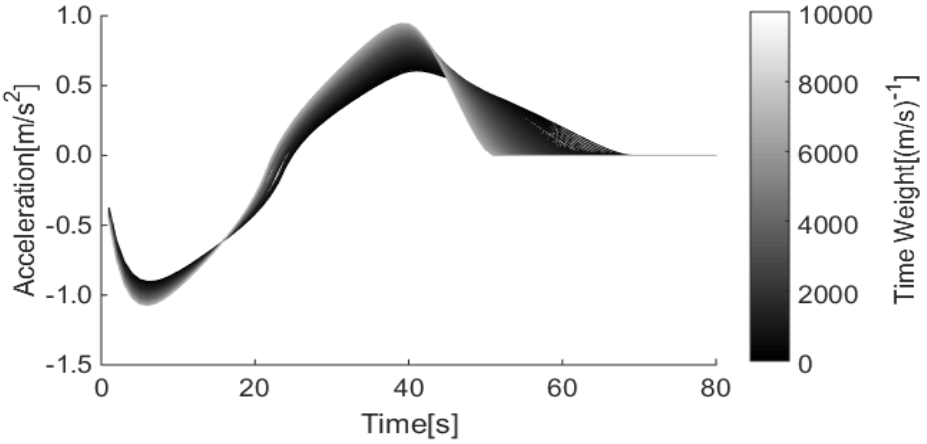


Figure 2.25 Acceleration Trajectories for Different Time Weights

case is used to demonstrate the effect on fuel consumption and acceleration, and the multiple-intersection case is used to demonstrate the influence on the intersection crossing window. The acceleration trajectories are shown in Figure 2.25, and the fuel consumption and traveling time results are shown in Figure 2.26. It can be seen from the motion trajectories that with increasing weight on travel time, more aggressive acceleration is used. Also, fuel consumption increases with higher time weighting, while travel time decreases. The trajectories for multiple-intersection cases are shown in Figure 2.27, and the corresponding fuel consumption and travel time are shown in Figure 2.28. With the

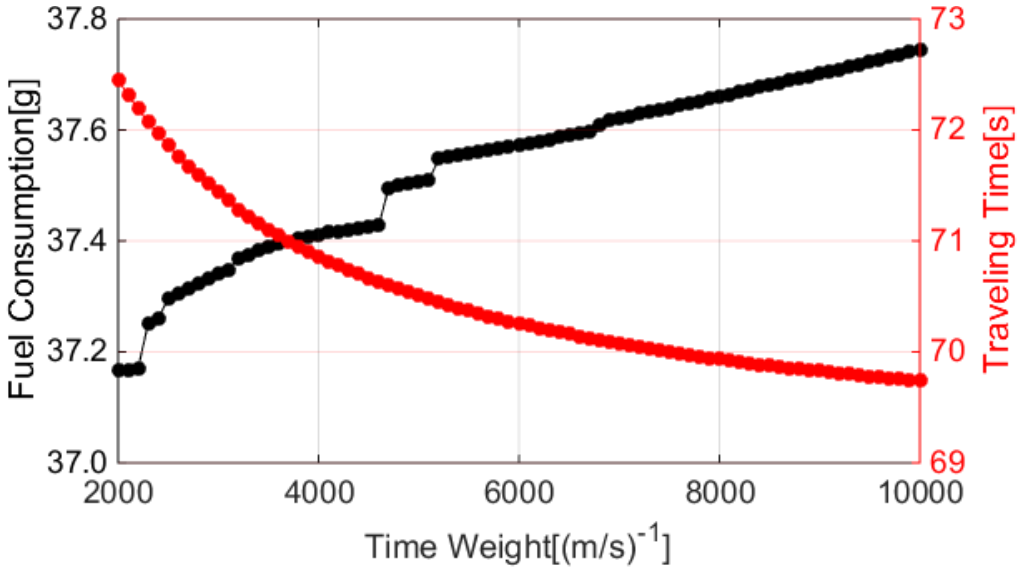


Figure 2.26 Fuel Consumption and Traveling Time for Different Time Weights

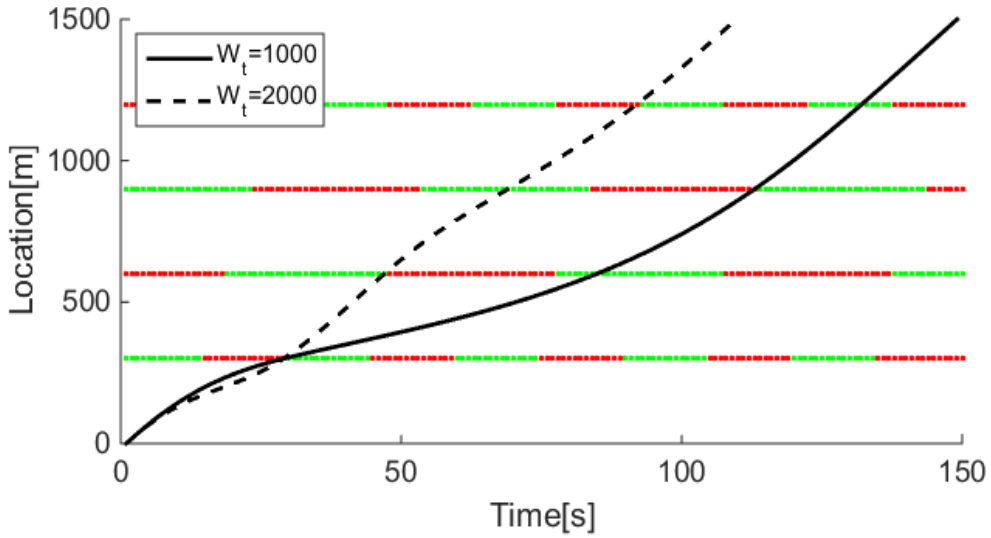


Figure 2.27 Vehicle Trajectories for Different Time Weight

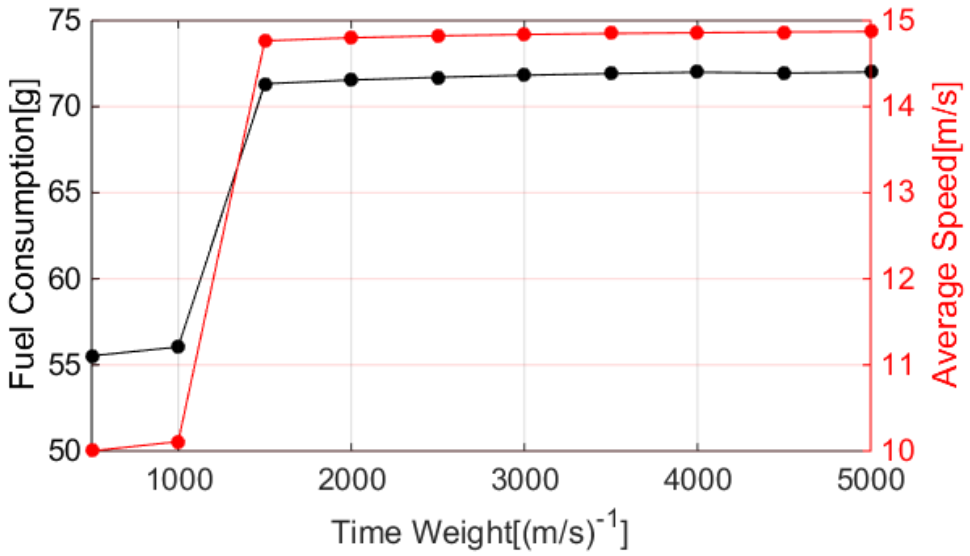


Figure 2.28 Multi-intersection time-fuel trade-off

increase in time weight, the vehicle uses more aggressive acceleration and fuel consumption increases.

2.5 Conclusion

With the information of broadcast traffic signal, a vehicle's speed trajectory can be optimized while approaching signalized intersections. We show that both fuel consumption and travel time can be reduced. The analysis assumes that no other vehicle is present. Also, the analysis is based on the connected automated vehicle assumption, which means the computed speed trajectory is followed precisely. However, a driver-assistance speed advisory would be a more practical application in the near-term. In other words, the results shown in this chapter can be treated as an optimistic upper bound for benefits that can be achieved.

CHAPTER 3
Data-Driven Fuel Consumption Model for Eco-Routing

3.1 Introduction

A fuel consumption model that can be used to compute the fuel consumption cost for the road links is the basis for eco-routing algorithm development. Such a model should (i) use credible physics-driven simulation models (such as Autonomie [55]), (ii) the driving speed should be from vehicle data which reflects the real-world operating conditions, and (iii) the model should fit the expected value from many trips instead of matching individual trips. We use the Gaussian Mixture Regression (GMR) to build our model [61]. The GMR models the joint density of model input and output, then derives the conditional expectation of the output from joint density functions of the inputs and output, thus the model is invariant under any coordinate system. The framework of our approach is summarized in Figure 1.4. After the fuel consumption model is developed, we use it to evaluate the expected fuel consumption of different routing strategies, including shortest-distance, shortest-time, eco-routing, and travel-time-constrained eco-routing. The main contributions of our work include: 1) a nonparametric data-driven fuel consumption model based on real-world driving data and Autonomie fuel consumption simulations; 2) a

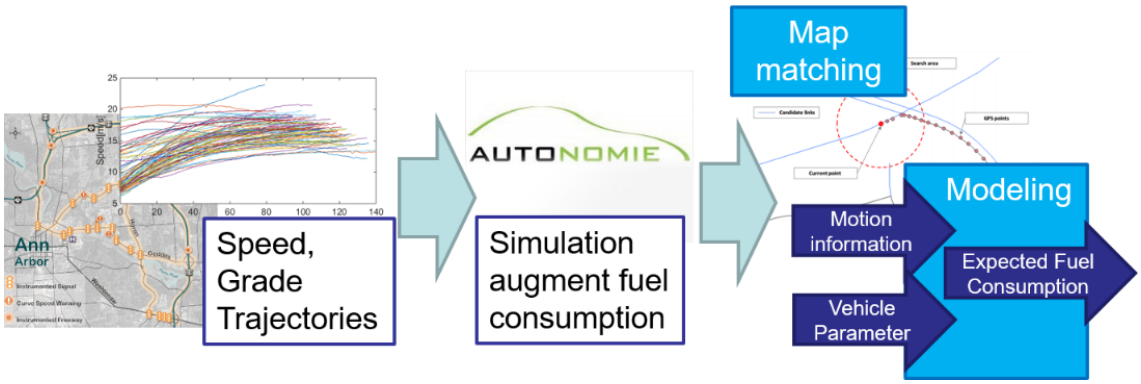


Figure 3.1 Fuel consumption modeling framework from connected vehicle data

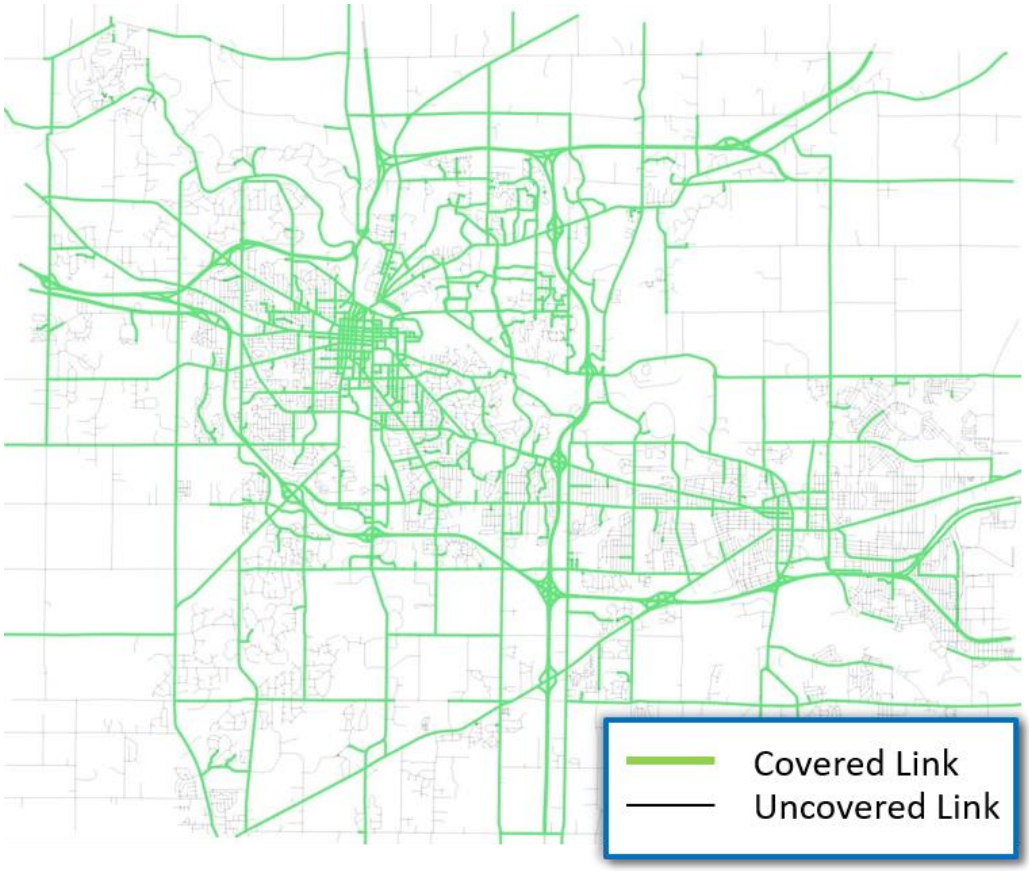


Figure 3.2 Links with more than 100 trips each from the queried data

constrained eco-routing strategy addressing the tradeoff between travel time and fuel consumption; and 3) we studied the fuel consumption and travel time trade-off of different routing strategies.

The rest of the chapter is organized as follows. The naturalistic driving data used, the Autonomie model, the Gaussian Mixture Regression model (GMR), and the constrained eco-routing method are presented in Section 3.2. Section 3.3 presents results and discussion. Conclusions and future work are given in Section 3.4.

3.2 Methodology

3.2.1 Naturalistic Driving Data

The real-world travel speed and trajectories are obtained from the Safety Pilot Model Deployment (SPMD) database [76]. The SPMD program aims to deploy and demonstrate connected vehicle technologies. It records naturalistic driving data from up to

2,842 equipped vehicles, which is about 3% of the total vehicle population in Ann Arbor, Michigan, for more than three years. As of April 2016, 56.2 million kilometers have been logged, making SPMD one of the largest naturalistic driving databases. The query criteria used for this study are as follows:

- From May 2013 to October 2013
- All passenger cars
- Trip duration longer than 10 minutes and shorter than 1 hour

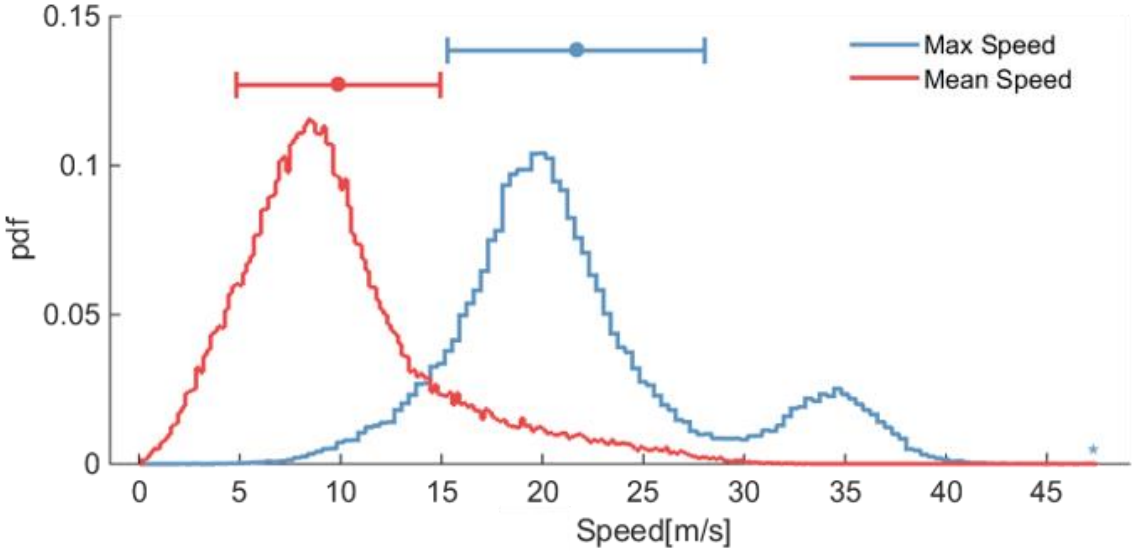


Figure 3.3 Trip Average Speed and Trip Max Speed Histogram

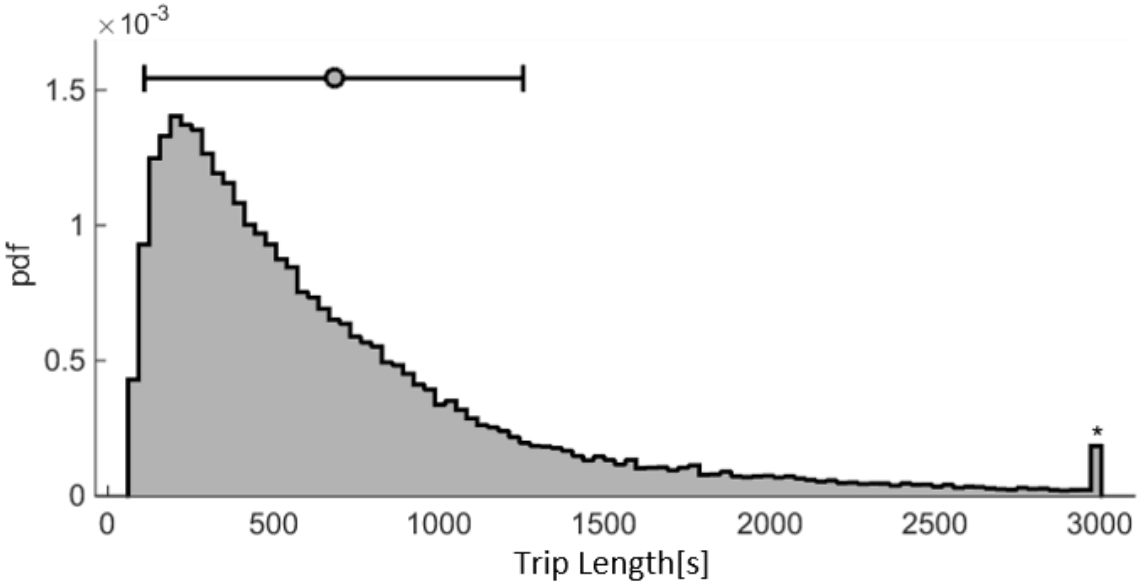


Figure 3.4 Trip Duration Histogram

- Trip distance longer than 300 meters
- Trips in the Ann Arbor area: latitude between 42.18^{o} and 42.34^{o}, and longitude between -83.85^{o} and -83.55^{o}

The queried results include 321,945 trips, with a total distance of 3.7 million kilometers and a total time of 93,926 hours from 2,468 drivers. The data covers 9,745 of the 11,506 links in the Ann Arbor area, with 5,599 links covered by more than 100 trips. The links with more than 100 trips are highlighted in Figure 3.2, which consist of major roads, minor roads, ramps, and highway sections. The trip statistics are summarized in Figure *3.3* and Figure 3.4. A clustering analysis on trip maximum speed indicates that 14.5% of the trips involve highway driving.

The speed and grade trajectories are used as the inputs to Autonomie [55], a microscopic fuel consumption model developed by the Argonne National Lab. The key vehicle parameters are listed in Table 3.1. We assume the target vehicle is a mid-sized gasoline engine vehicle. Including multiple vehicles and powertrain types may be considered in the future work.

Table 3.1 Key vehicle parameters for Autonomie simulations

Vehicle Mass [kg]	1,246
Max Engine Power [kW]	178.7
Max Engine Efficiency [%]	36
Max Engine Speed [rad/s]	628.2
Idle Engine Speed [rad/s]	62.8
Transmission Gear Number	6
Fuel Type	Gasoline

3.2.2 Data Processing

Speed trajectories and grade profiles are required to use Autonomie for fuel consumption simulations. However, the elevation measurement from the onboard GPS suffers from noise and bias. The median filter is a simple and efficient way for removing the shot noise and is widely used in image processing. In our research, a one-dimensional median filter [128] is employed.

$$\begin{cases} y(k) = \text{median of } x\left(k - \dfrac{n-1}{2} : k + \dfrac{n-1}{2}\right) \text{ if } n \text{ is odd} \\ y(k) = \text{median of } x\left(k - \dfrac{n}{2} : k + \dfrac{n}{2} - 1\right) \text{ if } n \text{ is even} \end{cases} \tag{3.1}$$

After the shot noise is removed, a simple low pass filter is applied to deal with the high-frequency noise. A third-order Butterworth digital filter is applied, and the cutoff frequency is chosen to be 0.01 Hz. Furthermore, to avoid phase distortion after IIR filtering, a zero-phase digital filtering technique is adopted by processing the input signal in the forward and reverse directions. An example filtered grade trajectory is shown in Figure 3.5. Finally, the grade estimated from all vehicles passing the link is used to estimate the mean link grade, which is used to augment the original map and estimate the grade trajectories for Autonomie simulations.

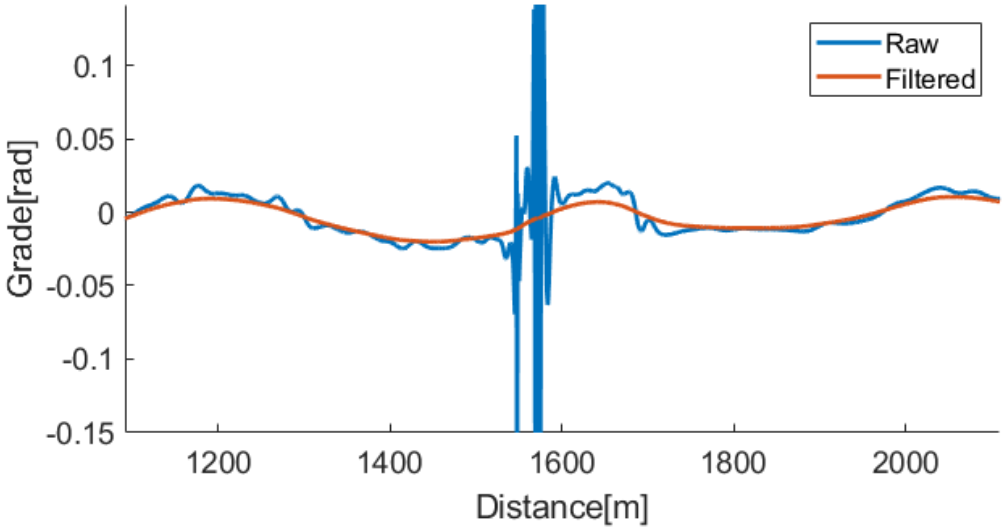


Figure 3.5 Low-pass filtering example of the trip grade data

3.2.3 Fuel Consumption Model

We use simulation output from Autonomie as the ground truth to develop our fuel consumption model, which fits the average fuel consumption of all trips on all road links in Ann Arbor. We treat the speed limit as a categorical variable and fit a distinct set of model parameters for links with different speed limits. The fuel consumption model for each speed limit category is obtained using the Gaussian Mixture Regression model (GMR) technique. Instead of modeling the regression function directly, GMR models the joint distribution of input and output variables and get the regression function through the

conditional distribution of the output as functions of the inputs. We denote the input variable as $= [x_1, \dots, x_i, \dots, x_P] \in R^{N\times P}$, where $x_i \in R^N$ are the individual input variables, *N* is the sample size, *P* is the number of input variables, and *Y* is the output variable, i.e., fuel consumption. The optimal model parameters are obtained by solving

$$\theta^* = argmin_\theta \|Y - f(x,\theta)\| \tag{3.2}$$

where $f(x,\theta)$ is the modeled regression function. The objective of the optimization problem is to minimize the norm of the regression error, which is equivalent to maximize the conditional likelihood of the output on the input variables

$$\theta^* = argmax_\theta \prod_{i=1}^{N} p(Y_i|X_i,\theta) \tag{3.3}$$

The joint distribution of input and output can be factorized as

$$P(Y,X|\theta) = P(Y|X,\theta)P(X|\theta) \tag{3.4}$$

Since $P(X|\theta)$ depends only on the input variable and thus is independent of θ, maximize the conditional likelihood of output is equivalent to maximize the joint likelihood function of the input and output.

$$\theta^* = argmax_\theta \prod_{i=1}^{N} P(Y_i,X_i|\theta) \tag{3.5}$$

In GMR, the joint distribution is modeled as a Gaussian mixture model (GMM).

$$f_{X,Y}(x,y) = \sum_{k=1}^{K} \pi_k f_{X,Y,k}(x,y) \tag{3.6}$$

$$f_{X,Y}(x,y) = \sum_{k=1}^{K} \pi_k f_{Y|X,k}(y|x) f_{X,k}(x) \tag{3.7}$$

where $f_{X,Y}(x,y)$ is the overall joint density function, π_k is the mixing coefficient for each component, $f_{X,Y,k}(x,y)$ is the joint density for each component, which follows a multivariate Gaussian distribution. For each component of GMM, the conditional distribution of output on the input still follows Gaussian distribution and can be presented in a closed-form. The marginal distribution of *X* is

$$f_X(x) = \int f_{X,Y}(x,y)dy = \sum_{k=1}^{K} \pi_k f_{X,k}(x) \tag{3.8}$$

Thus, the conditional density of output is

$$f_{Y|X}(y|x) = \sum_{k=1}^{K} w_k(x) f_{Y|X,k}(y|x) \tag{3.9}$$

where the posterior of component probability $w_k(x)$ is obtained from the marginal distribution of *X*.

$$w_k(x) = \frac{\pi_k f_{X,k}(x)}{\sum_{k=1}^{K} \pi_k f_{X,k}(x)} \tag{3.10}$$

One of the most popular approaches to obtain parameters of the GMM for the joint density is to apply the Expectation-Maximization (EM) algorithm and use the maximum likelihood method. In the Expectation (E) step, the mixing coefficient is estimated using the mean and covariance of each component by calculating the posterior; in the Maximization (M) step, the mean and covariance are estimated from the maximum likelihood method using the mixing coefficient from the E step. To apply the EM algorithm, one needs to specify the component number of the GMM, which can be achieved through cross-validation. However, since we have multiple sets of parameters due to the categorical variable (road link speed limit), specifying the component number for each speed limit through cross-validation can be time-consuming. Thus, instead of the EM algorithm, we adopt the Bayesian modeling framework, which models the parameters as latent random variables and inference the expectation of the parameters from the data [129]. Multiple approaches can be used to solve the inference problem, including Markov Chain Monte Carlo (MCMC) and Variational Inference (VI). We apply the VI approach to get the expected values of the parameters due to the large sample size. The approach is summarized as follows. Denote $\tilde{X} = [X, Y]$ as joint of input and output, $Z = \{z_{nk}\}_{N\times K}$ as the indicator variable of the component for each data point, which is a binary variable. The conditional likelihood of Z on mixing coefficient π is

$$P(Z|\pi) = \prod_{n=1}^{N} \prod_{k=1}^{K} \pi_k^{z_{nk}} \tag{3.11}$$

The parameters are modeled as random variables with their corresponding conjugate priors, i.e., Dirichlet distribution for π and Gaussian-Wishart distribution for mean and covariance.

$$P(\pi) = Dir(\pi|\alpha_0) = C(\alpha_0) \prod_{k=1}^{K} \pi_k^{\alpha_0 - 1} \tag{3.12}$$

$$P(\mu, \Sigma) = P(\mu|\Sigma)P(\Sigma) = \prod_{k=1}^{K} N(\mu_k|m_0, \beta_0\Sigma_k) W(\Sigma_k^{-1}|W_0, v_0) \tag{3.13}$$

where $\alpha_0, m_0, \beta_0, W_0, v_0$ are hyper-parameters. The hidden variables to inference include the indicator variable Z and the model parameters π, μ, Σ. The joint distribution is factorized as

$$P(\tilde{X}, Z, \pi, \mu, \Sigma) = P(\tilde{X}|Z, \pi, \mu, \Sigma)P(Z|\pi)P(\pi, \mu, \Sigma) \tag{3.14}$$

Thus, given data $\tilde{X}$, the posterior of latent variables is

$$P(Z|\pi, \tilde{X})P(\pi, \mu, \Sigma, \tilde{X}) \propto P(\tilde{X}|Z, \pi, \mu, \Sigma)P(Z|\pi)P(\pi, \mu, \Sigma) \tag{3.15}$$

The VI approach uses a tractable posterior distribution of the hidden variables to approximate the original posterior distribution and minimize the Kullback-Leibler (KL) divergence between the actual distribution and the approximated distribution or equivalently, maximize the evidence lower bound (ELBO). The approximate posterior distribution of Bayesian GMM using mean field approximation approach is

$$q(Z, \pi, \mu, \Sigma) = q(Z)q(\pi, \mu, \Sigma) \tag{3.16}$$

The approximate posterior can be obtained by solving the ELBO maximization problem. Since the sum of ELBO and KL divergence between the actual posterior and approximate posterior is the total loglikelihood of samples, maximize ELBO is equivalent to minimize KL divergence between the actual posterior and the approximate posterior.

$$\max_{q(Z), q(\pi,\mu,\Sigma)} \int q(Z)q(\pi, \mu, \Sigma) \ln \frac{P(\tilde{X}, Z, \pi, \mu, \Sigma)}{q(Z)q(\pi, \mu, \Sigma)} dZ d\pi d\mu d\Sigma \tag{3.17}$$

It can be shown [129] that the stationary point of the ELBO maximization problem under mean field approximation satisfies

$$\ln q^*(Z) = E_{\pi,\mu,\Sigma}\left(\ln p(\tilde{X}, Z, \pi, \mu, \Sigma)\right) + const \tag{3.18}$$

$$\ln q^*(\pi, \mu, \Sigma) = E_Z\left(\ln p(\tilde{X}, Z, \pi, \mu, \Sigma)\right) + const \tag{3.19}$$

From the stationary point condition, we can update $q(Z)$ and $q(\pi, \mu, \Sigma)$ alternately and iterate until convergence. The algorithm is initialized with hyperparameters of prior distributions. The approximated posterior of Z is first updated through (3.18), the mixing coefficient, mean and covariance are then obtained using the variation posterior of Z. For more details, one can refer to chapter 10 of [129]. The expectation of the mixing coefficient is

$$E(\pi_k) = \frac{\alpha_0 + N_k}{K\alpha_0 + N} \tag{3.20}$$

For a component with a small sample size, $N_k \approx 0$, if a small hyperparameter α_0 is used, as sample size approaches infinity

$$\lim_{N\to\infty} E(\pi_k) = \lim_{N\to\infty} \frac{\alpha_0 + N_k}{K\alpha_0 + N} = 0 \tag{3.21}$$

Thus, a small hyperparameter for mixing coefficient can be used to remove the redundant components. As suggested by [129], during the iterations, point estimation of mixing coefficient can be used to remove components that provide insufficient contribution to explain the data. The algorithm can start from a large initial value of component number and allow surplus components to be pruned out. In this way, we do not need to specify the component number for GMM. As the sample size increases, the influence of hyperparameters decreases. To see this, take the mixing coefficient for example, since α_0 and K are finite, as N and N_k approaches infinity, the expectation is determined by the total sample size and the sample size for each component. Thus, the algorithm is less sensitive to tuned parameters compared with other algorithms such as SVM and neural networks.

Table 3.2 Input variables for fuel consumption model

Motion Related	Average Speed [m/s]
	Speed Change [m/s]
Link Related	Average Grade [rad]
	Link Length [m]
	Posted Speed Limit [m/s]

The input variables we use for the fuel consumption model are listed in Table 3.2. We include both linear and the 2nd order terms, including cross-coupling 2nd order terms of the input variables. Since we treat the speed limit as a categorical variable, with the assumption that the posted speed limit can approximate free-flow speed, the average speed is also an indicator of the congestion status. Speed change and average grade are included to capture the kinetic and potential energy change.

3.2.4 Constrained Eco-Routing

To evaluate the benefit of eco-routing, we developed a travel-time-constrained eco-routing strategy. In this study, we define the links as nodes in a routing graph, and a directed edge connects two nodes if traveling from one link to its adjacent link is allowed. By using this definition, we can include speed change as part of the action cost to evaluate the expected fuel consumption. In this problem, we model all links as directed and do not allow U-turns. The algorithm is based on dynamic programming [130], which solves the optimization problem recursively based on the Bellman optimality principle.

$$x_i^* = \text{argmin}_{x_i \in adj_{out}(x_{i-1})} g(x_i, x_{i-1}) + f^*(x_i) \tag{3.22}$$

$$f^*(x_{i-1}) = \min_{x_i \in adj_{out}(x_{i-1})} g(x_i, x_{i-1}) + f^*(x_i) \tag{3.23}$$

$$f^*(x_d) = 0 \tag{3.24}$$

where x_i is the optimal next link location, x_{i-1} is the last link location. The next links should be in the adjacent set of the last link. $f^*(x_i)$ is the optimal value function of the next link. $g(x_i)$ is the transition cost defined as the weighted sum of travel time and fuel consumption in the travel-time-constrained eco-routing. $f^*(x_d)$ is the value function associated with the destination link, and is defined to be 0. The transition cost is defined as

$$g(x_i) = (1 - w_t)c(x_i, x_{i-1}) + w_t t(x_i) \tag{3.25}$$

where $c(x_i, x_{i-1})$ is the expected fuel consumption and $t(x_i)$ is the expected travel time for link x_i. To address the travel time constraint, a soft constraint is defined with respect to the time limit t_c. The soft constraint is achieved through a weighting parameter w_t between fuel consumption cost and travel time cost. The soft constraint is modeled with a sigmoid function as shown in Figure 3.6, where the travel time limit is calculated from

$$t_c(x_i) = (1 + \epsilon)t^*(x_i) \tag{3.26}$$

where ϵ is a constant and $t^*(x_i)$ is the travel time of the shortest time solution from the destination to link x_i. The travel time constraint indicates that we allow the travel time to increase no more than a certain percentage compared with the travel time of the fastest route. For shortest time routing and unconstrained eco-routing, w_t in (3.25) is set to be 1 and 0, respectively. For shortest distance routing, the transition cost is the link length.

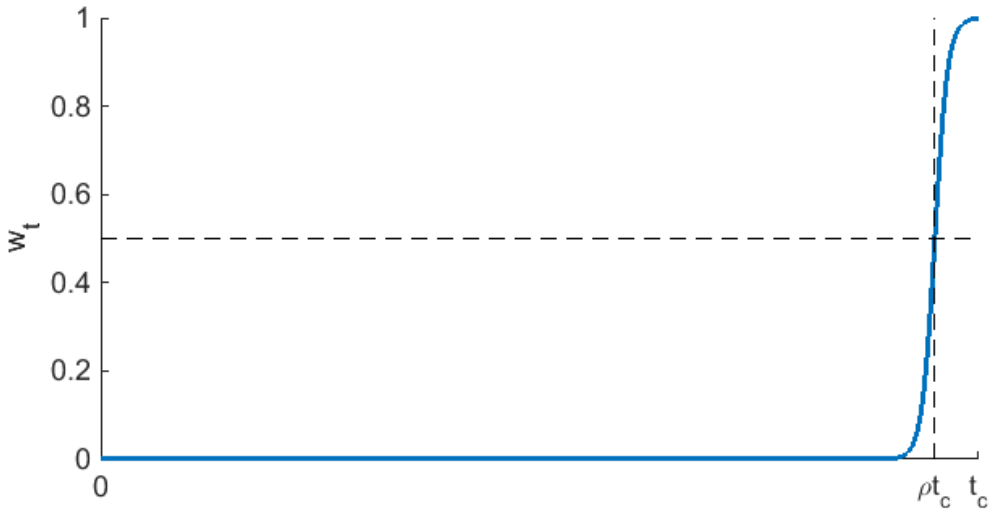
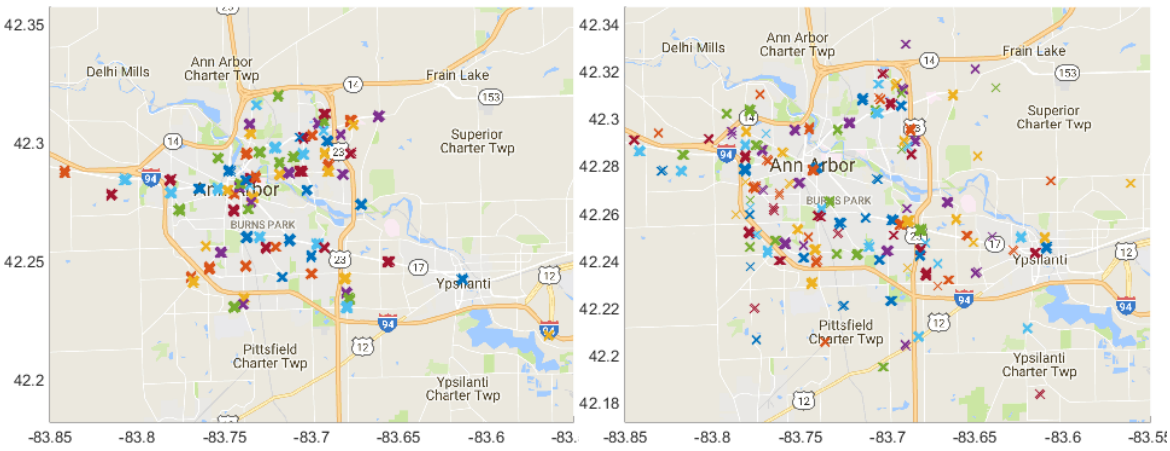


Figure 3.6 Weighting parameter for travel-time-constrained eco-routing

3.2.5 Travel Demand Location Identification

To estimate the expected fuel consumption and travel time for different routing algorithms, we use origin-destination pairs from real-world driving data. We assume that the number of vehicles using the proposed routing algorithm is small, i.e., the routed vehicles do not cause notable change to the travel speed of the links. The data to estimate travel demand is from May 2013 to October 2013, from 17:00 to 19:00 on weekdays. 25,001 trips were identified within the specified time. Since our objective is to identify frequently visited locations, the origin and destination locations are identified through a density-based cluster algorithm OPTICS [131]. The advantage of this algorithm compared with other distance-based clustering algorithms such as DBSCAN [132] is that it can cluster data with density change, which is critical in our analysis since the spatial densities of trip origin and destination locations can be affected by factors such as parking lot size. The algorithm is summarized as follows.

Given a set of points, define ϵ as the maximum distance between two points that can be considered to belong to the same cluster, and m as the minimum number of points required to form a cluster. A point p is a core point if at least m points are found within its ϵ neighborhood. For each point, the core distance d_c is defined as the minimum radius for it to be a core point of a cluster. For each pair of points, the reachability distance $d_r(p_i.p_j)$ is defined as the minimum distance from p_i to p_j if p_j is a core point. Thus, the reachability distance $d_r(p_i, p_j)$ cannot be smaller than the core distance of p_j. Once the core distance



(a) (b)

Figure 3.7 Trip locations identified with OPTICS: (a) Trip starting locations; (b) Trip ending locations

and reachability distance are identified for all points, the points are ordered by recursively selecting the point with minimum reachability distance from the current point in the unordered point set. In this way, the algorithm keeps clusters near each other in the output order. We only include trips happening at least once per week. There are 3,031 frequently visited origin-destination pairs identified, and the identified 80 starting and 123 ending locations are shown in Figure 3.7 These O-D pairs will be used later in this chapter as representative travel demands to compute the benefits of eco-routing.

3.3 Results and Discussion

3.3.1 Fuel Consumption Model

The fuel consumption model accuracy is measured using the coefficient of determination (R^2) and mean absolute percent error (MAPE). Since the objective of the model is to predict the conditional expectation of fuel consumption on motion and link variables, we compare the model output with the conditional expectation of fuel consumption given the average speed and speed change. To get the conditional expectation, we fit individual GMR for all links with more than 100 trips. Through the model of individual link, we can get the conditional expectation of fuel consumption as the complete model described in Section 3.2.3. We randomly selected 70% of the links with more than 100 events as the training dataset, and the rest as verification dataset. We use the conditional expected fuel consumption of test dataset as the ground truth. We compared

our model with several benchmarks including the average speed model [51] shown in (3.27), the power balance model which is the foundation of MOVES [56] as shown in (3.28), and the neural network model.

$$\ln\left(\frac{f}{t}\right) = \beta_0 + \beta_1 v + \beta_2 v^2 + \beta_3 v^3 + \beta_4 v^4 + \beta_5 s \tag{3.27}$$

$$f = \beta_0 vt + \beta_1 vat + \beta_2 svt + \beta_3 v^3 t \tag{3.28}$$

where f is the expected link fuel consumption, t is average link travel time, v is average link travel speed, a is average link acceleration, s is average link grade, $\beta_1, \ldots, \beta_5$ are parameters of the corresponding model. Parameters of the benchmark models are also estimated from the training dataset. For the neural network model, we used a two-layer structure with two fully connected layers, and sigmoid function as the activation function for the output of layer 1. The relative error histograms of the models are shown in Figure 3.8 and model performance are summarized in Table 3.3.

From the histogram and performance metrics, we can see that both our GMR model and the neural network model have superior performance over the other two models. Neural network models with well-tuned structure and parameters can fit the training data well. However, the main advantage of the GMR model is that it has significantly fewer parameters to be tuned compared with the neural network model. Therefore, it should be more robust compared with the neural network model.

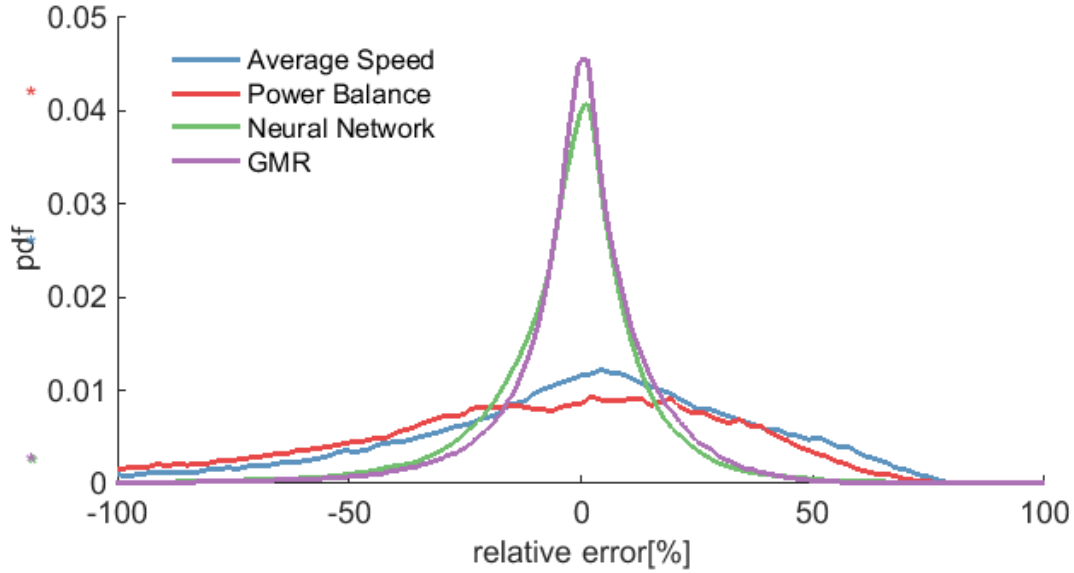
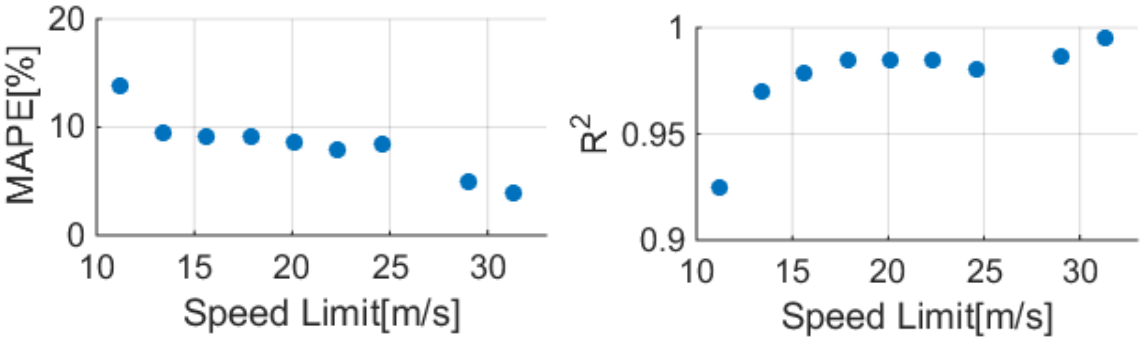


Figure 3.8 Histograms of the prediction error of the fuel consumption models

Table 3.3 Performance of the fuel consumption models

Model	R^2	MAPE [%]
Average speed model	0.77	37.63
Power balance model	0.86	46.22
Neural Network	0.98	15.60
GMR	**0.98**	**10.08**



(a) (b)

Figure 3.9 Model performance for different speed limits: (a) MAPE; (b) R^2

The GMR model performance for links with different speed limits are shown in Figure 3.9. The worst performance happens at links with speed limit 11.18 m/s (25 mph) with MAPE 13.78%. The MAPE for links with higher speed limits is less than 10%. The reason, we believe, is that links with lower speed limit contain more speed and traffic variation due to traffic signals, cross-walk, bus stops, etc. Also, at low speed and low torque, the engine fuel consumption is highly nonlinear, while for high power operation, the fuel consumption – power relation is more linear.

3.3.2 Routing Results

The routing algorithm is applied to the 3,031 identified frequent OD pairs as described in Section 3.2.5. The studied Ann Arbor traffic network consists of 21,569 directed links with variate link types including local, minor, major, collector, ramp, and highway. The computation time to solve all-to-one routing result is around 13 seconds on a computer with Intel Core i7 and 16 G RAM. Considering the requirement for the travel time of shortest-time routing, the computation time for constrained eco-routing is about 26 s. The routing costs are evaluated based on historical average speed during the studied

hours. The uncovered links are imputed with their posted speed limits. Since they are never traveled by any sample vehicles over six months, we hypothesize these links are usually not congested and the posted speed limit is a reasonable approximation for the travel speed. To get the historical average speed, we use GMM to approximate average speed distribution of individual links and estimate the posterior of mixing coefficient based on speed during the sampled hours. Samples of local and highway speed models for one road

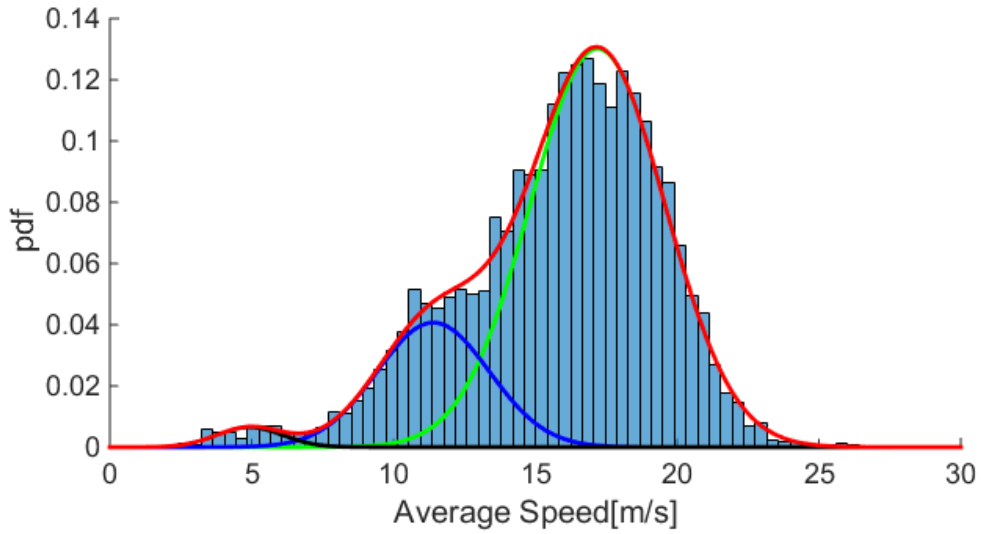


Figure 3.10 Speed histogram and GMM fitting for one local road section with a speed limit at 17.88 m/s (40 mph)

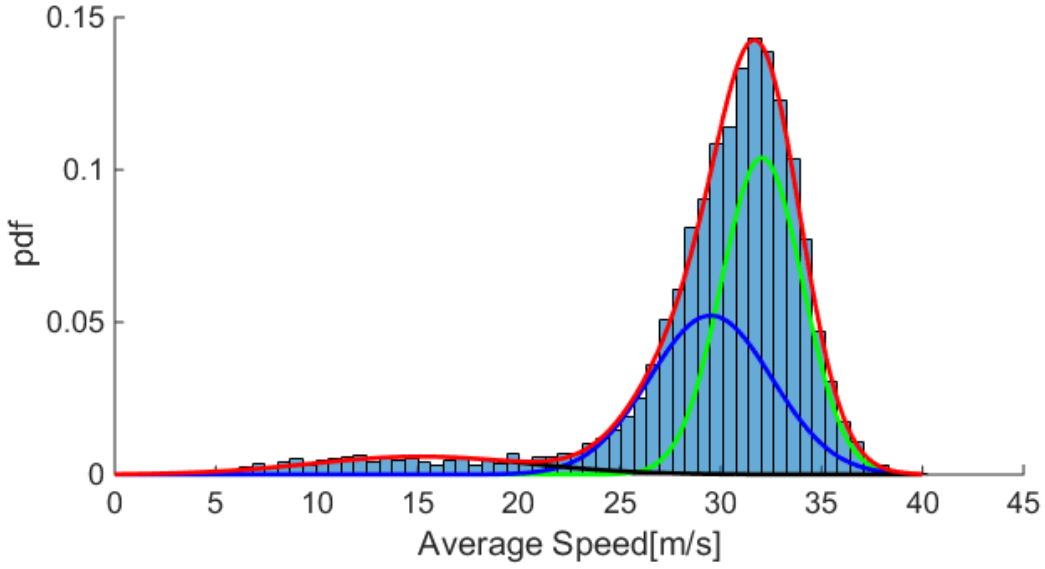


Figure 3.11 Speed histogram and GMM fitting for one highway road section with a speed limit 31.29 m/s (70 mph)

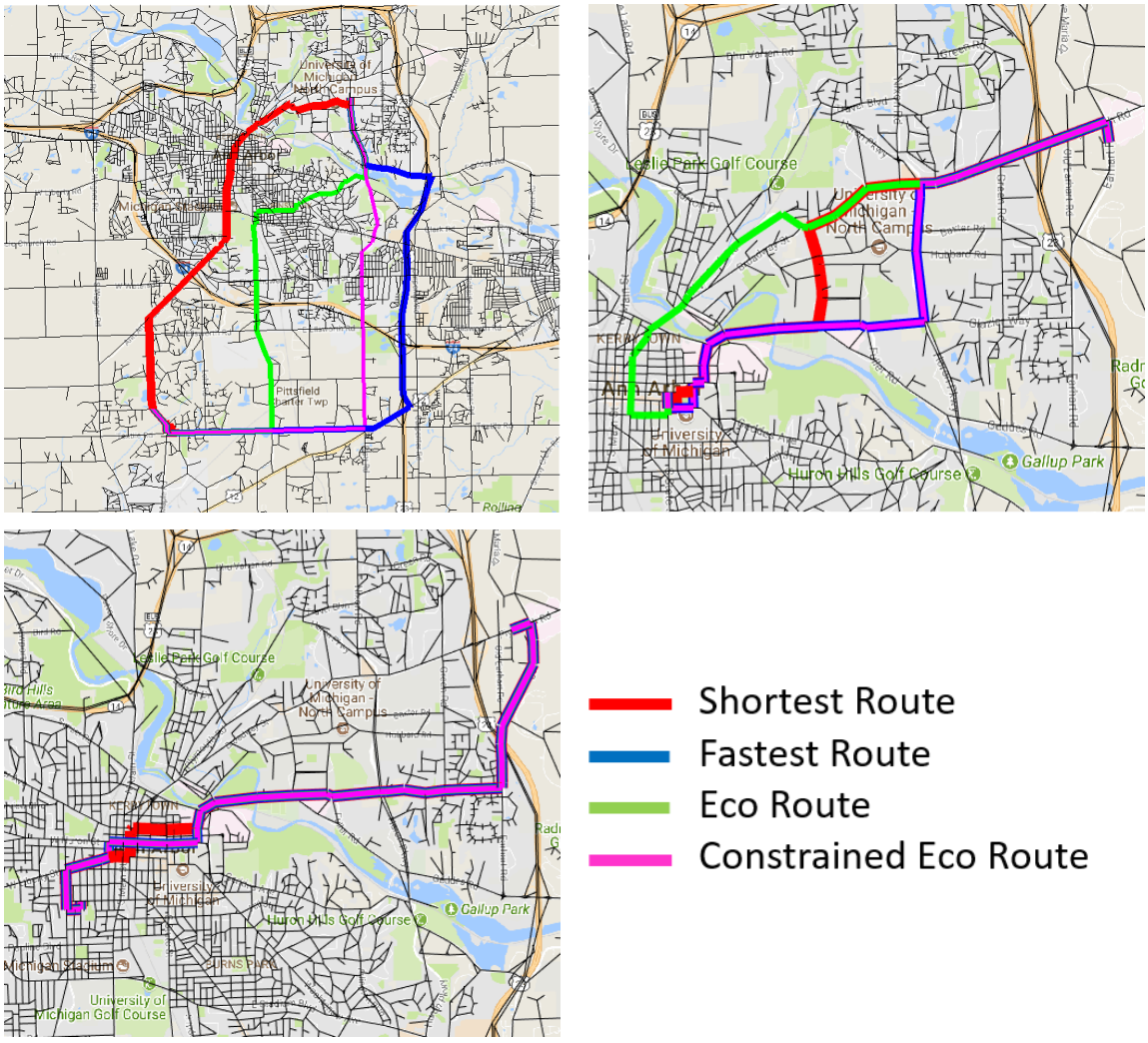


Figure 3.12 Sampled routing results

section are shown in Figure 3.10 and Figure 3.11. The expectation of travel speed is estimated using the estimated posterior of the mixing coefficient.

To compare travel time and fuel consumption of different routing strategies, travel time and fuel consumption of different strategies are normalized with the travel time of the fastest route and the fuel consumption of unconstrained eco-route, respectively. Some sampled routing results are shown in Figure 3.12. The normalized costs are shown in Figure 3.13. The scatter plot is overlaid with the expectation of cost estimated with the OD pair travel frequency. The error bars for each routing solution are 10% and 90% percentiles respectively. The expected values travel time and fuel consumption are summarized in Table 3.4.

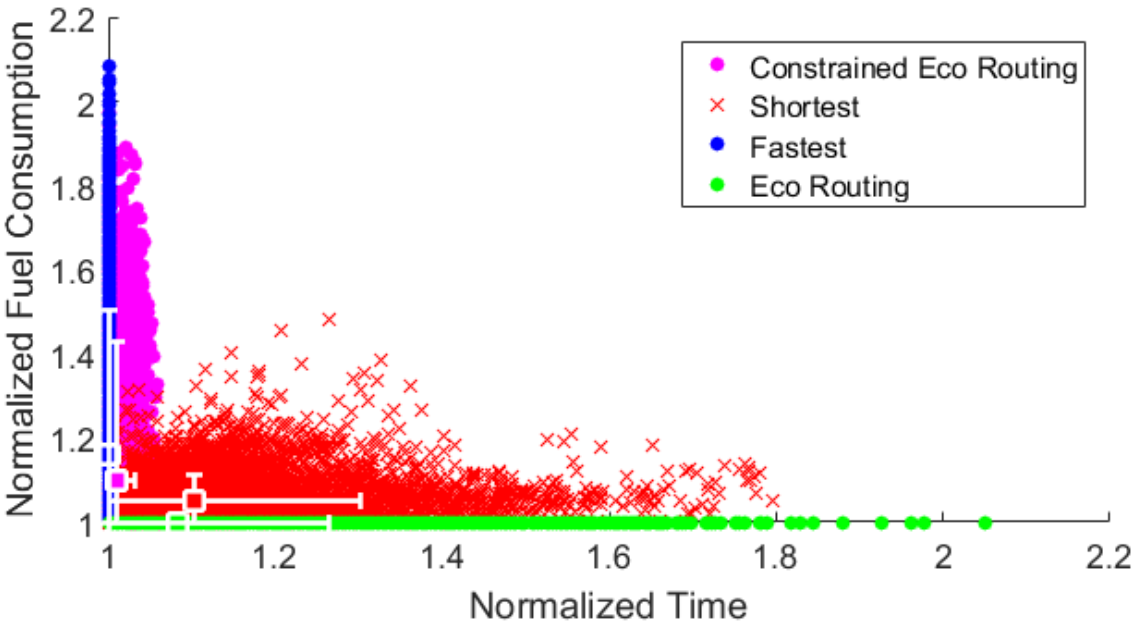


Figure 3.13 Normalized travel time and fuel consumption for different routing strategies during the evening rush hour (16:00 – 18:00)

Table 3.4 Expected travel time and fuel consumption of different routing strategies during the evening rush hour (16:00 – 18:00)

	Fuel consumption [kg]	Travel Time [s]
Shortest	0.4809	611.37
Fastest	0.5312	554.45
Eco-routing	0.4576	601.04
Constrained eco-routing	0.5038	559.49

From the results, we can see that the shortest path consumed less fuel compared with the fastest routing algorithm, while the travel time is increased significantly. Also, with a maximum of 6.48% increase in travel time, the constrained eco-routing solution has expected fuel saving of 5.16% and the maximum saving is 51.8%, compared with the fastest-path solution. It is also noted that for the given OD pairs, 28% of the eco-routing solution is identical to the fastest-path solution, and 27% is the same as the shortest-path solution. For constrained eco-routing results, 55% is the same as the fastest-route solution, and 27% is the same as the shortest-path solution. Besides that, 28% of the shortest path and fastest-path are the same. The difference between eco-routing and constrained eco-routing is due to the travel time constraints. To see the influence of traffic status on the routing results, we normalize the results of different strategies with the travel time of fastest

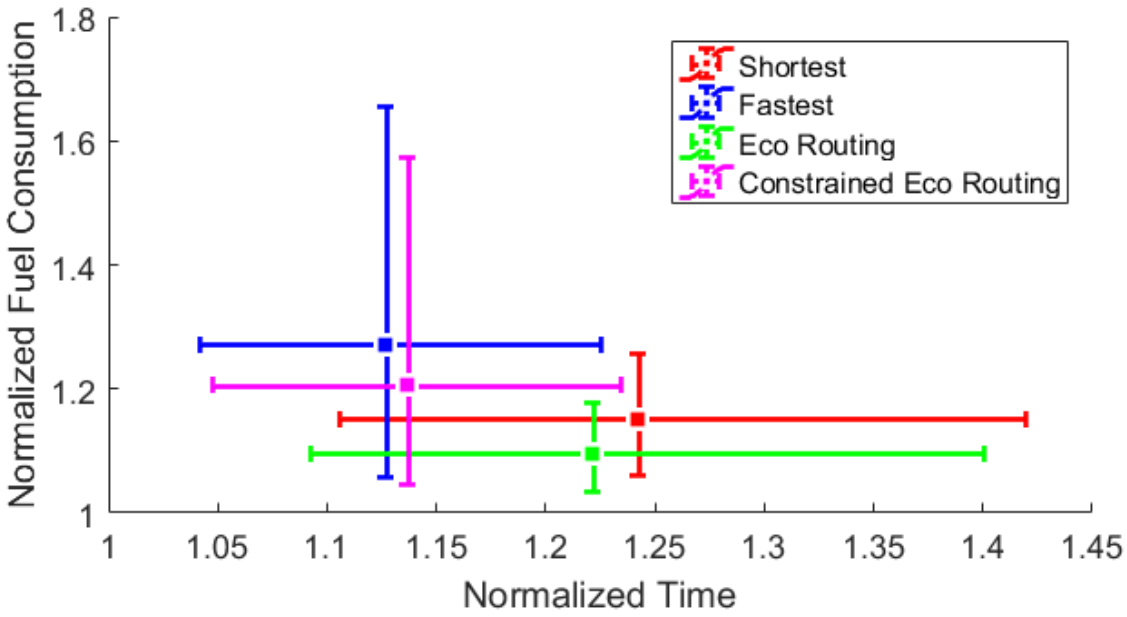


Figure 3.14 Travel time and fuel consumption obtained with historical cost normalized with results from routing results of posted speed limit

Table 3.5 Expected travel time and fuel consumption of different routing strategies under the free-flow condition (posted speed limit)

	Fuel consumption [kg]	Travel Time [s]
Shortest	0.4370	535.71
Fastest	0.4884	492.09
Eco-routing	0.4177	536.79
Constrained eco-routing	0.4711	495.09

route and the fuel consumption of the unconstrained eco-route under free traffic condition of which routing costs are estimated using the posted speed limits. The results are shown in Figure 3.14. The expected values of travel time and fuel consumption under the free-flow condition (posted speed limit) are summarized in Table 3.5.

The results show that with congestion during rush hour, travel time and fuel consumption are increased compared with the free-flow case. Shortest-path results on average take 14.12% more time and 10.04% more fuel. Fastest-path takes 12.67% more time and 8.74% more fuel (compared with non-rush hour-results). The constrained eco-routing increased fuel consumption by 6.96%, lowest compared with other routing strategies, but the travel time increases by 13.01%, which is more than the fastest-path solution, but less than the shortest-path solution. In summary, constrained eco-routing

achieves a trade-off between travel time and fuel cost. It also seems that the shortest routing is a good approximation to (unconstrained) eco-routing. If no other information is available, and when fuel consumption is the only concern, shortest-distance routing can be used.

3.4 Conclusion and future work

A nonparametric fuel consumption model is developed to estimate expected link fuel consumption conditional on prevailing trip speed and road link variables. The model parameters are estimated from a large scale connected vehicle test database with simulated fuel consumption from the Autonomie software. The model is used to calculate constrained eco-routing results, which saves 3.54% fuel while incurring 0.6% longer travel time, compared with the fastest-route solution during non-rush hours. During the rush hour, the results of the constrained eco-routing strategy are 5.16% lower fuel and 0.9% more time, compared with the fastest-route solution.

CHAPTER 4
Eco-Mobility on Demand with Ride-Sharing

4.1 Introduction

Today's Mobility-on-Demand (MOD) service matches drivers with passengers based on their distance away or time-to-pickup. When the vehicles become driverless, fuel cost becomes an important element and may be considered in vehicle-passenger pairing, as well as route choice. This is the basis of the Eco-MOD concept we are studying in this Chapter. To include fuel consumption in the objective and integrate MOD fleet control with the Eco-Routing [74] concept, we developed a fleet control algorithm based on the work in [65] with customer wait time and delay time modeled as constraints. We propose a MOD fleet control algorithm, Eco-MOD, to minimize the fleet operation cost (fuel consumption) while satisfying the customers' travel time constraints. In our numerical study, travel demands are generated by POLARIS [75], a mesoscopic agent-based transportation model developed by the Argonne National Lab. It was calibrated with data from the Safety Pilot Model Deployment (SPMD) project [76] so that it emulates the Ann Arbor vehicle trips. The calibration dataset consists of trip information collected from up to 2,800 vehicles since 2012. To evaluate the performance of Eco-MOD under realistic transportation environment, we developed a microscopic traffic simulator based on SUMO [77] and performed a case study in Ann Arbor with generated travel demand.

The main contributions of this Chapter include: 1) a MOD fleet control algorithm which minimizes fleet fuel consumption while satisfying customer travel time constraints; 2) a simulation framework for MOD system with microscopic simulation using SUMO; 3) demonstrating the importance of including fuel consumption in optimization to reduce fleet operating cost.

The rest of this chapter is organized as follows: Section 4.2 presents the trip assignment algorithm. Section 4.3 presents the simulation framework to evaluate the

performance of the MOD fleet. Section 4.4 presents the Eco-MOD framework. Section 4.5 presents two approaches to estimate the fleet size necessary for the envisioned mobility service. Section 4.6 presents the simulation results. Conclusions and future work are given in Section 4.7.

4.2 Travel Demand Assignment

Our fleet control algorithm is based on the graph decomposition method proposed in [65]. The algorithm can solve the trip matching and routing problem for ride-sharing of thousands of vehicles and customers fast enough for real-world implementation. We further improve the algorithm by considering fuel consumption as part of the fleet operation cost.

As a starting point, we reproduce the work in [65] by assuming the road network is static and solving all optimal routes considering travel time and fuel consumption offline. The trip assignment algorithm is based on a shareability graph. The graph is defined as an undirected graph with nodes defined as customers and vehicles. The constraints for each customer consist of wait time and delay time. Wait time is defined as the time between the customer travel request and pickup. Delay time is defined as the difference between planned travel time and the shortest travel time possible after pickup, which is from the minimum-time routing solution from origin to destination. An edge exists between two customers if a vehicle can depart from the origin of one of the customers and fulfill the travel demands of both customers without violating travel time constraints. An edge exists between a vehicle and a customer if the demand can be served by the vehicle without violating travel time constraints. Thus, a necessary condition for a trip to be feasible is that the customers of the trip can form a clique with a vehicle presented in the shareability network. A clique is a subgraph such that every node is connected to every other node within the same clique. It is noted that the cliques do not need to be maximum cliques in the shareability graph. The cliques in a graph can be found with the Bron-Kerbosch algorithm [133] with worst-case time complexity $O(dn3^{d/3})$ where n is the number of nodes and d is degeneracy of the graph, which is a measure of sparseness. Instead of evaluating the cost of trips for all possible combinations of customers and vehicles, one

can solve single-vehicle-multiple-customer problems for every clique, a necessary condition for a trip to be feasible.

Trip scheduling for each clique is a traveling salesman problem (TSP) with pickup and delivery. The problem can be solved with multiple algorithms. If the number of customers is small, (e.g., less than 5), the exact solution can be found by Dynamic Programming in less than 1 sec on a standard desktop computer. Heuristic-based algorithms such as T-share [134] can be used to find the solution if the problem size is large. In our study, the vehicle capacity is assumed to be 4, and Dynamic Programming is used to find the optimal solution. The states are defined as

$$\boldsymbol{\delta}_{\mathrm{t}} = \left[\delta_{1,t}^{P}, \cdots, \delta_{i,t}^{P}, \cdots, \delta_{N,t}^{P}, \delta_{1,t}^{D}, \cdots, \delta_{i,t}^{D}, \cdots, \delta_{N,t}^{D}\right] = [\boldsymbol{\delta}_{\mathrm{t}}^{\mathrm{P}}, \boldsymbol{\delta}_{\mathrm{t}}^{\mathrm{D}}] \tag{4.1}$$

where $\delta_{i,t}^{P}$ and $\delta_{i,t}^{D}$ are indicator variable for pickup location and drop-off location of customer i at step t, respectively, the value is 1 if the location has been visited and 0 otherwise. If two customers have the same pickup or drop-off locations, we assign separate variables for them but define the transitional cost as 0. N is the total number of customers in the clique. The problem is then to find the optimal trajectory to travel from the initial state, which is $\boldsymbol{\delta}_0 = \{0\}_1^{2\mathrm{N}}$, to the terminal state, which is $\boldsymbol{\delta}_{\mathrm{T}} = \{1\}_1^{2\mathrm{N}}$. The valid pickup/dropoff constraints are

$$\boldsymbol{\delta}_{\mathrm{t}}^{\mathrm{D}} - \boldsymbol{\delta}_{\mathrm{t}}^{\mathrm{P}} \geq 0, \forall \mathrm{t} \tag{4.2}$$

The constraint indicates that the drop-off locations are visited after the pickup locations of each customer. The vehicle capacity constraints are

$$\sum_{i} \delta_{i,t}^{P} - \delta_{i,t}^{D} \leq V_c, \forall \mathrm{t} \tag{4.3}$$

where V_c is the capacity of the vehicle (=4), limiting the number of onboard customers. The continuity constraint is defined as

$$\|\boldsymbol{\delta}_{\mathrm{t}+1} - \boldsymbol{\delta}_{\mathrm{t}}\| = 1, \forall \mathrm{t} \tag{4.4}$$

This constraint indicates that only one pickup/dropoff happens for each state. If the objective for fleet assignment is to minimize waiting time and delay time of customers, the transitional cost is defined as

$$g(t, t+1) = \sum_{i} T_{t,t+1}\left(\left(1 - \delta_{i,t}^{P}\right) + w_D\left(\delta_{i,t}^{P} - \delta_{i,t}^{D}\right)\right) \tag{4.5}$$

where $T_{t,t+1}$ is the travel time from the location associated with the state of t to $t+1$, w_D is the weighting parameter between waiting time and travel time. If the objective of the fleet assignment is to minimize fuel consumption of the fleet, the fuel consumption of traveling between locations associated with the states is used as the transitional cost. The objective of the traveling salesman problem is to minimize the sum of transitional cost from the initial state to the terminal state

$$J_{TSP} = \sum_{t=0}^{T-1} g(t, t+1) \tag{4.6}$$

where J_{TSP} is the objective of the TSP step of each clique. A trip is feasible if the waiting time and delay time constraints are satisfied for all customers in the clique. After all feasible trips are found through solving the scheduling problem for all cliques, the optimal trip assignment problem is formulated as a bipartite matching problem and solved through Integer Linear Programming (ILP). The cost of a trip is defined as c_t^i for trip i. In the time-minimization formulation, c_t^i is defined as the weighted sum of total wait time and delay time of customers served. In the Eco-MOD formulation, c_t^i is defined as the total trip fuel consumption. The states of the system are δ_t which is the indicator variable for trip/clique and δ_c which is the indicator variable for a customer and can be represented as a function of δ_t. At an assignment instant, if there are m feasible trips from TSP step and n customers, then $\delta_t = \{\delta_t^i \in \{0,1\}, i \in \mathbb{N}, 1 \leq i \leq m\}$ and $\delta_c = \{\delta_c^i \in \{0,1\}, i \in \mathbb{N}, 1 \leq i \leq n\}$. δ_t^i is 1 if trip i is selected and δ_c^i is 1 if customer i is assigned. The objective function is then

$$\sum_{i=1}^{m} c_t^i \delta_t^i + \sum_{i=1}^{n} D\left(1 - \delta_c^i\right) \tag{4.7}$$

where D is the penalty for unserved customers. In the original fleet control problem, a weighted sum of total wait time and delay time of each trip is used as the cost, and in the EcoMOD framework, the total fuel consumption is used as the cost. The constraint is that each vehicle can only serve one trip in each solution, i.e.

$$\sum_{i=1}^{m} a_j^i \delta_t^i \leq 1, \forall j, \tag{4.8}$$

where a_j^i is the indicator variable for vehicle j and trip i, $a_j^i = 1$ if vehicle j can serve trip i. The constraint for all customers is that a customer is either assigned or ignored

$$\sum_{i=1}^{m} b_j^i \delta_t^i + \left(1 - \delta_c^j\right) = 1, \forall j, \tag{4.9}$$

where b_j^i is the indicator variable for customer j and trip i, $b_j^i = 1$ if customer j is served by trip i. With linear constraints and the objective function, the trip assignment problem is an integer linear programming. Since all candidate trips are feasible from construction, the travel time constraints are satisfied. For online optimization, we follow the approach in [65] to keep a pool of customers, and a customer is removed from the pool if it is picked up by a vehicle or cannot be served within the time constraint. If a customer is ignored, a vehicle from the idling fleet is assigned to serve the customer with the minimum wait time as the objective. A rebalancing algorithm using MPC is presented in chapter 5. The framework to solve the fleet control problem is summarized in Figure 4.1. Gurobi [50] is

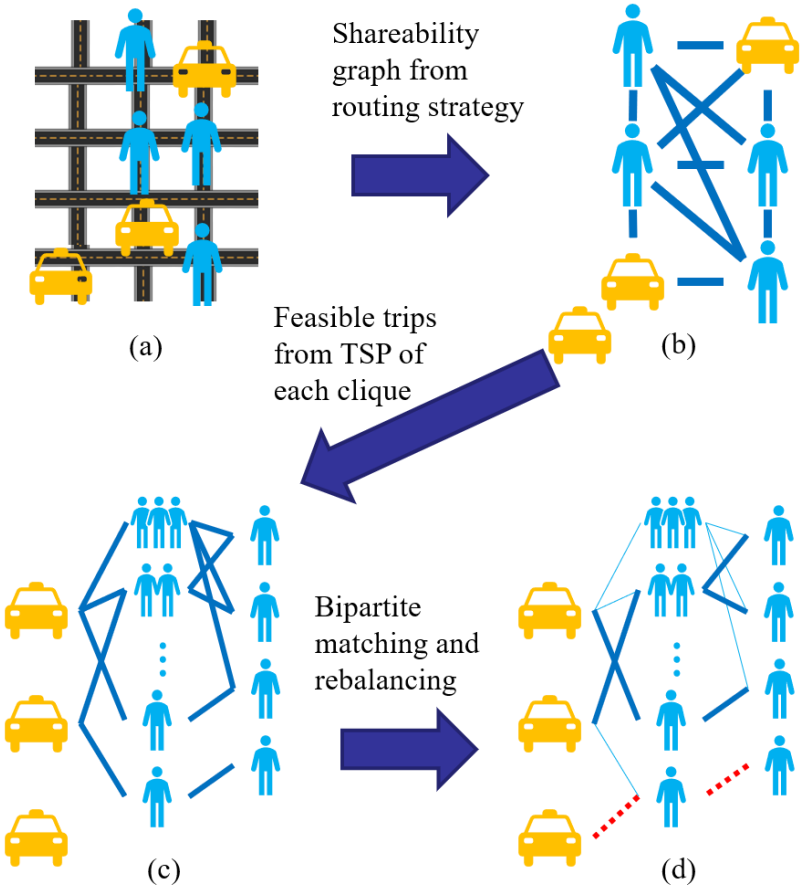


Figure 4.1 Travel demand assignment framework: (a) system receive travel demand; (b) shareability graph formulation based on routing strategy; (c) solve TSP for each clique in shareability graph to get all feasible trips; (d) assign trips to vehicles and assign ignored customers to idling vehicles for rebalancing, with thick solid line indicating feasible trip assignment and dashed line showing rebalancing assignment

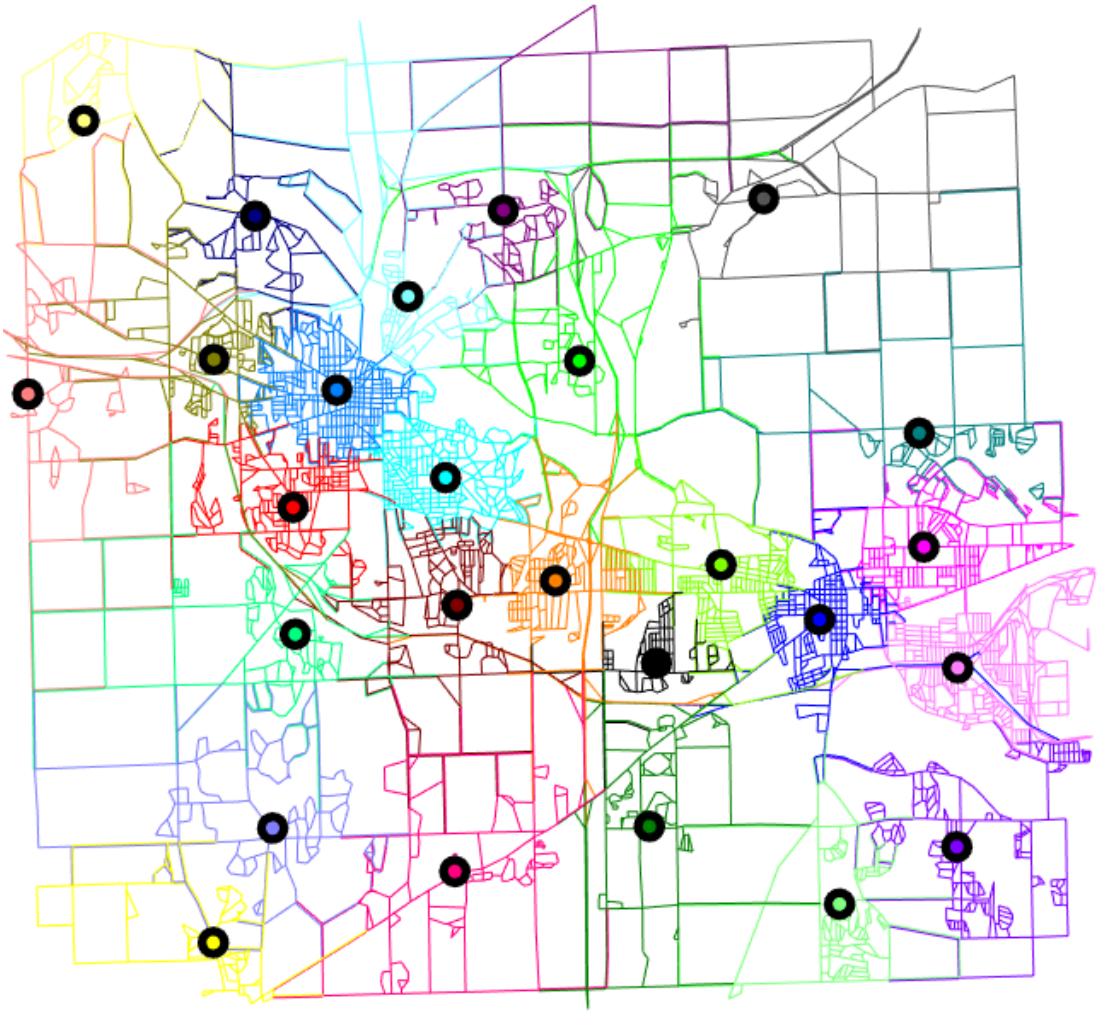


Figure 4.2 Network partition using the proposed algorithm in the approximation space for generated demand, partition centers are denoted as circles

used to find the solution to the ILP problem. The optimization problem is solved every assignment interval, reacting to new travel requests.

The road network partition results shown in Figure 4.2 from Chapter 5 is used to reduce the search space for shareability graph construction. Details of the road network partition algorithm will be discussed in Chapter 5. When constructing the customer – customer and customer – vehicle edges, we only consider customers and vehicles located in the same partition or adjacent partitions. The geometric heuristic for search space reduction is based on the construction of the partitions, where the expected travel time to the nearest partition center is minimized.

4.3 Traffic Network Simulator

4.3.1 Background

POLARIS is an agent-based software developed by the Argonne National Lab [75] focusing on travel demand and mesoscopic traffic simulations. The travel demands are generated using the ADAPTS (Agent-based Dynamic Activity Planning and Travel Scheduling) model, which formulates the activity planning of individuals as a dynamic model [135]. The demand model is calibrated by the Argonne National Lab using the dataset from the Safety Pilot Model Deployment project to simulate the city of Ann Arbor. The data is aggregated over 5 months, from May 2013 to October 2013. The data coverage from 16:00 to 16:30 on weekdays from the calibration dataset is shown in Figure 4.3. The average sample size per minute is shown in Figure 4.4. During the evening rush hour, the highway links and major links are covered by more than 80 events, thus the average speed of those links can be estimated. The estimated average speed ratio (average speed

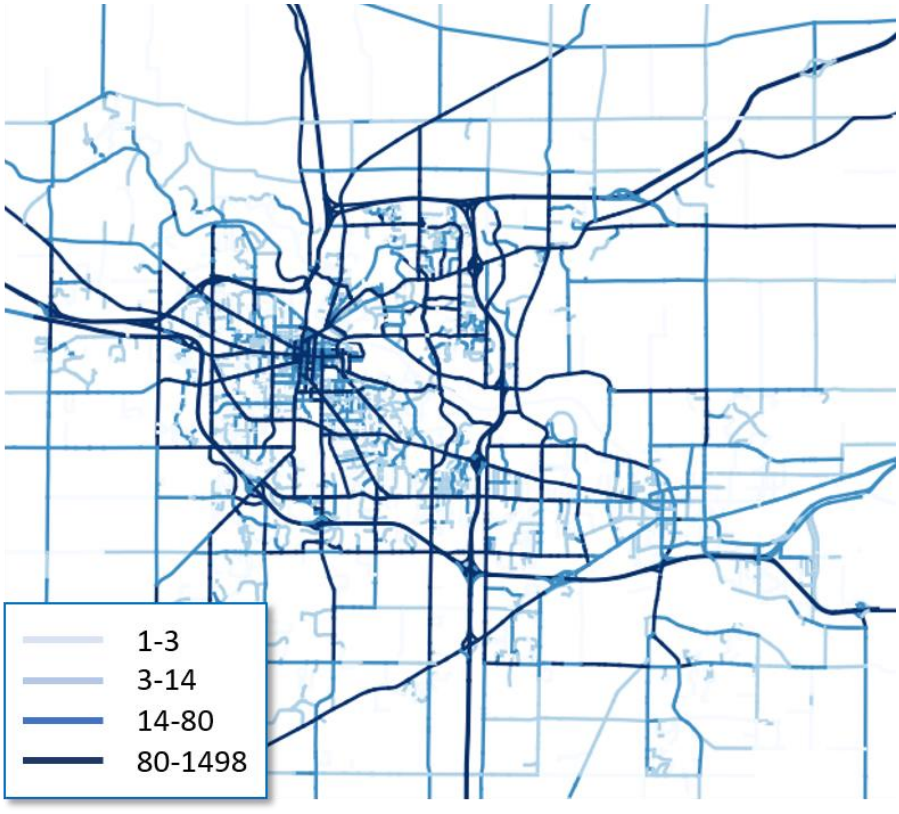


Figure 4.3 Observations per link from 16:00 to 16:30 during weekdays in calibration dataset from Safety pilot Model deployment

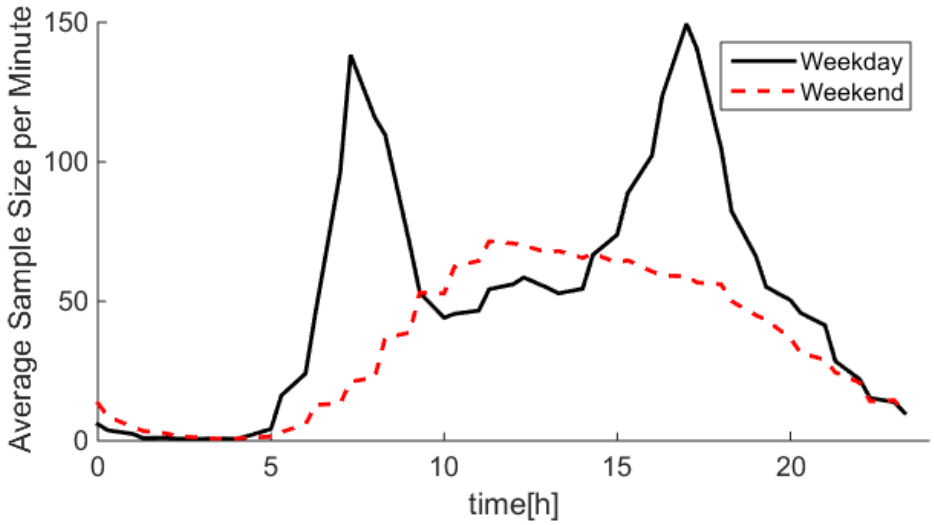


Figure 4.4 Average sample size per minute

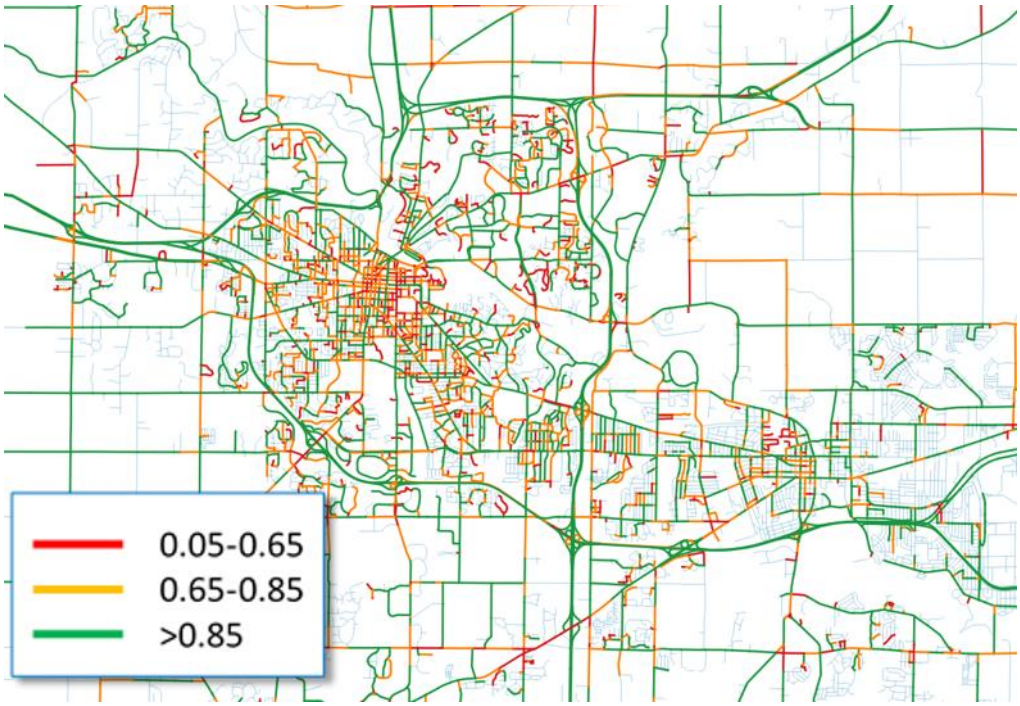


Figure 4.5 Estimated Average Speed Ratio (average speed normalized with posted speed limit) During Weekday Morning Rush Hour (7:30-8:00)

normalized against the posted speed limit) during morning rush hour (7:30 – 8:00) on weekdays is shown in Figure 4.5. The comparison between sampled calibration dataset and The POLARIS generated travel demand are shown in Figure 4.6, Figure 4.7, and Figure 4.8.

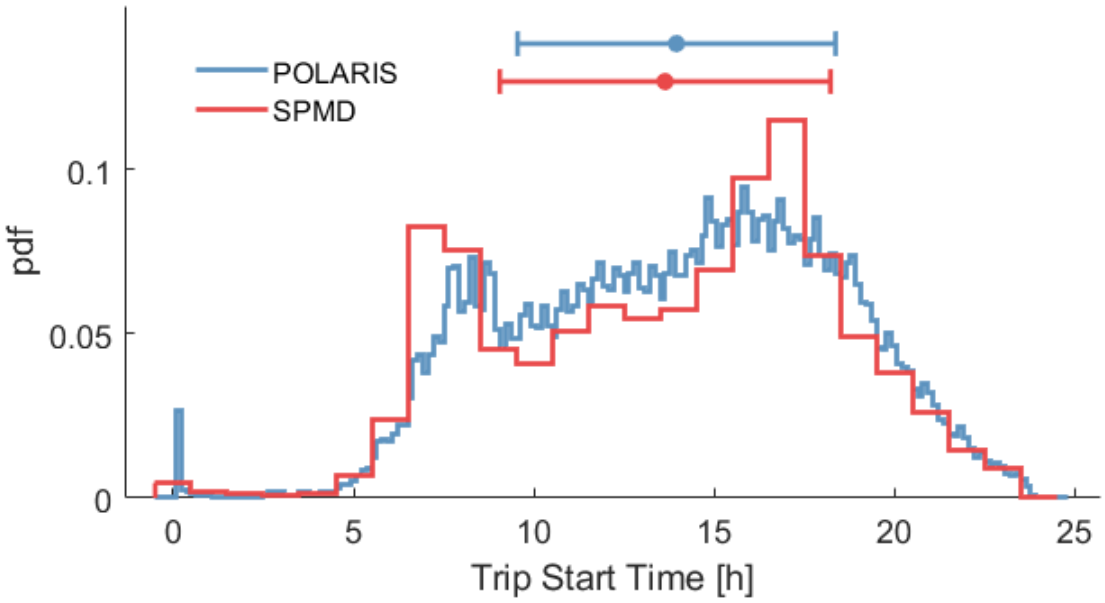


Figure 4.6 Distribution of trip start time from SPMD and POLARIS simulations

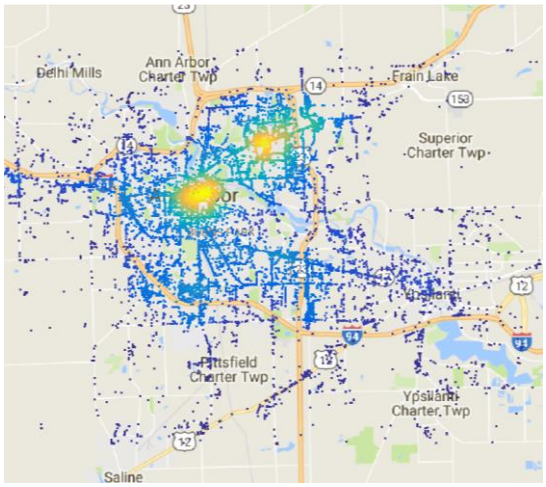


Figure 4.7 SPMD sampled initial trip location heatmap during the evening rush hour (17:00-18:00)

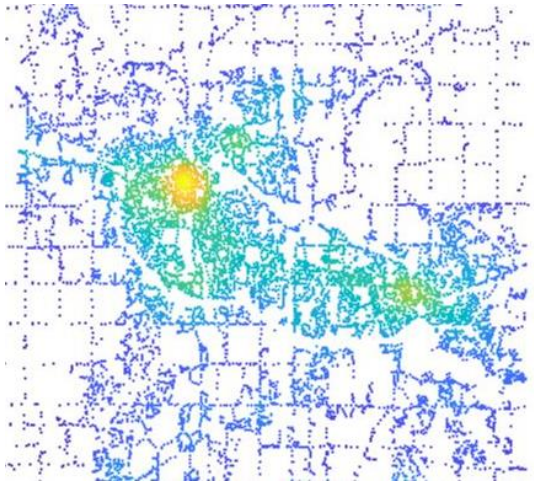


Figure 4.8 POLARIS simulated initial trip location heatmap during evening rush hour (17:00-18:00)

As a mesoscopic simulator, POLARIS's ability to simulate individual vehicles is limited. Thus, POLARIS is used as travel demand generator, with which 110,000 trips are generated from 17:00 to 19:00 and a microscopic transportation simulator, Simulation of Urban Mobility (SUMO) [77] is used for realistic verification. However, due to the difference in link models, demand generated by POLARIS cannot recreate the observed average link speed from SPMD, thus it's used as prior for demand calibration using the SPMD data.

4.3.2 Detailed verification model

SUMO is an open-source microscopic traffic simulator with the ability to generate realistic speed profile. In the simulations, the background traffic is calibrated using data from SPMD and demand generated by POLARIS is used as prior. A random subset of demands is assumed to be served by the MOD fleet. We assume the ratio of MOD customers to the total demand is fixed. The fleet size is assumed to be fixed and ride-sharing is allowed. A fleet controller based on Matlab is used to control the route choice of the MOD vehicles. The simulation framework is summarized in Figure 4.9.

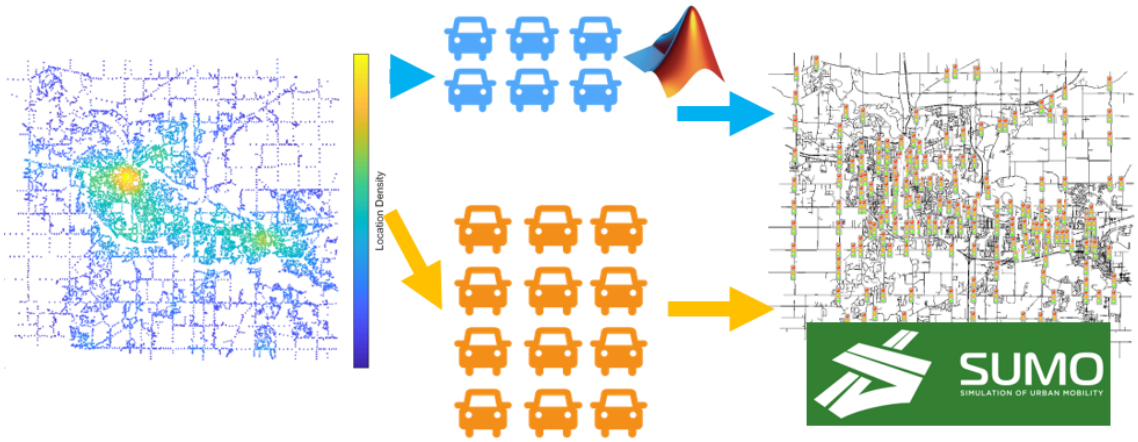


Figure 4.9 Transportation Simulation Framework

The background traffic model is calibrated using the measured average speed from SPMD. In the calibration process, we focus on route choice and travel demand distribution. Demand generated by POLARIS is used as prior for demand distribution estimation from the measured average speed. The microscopic model parameters including the car-following model and the lane-change model parameters are obtained from [136]. In the simulation framework, we only consider passenger cars. To estimate the demand distribution given the average speed measurement, we use a data-driven approach to model the relationship between the vehicle density and the average travel speed for links in SUMO, which is used to estimate the expected flow rate at each link given the measured average speed. A second-order polynomial is used when the density is below critical density for simplicity. When the vehicle density is higher than the critical density $\rho_{critical}$, we assume the average speed is constant.

$$\overline{v_n} = \begin{cases} \epsilon & \rho \geq \rho_{critical} \\ \alpha_2\rho^2 + \alpha_1\rho + \alpha_0 & otherwise \end{cases} \tag{4.10}$$

where $\overline{v_n}$ is normalized average speed, defined as average speed normalized by the free-flow speed. ρ is the vehicle density on each link, ϵ is the normalized average speed when vehicle density is higher than the critical density. Flow rate, vehicle density, and average speed are related by

$$q = N\rho\bar{v} \tag{4.11}$$

where $\bar{v}$ is average speed, q is flow rate, and N is the number of lanes. The simulated fundamental diagram and polynomial regression are shown in Figure 4.10, where simulated data are shown in the scatter plot, and the regression model is shown in the solid line. Given measured average speed from SPMD, the flow rate $\widehat{q_{SPMD}}$ is estimated.

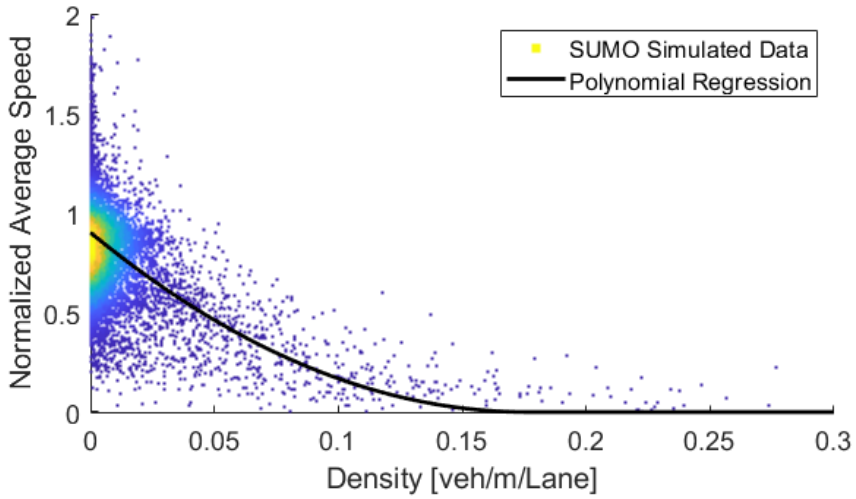


Figure 4.10 Data driven link model of SUMO

To estimate travel demand and route choice, we assume the drivers follow the shortest distance route, empirical shortest time route or real-time shortest time route. Under the assumption that the system has reached steady state, given flow rate between origin-destination pair q_{od}, the flow rate for each link is given by

$$q_l = \sum_k \left(q_{od}^{k,d} i_{od,d}^{k,l} + q_{od}^{k,t} i_{od,t}^{k,l}\right) \tag{4.12}$$

where $i_{od,d}^{k,l}$ and $i_{od,t}^{k,l}$ are indicator variables to show that link l is used by OD pair k following shortest distance route and empirical shortest time route respectively, $q_{od}^{k,d}$ and

$q_{od}^{k,t}$ are flow rate for OD pair k following the shortest distance route and empirical shortest time route, respectively.

$$q_{od}^{k} = q_{od}^{k,d} + q_{od}^{k,t} \tag{4.13}$$

The objective of calibration is to minimize the difference between the simulated flow rate and the estimated flow rate from SPMD

$$\min_{q_{od}^{k,t}, q_{od}^{k,d}} \sum_{l} \left\| q_l - \widehat{q_{l,SPMD}} \right\|^2 + \sum_{k} \psi \left\| q_{od}^{k} - q_{od,POLARIS}^{k} \right\|^2 \tag{4.14}$$

where $\widehat{q_{l,SPMD}}$ is the estimated link flow rate from SPMD, $q_{od,POLARIS}^{k}$ is the generated OD flow rate from POLARIS. ψ is the weighting parameter between flow rate approximation and the regularization term using POLARIS. Given the assumption that the OD flow rate follows Gaussian distribution, the objective function is equivalent to the maximum-a-posterior estimation of OD flow rate using POLARIS OD flow rate as prior. Assuming the total flow rate follows the total flow rate generated by POLARIS, we have the constraint

$$\sum_{k} q_{od}^{k} = \sum_{k} q_{od,POLARIS}^{k} \tag{4.15}$$

The objective function is quadratic in OD flow rate and the constraints are linear, thus the optimization problem is convex. The optimization problem is solved using Gurobi. The shortest distance route and the empirical shortest time route are generated offline, and the ratio of drivers following shortest distance in each OD flow is obtained by solving (4.14) and the ratio of drivers follow shortest time routes is estimated by simulation. To generate the empirical shortest time route, we use the measured average speed, and assume vehicle on links with not enough data follows the posted speed limit. We assume that drivers follow real-time shortest time routes are uniformly distributed in the road network. Also, we assume that the average speed on each link is normally distributed. The real-time routing ratio with the maximum likelihood of average speed is selected as the optimum value. If variances of average speed distribution are equal for all links in the network, this is equivalent to minimize the squared error between simulated and measured mean value of average speed. The average speed distribution and the fitted Gaussian distribution of 6 selected links with the sample size larger than 800 are shown in Figure 4.11.

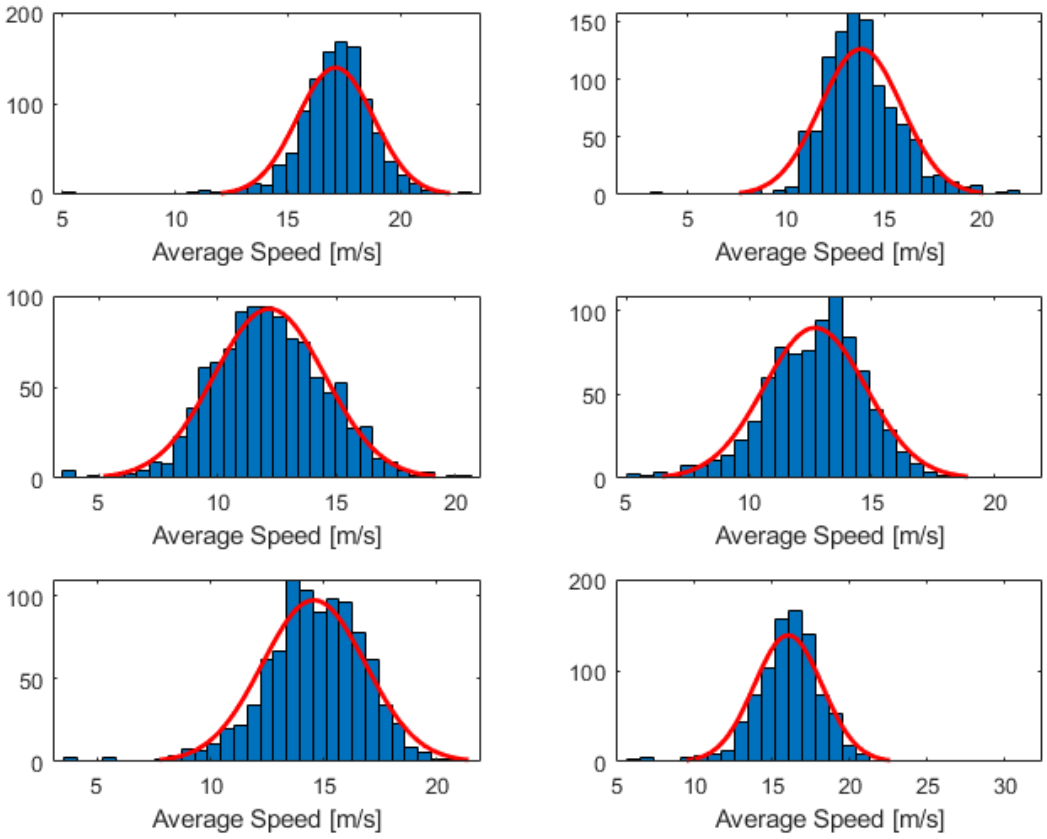


Figure 4.11 Average speed distribution of 6 selected links

4.4 Eco-MOD

Two levels of strategies can be used by the MOD fleet to reduce fuel consumption. On the trip assignment level, the objective function for fleet assignment of feasible trips can be the total fleet fuel consumption instead of the sum of individual's wait time and delay time as defined in the original fleet assignment problem. However, for assignment of the rebalance fleet, where the main objective is to serve the customers whose travel demand cannot be satisfied within the travel time constraint, we use their travel time as the objective function when assigning idling vehicles on the rebalancing trips. At the trip execution level, the routing strategy can be either shortest-time routing or eco-routing, and the corresponding routing cost is applied for trip assignment. Nine test configurations are defined based on different combinations of cost function and routing strategy to assess the fuel-saving benefit from the two levels. In all configurations, the rebalancing trips are assigned to minimize the travel time under the corresponding routing policy. The

configurations are summarized in Table 4.1, where the assignment of feasible trips is denoted as assignment, and the assignment of reactive rebalance trips is denoted as rebalance. Configuration 9 is the baseline where personal vehicles are used.

As shown in Table 4.1, configurations 1-4 minimize trip time, which is defined as the sum of wait time and delay time using different routing strategies, and configurations 5-8 minimize total fuel consumption of the fleet. The travel time requirement of customers are addressed as constraints and are satisfied by the graph decomposition based formulation. Configuration 8 is fuel-consumption-oriented, which consumes the least amount of fuel but result in the longest travel time. Configuration 1 is travel-time-oriented, which has the least travel time but consumes the most fuel. The configurations are compared with the baseline that personal vehicles are used for the trip. Routing strategies of configuration 2 and 6 are determined by the occupancy of the vehicles. If the vehicle is occupied, then the shortest time route is used. Otherwise, the eco-route is used. The baseline is configuration 9, for which case personal vehicles are used for the trips. The route choice is determined by the calibration of traffic simulator in the previous section.

Table 4.1 MOD Fleet Assignment Strategy Configuration Summary

	Assignment Cost	Assignment Routing Strategy	Rebalance Routing Strategy
1	Customer Trip Time	Shortest Time Routing	Shortest Time Routing
2	Customer Trip Time	Shortest Time / Eco Routing	Shortest Time / Eco Routing
3	Customer Trip Time	Eco Routing	Shortest Time Routing
4	Customer Trip Time	Eco Routing	Eco Routing
5	Fleet Fuel	Shortest Time Routing	Shortest Time Routing
6	Fleet Fuel	Shortest Time / Eco Routing	Shortest Time / Eco Routing
7	Fleet Fuel	Eco Routing	Shortest Time Routing
8	Fleet Fuel	Eco Routing	Eco Routing
9	-	Shorest Distance/Time	-

4.5 Fleet Size Estimation

To estimate the size of the fleet required to serve the travel demands, we apply the distance-based approach from [137] and the queuing network approach from [138]. When

using the methods to estimate fleet size, we assume that each vehicle can only serve one customer. Thus the upper bound of fleet size is estimated for each method. However, this does not indicate that all travel demands can be served within their time constraints using the algorithm described in section 4.2. In the demand assignment step, idling vehicles are sent to serve customers whose time constraints cannot be satisfied by the assignment trips, while [138] assumes customers cannot be served will leave the system instead of waiting for available vehicles and [137] does not consider travel time. Thus, a parametric study is performed to analyze the influence of fleet size on system performance. In this section, we briefly introduce the fleet size estimation methods from [137] and [138].

4.5.1 Distance-Based Approach

Given the average trip distance, average travel speed and trip generation rate, the fleet size can be estimated by

$$F = \sum_{O,D} \frac{\lambda_{OD}\left(d^{OD} + D(f^D, f^O)\right)}{v_{OD}} \tag{4.16}$$

where d^{OD} is average trip distance, λ_{OD} is the generation rate of trip served by the fleet traveling from O to D, v_{OD} is average travel speed of the fleet, $D(f^D, f^O)$ is the travel distance corresponding to rebalancing flow from O to D. Given partitioned network defined in Chapter 5, the rebalancing flow estimation problem can be formulated into a minimum cost flow problem, where sources and sinks are partitions where the density of destination distribution is larger and smaller than origin distribution respectively. The objective function is to minimize the total cost for rebalancing flow

$$\min_{\mathrm{r_{ij}}} \sum_{ij} c_{ij} q_{ij} \tag{4.17}$$

where q_{ij} is the flow rate on edge ij and c_{ij} is the corresponding cost. The flow needs to meet the flow conservation constraint

$$\sum_{j} q_{ij} - \sum_{k} q_{ki} = d_i - o_i \tag{4.18}$$

where q_{ij} and q_{ki} are the flow out of and into partition i, respectively. d_i is the density value of destination distribution and o_i is the density value of origin distribution at partition i. Given the rebalancing flow on each edge, the expected travel distance can be estimated.

4.5.2 Queuing Network Based Approach

The distance-based approach can provide a simple estimation of the fleet size that can cover the travel distance of trips, but the wait time can be too long. Thus to estimate fleet size considering the availability of the vehicle in each partition, we apply the queuing network approach from [138]. Similar to the distance-based approach, each vehicle is assumed to serve only one customer. By assuming customers leave the system if the mobility needs cannot be satisfied upon entering the system, the MOD system is modeled as a closed Jackson network with respect to the vehicles. Each partition is modeled as a single-server (SS) node, and traveling between partitions are modeled as infinite-server (IS) nodes. For each partition, the customers are assumed to enter the system follow the Poisson process with the entering rate λ_k for partition k, thus the service rate for the SS node k is λ_k. The IS nodes models traveling between partitions, and each IS node corresponds to an edge connecting two SS nodes (partitions). The service rate λ_i of IS node i corresponds to the travel time between partitions, which we assume is independent of the number of vehicles at the node if the fleet size is small. The vehicles form a queue at each SS node while waiting for customers and are serviced when a customer arrives. The vehicle then moves from the origin SS node to the IS node connecting the origin and destination. After spending the corresponding travel time in the IS node, the vehicle is then moved to the destination SS node. Given fleet size F, the state space of the queuing network model is defined as

$$\mathcal{S} = \left\{(n_1, \dots, n_N): \sum_{i=1}^{N} n_i = F, n_i \geq 0\right\} \tag{4.19}$$

where n_i is the number of vehicles at node i. Given the routing probability r_{ij} from node i to node j, the closed queuing network satisfies the balance condition

$$\pi_i = \sum_j \pi_j r_{ji} \tag{4.20}$$

where π_i is the expected number of vehicles at node i, which is also known as the throughput. The stationary density function of the system follows the product form expression under the small fleet assumption

$$p(n_1, \dots, n_N) = \frac{1}{G(F)} \prod_{i=1}^{N} \left(\frac{\pi_i}{\lambda_i}\right)^{n_i} \tag{4.21}$$

where $G(F)$ is a normalization constant. The availability of node i is defined as the probability that the queue length at SS node i is larger than 0 [139]

$$A_i(F) = 1 - P(Q_i = 0) = \frac{\pi_i G(F-1)}{\lambda_i G(F)} \tag{4.22}$$

where Q_i is the queue length at node i. The availability for each SS node is solved using mean value analysis (MVA) following [140], which can be used to estimate the fleet size without solving the normalization constant explicitly. MVA is an iterative algorithm to calculate the mean wait time $W_i(F)$ and the mean queue length $L_i(F)$ of each node. Under the closed network formulation, the initial condition given by $W_i(0) = 0, L_i(0) = 0, \forall i$ since there is no vehicle in the network. Then for each fleet size F, the mean wait time and queue length for each node is calculated by

$$W_i(F) = \begin{cases} 1/\lambda_i & i \in IS \\ \frac{1}{\lambda_i}\big(1 + L_i(F-1)\big) & i \in SS \end{cases} \tag{4.23}$$

$$L_i(F) = \frac{F\pi_i W_i(F)}{\sum_{j=1}^{N} \pi_j W_j(F)} \tag{4.24}$$

Given queue length and expected wait time, [140] showed that the availability could be estimated by

$$A_i(F) = \frac{L_i(F)}{\lambda_i W_i(F)} \tag{4.25}$$

At steady state, the rebalancing flow is estimated by solving the minimum cost flow problem from the previous section. The updated service rate at SS node i is now the sum of customer arrival rate λ_i and rebalance vehicle departure rate $\sum_j q_{ij}$. However, due to the assumptions and approximation required by the approach, the system performance is evaluated with simulation.

4.6 Results and Discussion

In the following sections, numerical simulation results from the detailed verification model are presented. First, we demonstrate that our calibrated simulator can

recreate average speed during the evening rush hour of Ann Arbor, and then use the model to estimate the effect of eco-MOD at the city-scale. Then the fleet size required to serve 4% of the total travel demands for Ann Arbor during 17:00 to 19:00 is estimated using the distance-based approach and the queuing network-based approach. Due to the approximations made by the models, a parametric study with respect to fleet size is performed using the calibrated traffic simulator to evaluate the system performance. Finally, the numerical results of eco-MOD using configurations from section 4.4 are presented.

4.6.1 Traffic Simulator Calibration

Assuming microscopic driving behavior by using parameters from [136], the demand distribution and route choice are calibrated using data from SPMD. Links with more than 80 events are used for calibration. The marginal distribution of origins and destinations are shown in Figure 4.12 and Figure 4.13, with high density indicated by yellow and low density indicated by blue. Measured and simulated average speed normalized using posted speed limit from 17:00 to 17:30 are shown in Figure 4.14 and Figure 4.15 respectively, with low speed indicated by red and fast indicated by green, and links without enough data are shown in light gray. The relative error distribution is shown in Figure 4.16, with mean relative error equals -1% and the standard deviation equals 25%. As shown in Figure 4.14 and Figure 4.15, the simulation results show less congestion in

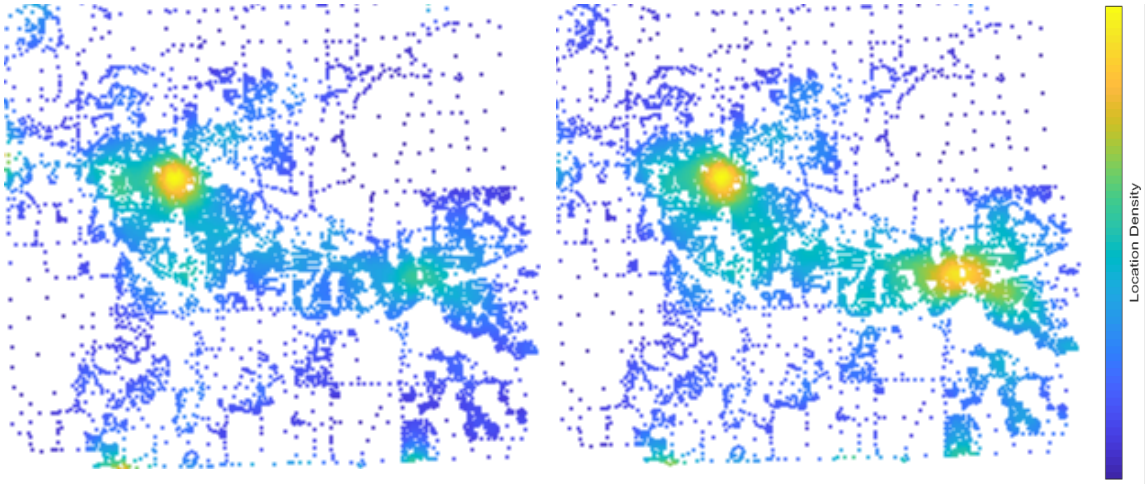


Figure 4.12 Marginal distribution of generated trip origins during weekday evening rush hour

Figure 4.13 Marginal distribution of generated trip destinations during weekday evening rush hour

the downtown area, possibly due to our assumption that the flow is only generated by passenger cars, thus our ability to simulate pedestrians and public transits in the downtown is limited. As a result, the extended stops due to pedestrian crossings and slow down due to bus stops are not captured in our model. However, developing a detailed high fidelity

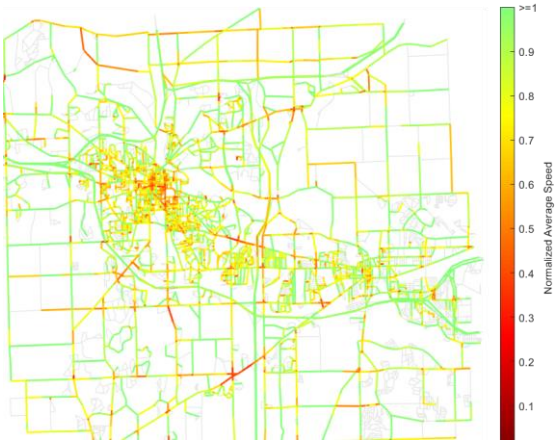


Figure 4.14 Measured normalized average speed from 17:00 to 17:30

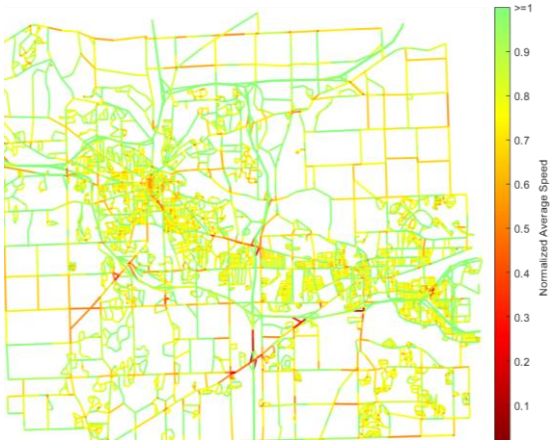


Figure 4.15 Simulated normalized average speed from 17:00 to 17:30

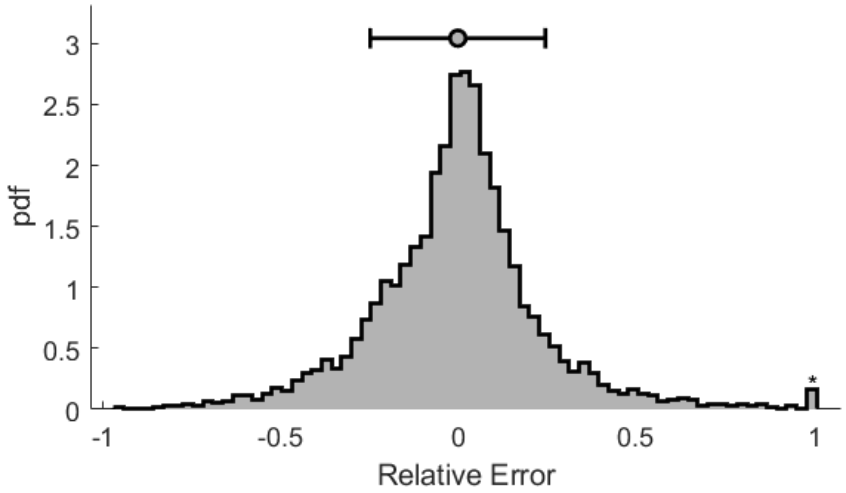


Figure 4.16 SUMO simulation average speed relative error distribution

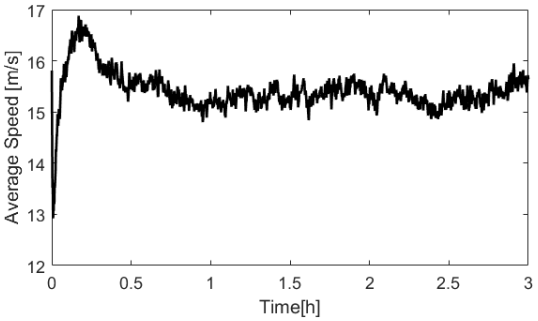


Figure 4.17 SUMO simulated network average speed

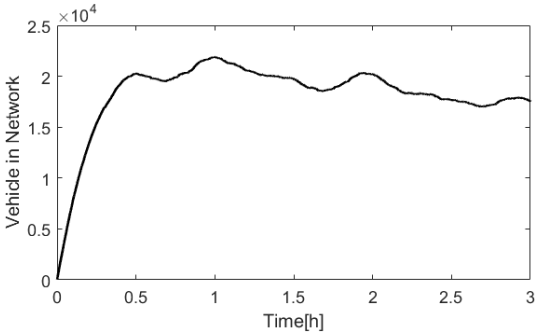


Figure 4.18 SUMO simulated active vehicle amount in network

traffic simulator is out of the scope of this study and is left for future work. In the simulation, 150,457 trips are generated from 17:00 to 19:00, following an average generation rate of 20.89 trips per second.

When using the simulator to evaluate the Eco-MOD framework, we simulate 16:00 to 19:00. Only background traffic is generated in the first hour to reach the steady-state of the traffic network. MOD fleet starts to be deployed in the second hour to reach the steady-state of service fleet. Data from the third hour is used to evaluate the efficiency of the system. Average speed and the total number of running vehicles of background traffic simulation are shown in Figure 4.17 and Figure 4.18, respectively. As shown in the figures, the system reaches the steady-state within the first hour, and the standard deviation of average speed is 0.18 m/s at steady state.

4.6.2 Fleet Size Estimation

The distance-based approach and queuing network-based approach are used to estimate fleet size required to serve 4% of the total travel demand in Ann Arbor during the

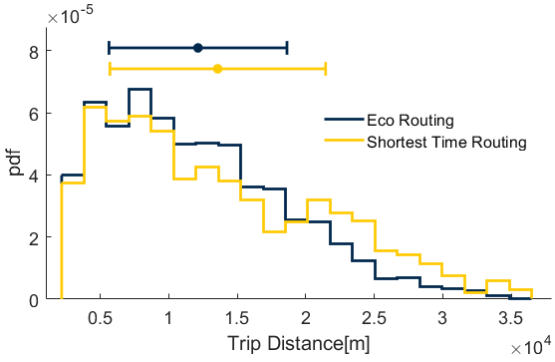


Figure 4.19 Generated Trip Travel Distance Distribution

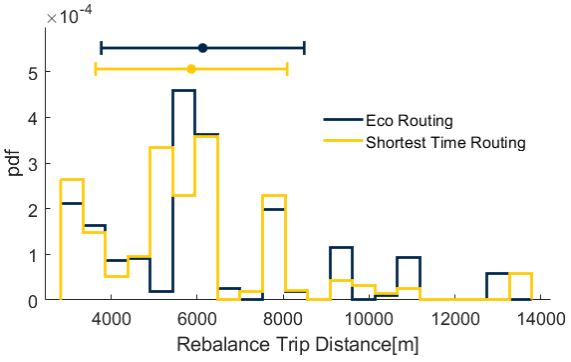


Figure 4.20 Rebalance Trip Travel Distance Distribution

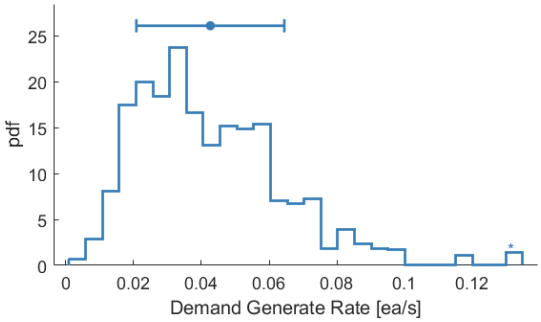


Figure 4.21 Demand Generation Rate Distribution of Partition Pairs

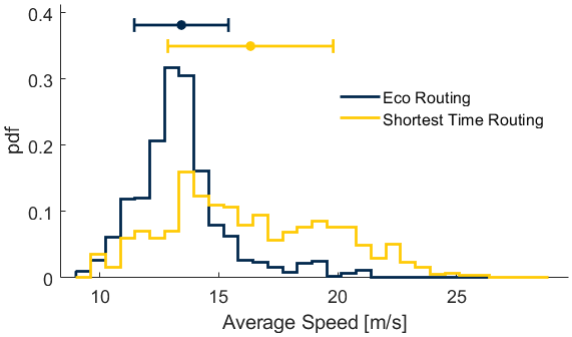


Figure 4.22 Average Speed Distribution of Partition Pairs

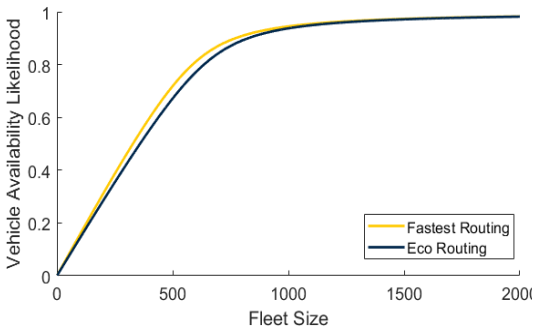


Figure 4.23 Vehicle Availability Estimated Using Queuing Network Model

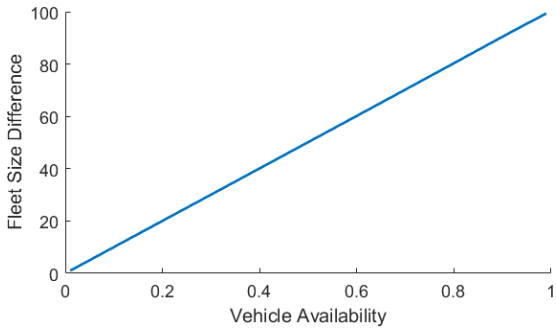


Figure 4.24 Number of Extra Vehicle Required for Eco Routing to Achieve Same Availability as Shortest Time Routing

evening rush hour. The average travel speed and distance are estimated for shortest time routing and eco-routing. The distributions of network statistics required to estimate fleet size using the distance-based approach are shown from Figure 4.19 to Figure 4.22. The estimated minimum fleet size for eco-routing is 1,176, and when shortest time routing is used, the minimum fleet size is 1,039 to serve 4% total travel demand generated from 17:00 to 19:00. Since the approach only addresses the minimum fleet size problem using travel distance and average speed, the wait time of customers can be too long [137]. Therefore, the distance-based approach can be used as a lower bound estimation if no shared ride is allowed.

Availability as a function of fleet size using both shortest time routing and eco-routing are shown in Figure 4.23 and the number of extra vehicles needed when eco-routing is used to achieve same availability as shortest time routing is shown in Figure 4.24. Due to the lower average speed results from eco-routing strategy, more vehicles are required to achieve the same availability compared with the shortest time routing. Under the assumptions of the queuing network based formulation, to achieve more than 95%

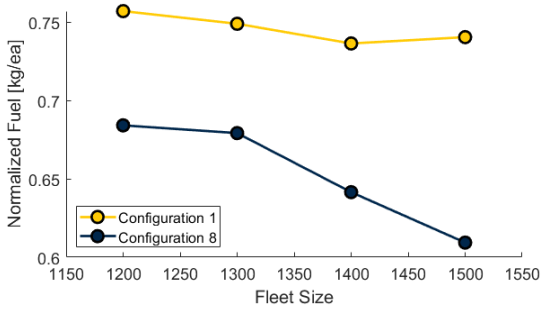


Figure 4.25 Fuel consumption normalized with served customer amount

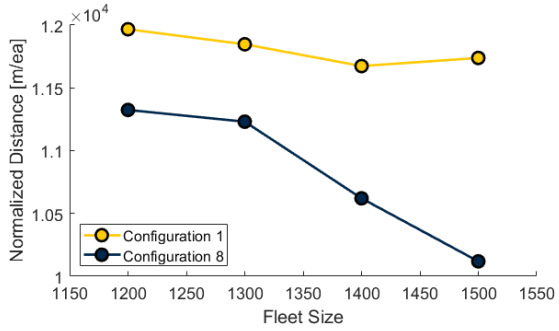


Figure 4.26 Travel distance normalized with served customer amount

availability for all partitions, 1,321 vehicles are required when using the eco-routing strategy, and 1,134 vehicles are required when using the shortest time routing strategy.

Numerical simulations are used for performance evaluation using different fleet sizes given the max wait time of 5 minutes and the max delay time of 5 minutes for configuration 1 and configuration 8. The fleet performance is summarized from Figure 4.25 to Figure 4.35. The fuel oriented configuration (configuration 8) consumes less fuel compared with the travel time oriented configuration (configuration 1) as shown in Figure 4.25, and so is total travel distance. Due to the fleet cost-oriented objective function in the assignment step, the average number of customers per vehicle is higher for configuration

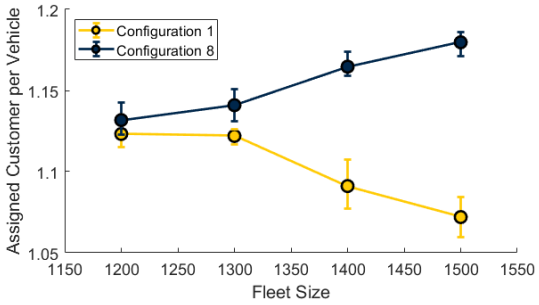


Figure 4.27 Average number of customers assigned per running vehicle

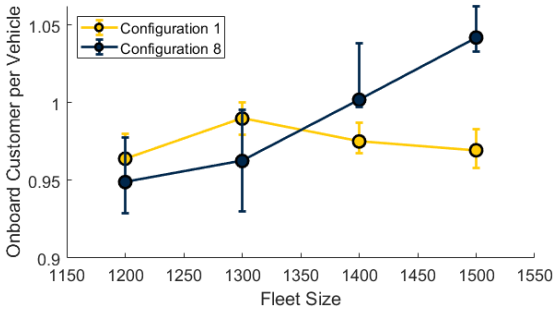


Figure 4.28 Average number of customers on-board per running vehicle

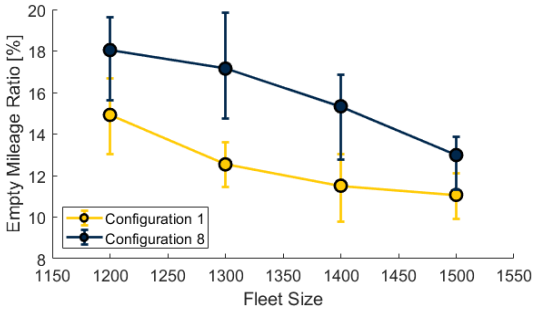


Figure 4.29 Empty vehicle travel mileage ratio

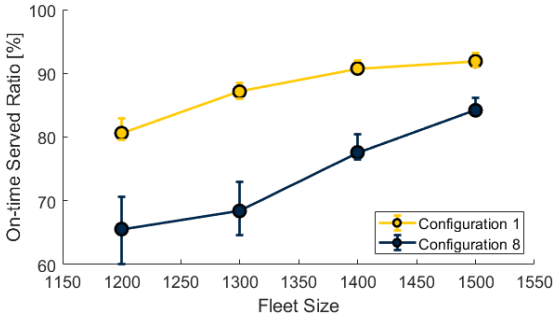


Figure 4.30 Customer served within travel time constraints

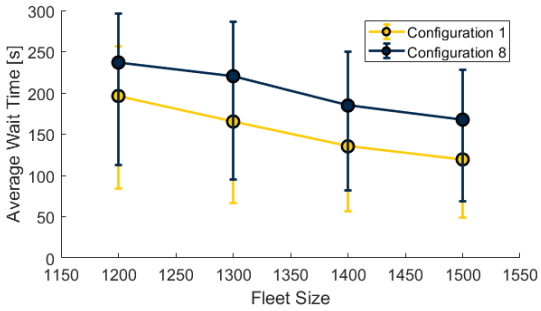


Figure 4.31 Average wait time

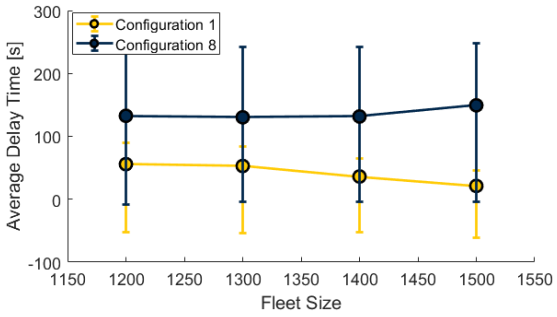


Figure 4.32 Average delay time

8, indicating more trips are shared when the fleet size is larger than 1,300. However, the average wait time and delay time of configuration 8 are longer than configuration 1 for all fleet sizes, and fewer customers are served within their travel time constraints. For all configurations, the ratio of customers served within time constraints increases with the increasing of fleet size. For configuration 1, 1,200 vehicles can serve more than 80% of the customers within time constraints, while 1,500 vehicles are required for configuration 8 to serve 80% customers. In the following sections, fleet size 1,500 is used.

4.6.3 MOD and Routing Strategy's Influence on Energy.

The main goal of the simulations is to assess the impact of different routing strategies on fuel consumption. In this Section, we fix the demand ratio served by the MOD fleet at 4% total demand during the weekday from 17:00 to 19:00. The simulated data from 18:00 ~ 19:00 is used for evaluation after the system reaches steady state. The fleet size is fixed at 1,500, which is the fleet size necessary to serve 80% of the customers within their travel time constraints for all configurations from the previous section. Of course, this means the fleet size is perhaps larger than truly necessary for some configurations. However, when the vehicles are not dispatched, they incur neither time nor fuel cost, and thus will not affect the final performance measures. The performance of shareability is shown in Figure 4.33. It can be seen that when the fleet cost is minimized,

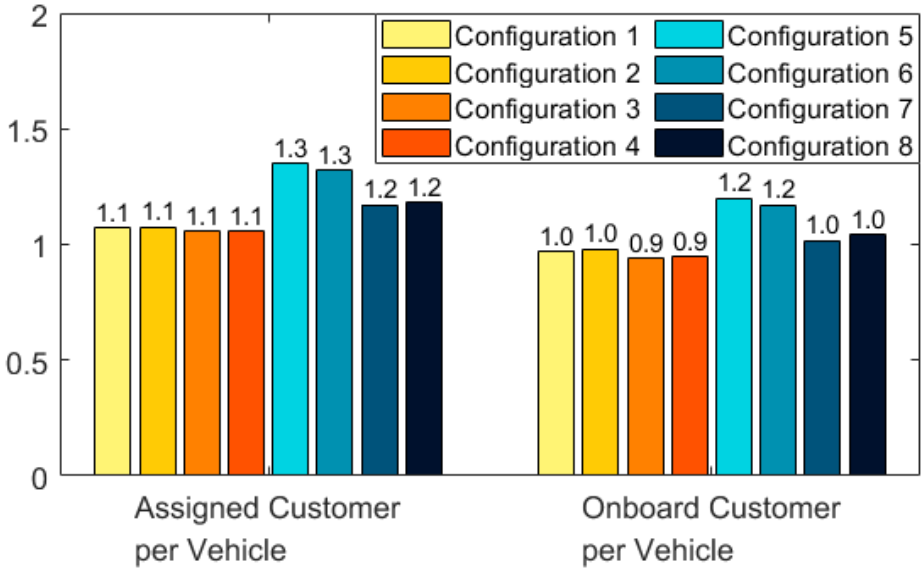


Figure 4.33 MOD algorithm performance comparison — average customer assigned and onboard of each vehicle

more shared trips are selected, and the average number of assigned customer per vehicle increases from 1.1 to 1.3, and the average number of onboard customers per vehicle increases from 1 to 1.2, indicating that more trips are shared and empty vehicle miles is reduced. However, due to the difference in the origin and destination distributions as well as the lower trip average speed, more rebalance trips are assigned for which no shared trips are allowed when eco-routing is applied. The increased rebalance trips lead to a reduced average number of customers assigned per vehicle from 1.3 to 1.2 and the number of onboard customers from 1.2 to 1 when the assignment objective is fleet fuel consumption.

The performance in fuel consumption and vehicle mileage are summarized in Figure 4.34 to Figure 4.35. When the objective function of the trip assignment is travel time and the shortest time routing strategy steady-state, the fuel consumption per customer is increased by 6.2% compared with the baseline that every customer uses personal vehicle. Use eco-routing for unoccupied vehicles can reduce fuel usage per customer served by 1.3%, but still consumes 4.8% more fuel compared with the baseline. However, if the objective function is to minimize the fleet fuel consumption, the fuel consumption per customer can be reduced by 9% to 12% compared with the baseline.

The results indicate that shared-rides have the potential to reduce trip fuel consumption by up to 12%, but if the fleet is not properly operated, the total fuel consumption can increase. The results also indicate that with the same objective function,

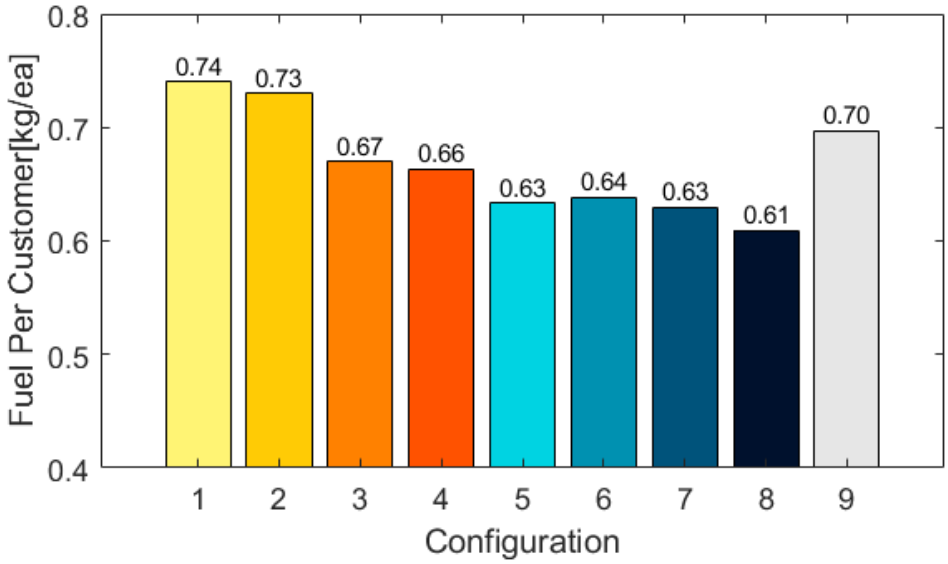


Figure 4.34 Fuel Consumption per Customer

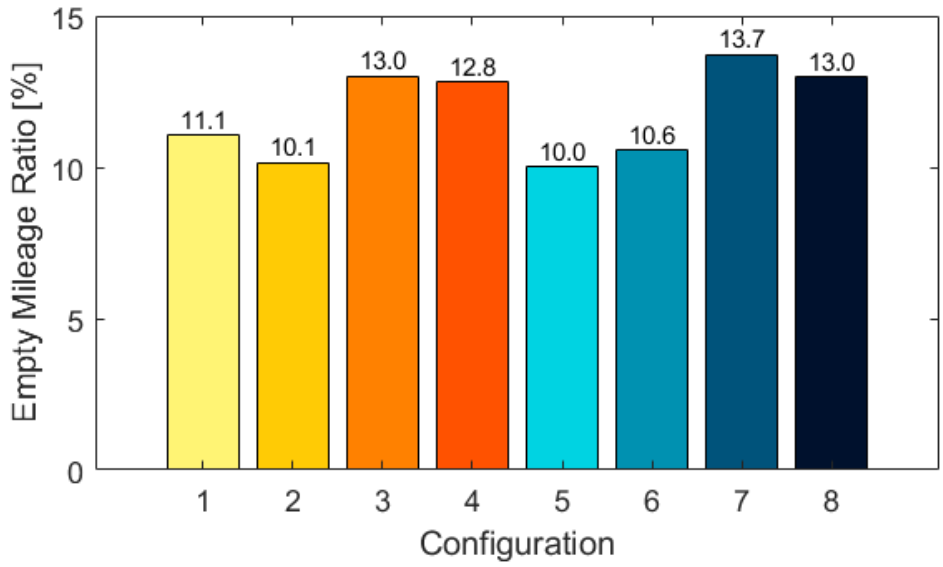


Figure 4.35 Empty travel distance ratio

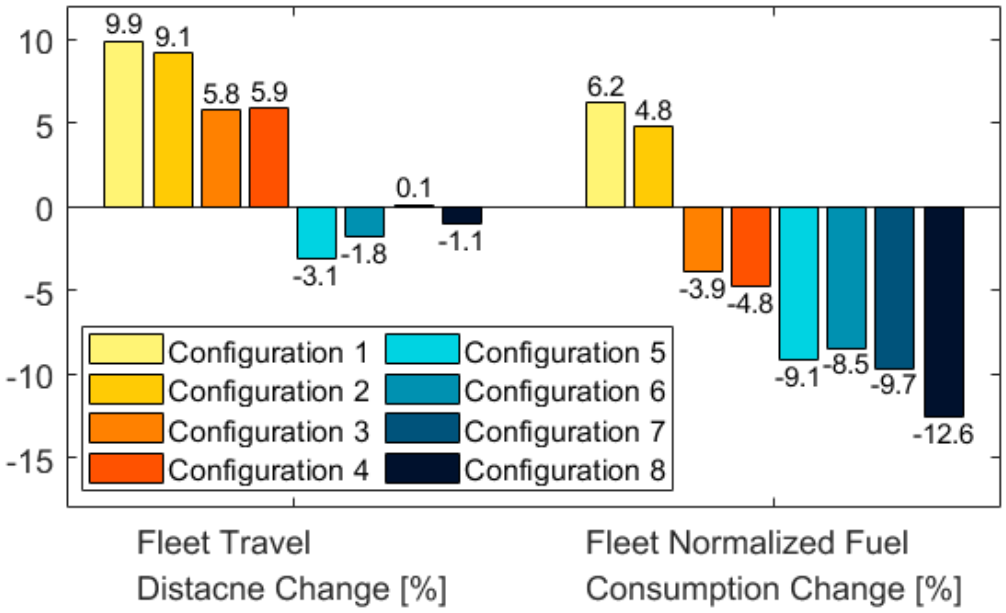


Figure 4.36 Change in fleet total travel distance and fuel consumption per customer

using eco-routing for trips can further reduce fuel consumption by 10% if the trip assignment objective is travel time, and 4% if the trip assignment objective function is fleet fuel consumption compared with the configurations that use the fastest route for trips. When fleet fuel is optimized, using eco-routing can reduce the average travel speed, and thus increase the ratio of rebalance trips and reduce shareability, making the additional benefit to fuel consumption limited.

The fleet mileage is reduced with the system optimal objective function due to the increase in shareability and usage of eco-routing. Compared with the baseline, the fleet mileage can be reduced by more than 3%. However, if the objective function is travel time, the mileage can be increased by 5.8% to 9.9% compared with the baseline. With the application of the eco-routing, the fleet travel mileage can be reduced by 3.7% for configurations 1-4 where routing strategy has little impact on shareability. However, for configurations 4-8, where the application of the eco-routing strategy reduces shareability of the fleet, the fleet travel mileage is increased by 3.3% due to the increase in rebalancing trips. With the increased rebalance trips, the empty vehicle traveling mileage is increased, as shown in Figure 4.35.

The travel time performance is summarized in Figure 4.37 and Figure 4.38. Since the wait time and delay time of customers are modeled as constraints for trip assignment, all configurations can serve more than 80% of the customers within the travel time constraints. The mean wait time when customers‘ travel time is minimized in the assignment step is up to 167 seconds when the fleet fuel is optimized and eco-routing is applied. Compared with traveling using the shortest time route individually, the average delay time can be up to 147 seconds due to the shared trips and lower average travel speed of eco-routing.

In summary, shared mobility has the potential to reduce total fuel consumption but may incur travel time increase due to shared trips. One potential solution is to use eco-

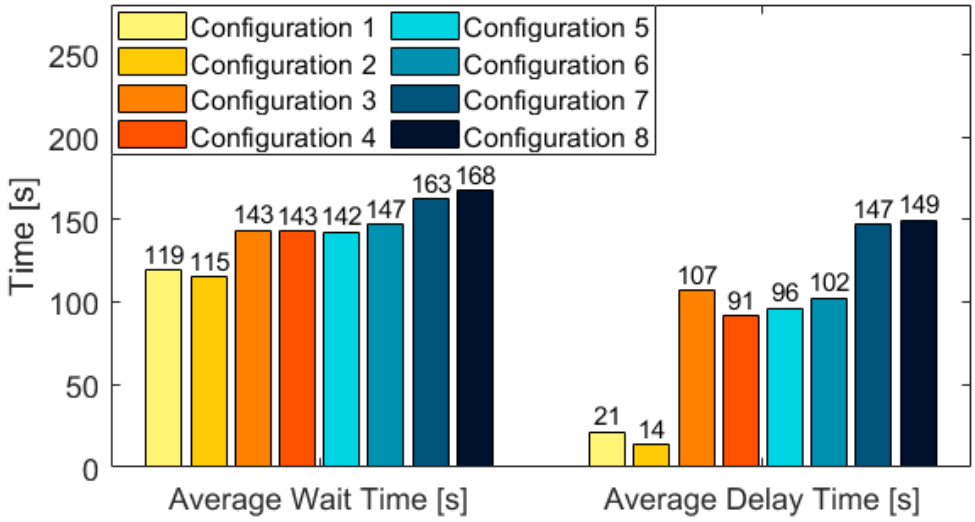


Figure 4.37 Time Performance Comparison of Configurations

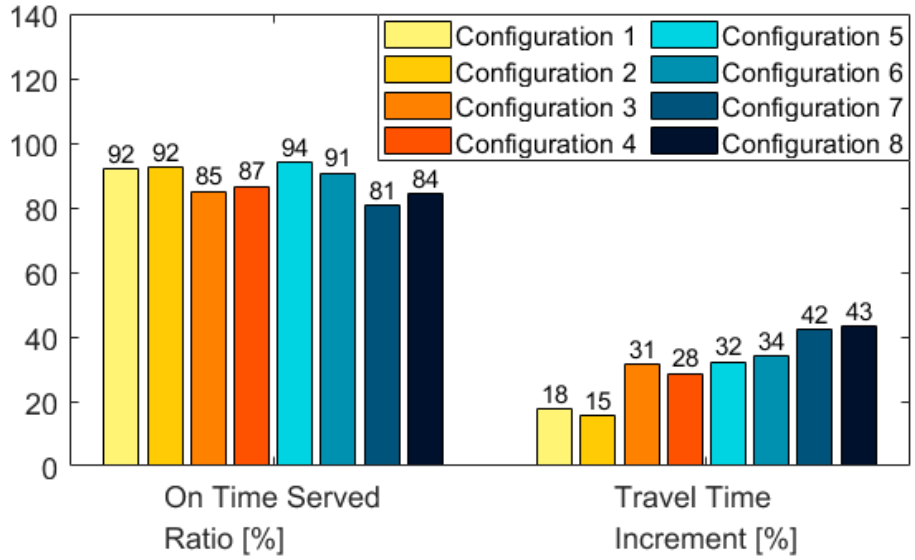


Figure 4.38 Ratio of customers served within travel time constraints

routing for trip execution, which would result in reduced fuel consumption with a small increase in travel time. The objective function can also be defined as a weighted sum of individual benefit and system benefit and parametric study can be used to find the Pareto optimal points.

4.6.4 MOD and Routing Strategy's Influence on Traffic Speed

The influence on traffic flow is evaluated with SUMO. The average speed of different configurations normalized using the posted speed limit are shown in Figure 4.39. Due to the increase in empty vehicle trips, the average normalized speed decreased 1% for

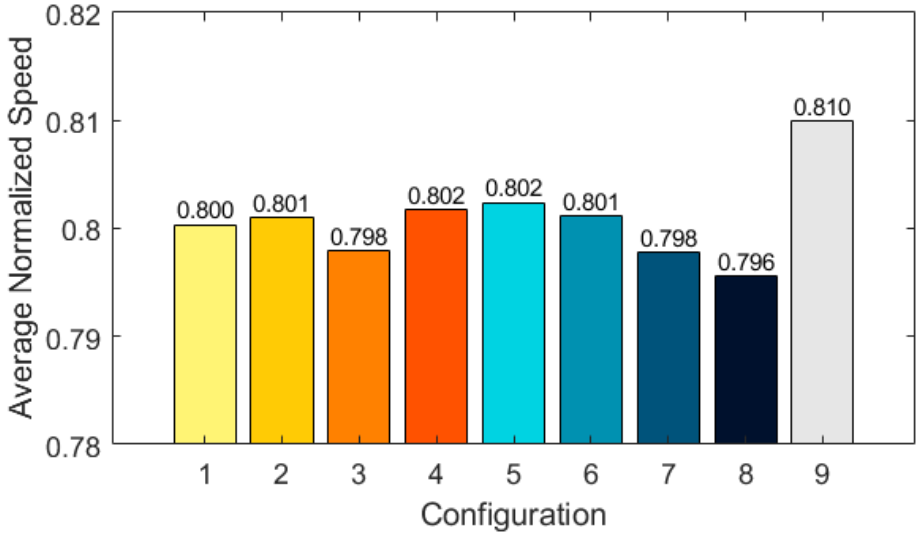


Figure 4.39 Average Link Relative Speed

Table 4.2 K-S test of road network normalized link average speed for all configurations against personal vehicle driving baseline

Configuration	1	2	3	4	5	6	7	8
K-S Test p value	8E-15	9E-11	1E-14	2E-11	1E-10	8E-11	7E-15	1E-13

the road network. Two-sample Kolmogorov-Smirnov test [141] is used to test against the null hypothesis that the normalized link speed for all configurations are from the same distribution as the baseline, and the results are summarized in Table 4.2. As shown in the table, the hypotheses are rejected with p-values less than 0.05 for all configurations, indicating the change in network average speed is statistically different from the baseline for all configurations. However, in the baseline, more than 5,000 vehicles are generated from the travel demand while the MOD fleet only consists of 1,500 vehicles, thus the MOD still outperforms personal driving in terms of traffic flow due to the reduced number of vehicles. Besides the influence on traffic flow, the shared fleet can also reduce parking spaces, which can further reduce travel time due to the reduction in travels searching for parking. However, the effect of parking space is not included in our simulations and is considered to be part of the future works.

4.7 Conclusion and Future works

An Eco-MOD fleet assignment framework is developed to minimize fleet fuel consumption while satisfying customer travel time constraints. The algorithm shows the potential to reduce fleet fuel consumption while serving more than 80% of the customers within their travel time constraints. Numerical simulations show that the benefits of EcoMOD increase with the percentage of customers using the service. However, in the current framework, the assignment only considers realization of travel demand at current step without considering expected future assignment cost and routing strategy is selected heuristically, thus developing an optimal routing strategy selection algorithm based on knowledge of fleet and demand distribution could be fruitful.

CHAPTER 5
Traffic Network Partition and Idling Vehicle Rebalancing

5.1 Introduction

Mobility-on-demand (MOD) services such as Uber and Lyft have changed the landscape of ground mobility significantly, especially in urban areas with dense population. When multiple passengers share the same vehicle (e.g., Lyft Line and UberPOOL), the service can reduce the number of vehicles on the road and parked on streets, and reduce congestion and energy consumption.

Today's MOD fleet management largely reacts to trip requests without utilizing predicted future supply and travel demand distribution. Continuous approximation [110] is used to study the dynamics of fleets and the influence of large fleets to congestion. Algorithms such as mixed integer programming [98], heuristic [103], and graph-based decomposition [107] were developed to assign the fleet to customers. A privacy-preserving algorithm was developed [109] to protect the location information of passengers without incurring significant performance degradation. However, the potential of the fleet is not fully utilized due to the nature of reactive control policies.

Knowledge of travel demand distribution plays a vital role in the control of MOD fleet. For carpool service with private cars, travel data can be used to identify the optimally combined trips for carpooling and can reduce daily car mileage by 44% [112]. Intelligent transportation technologies such as connected automated vehicles provide richer information about travel demand and enable better centralized MOD fleet control. Han et al. [113] showed that with driverless MOD fleet, the direct control approach is 29% more efficient compared with the current price-based indirect control. For service provided by commercial fleets, travel demand distribution can be used to control the idling vehicles for rebalancing [113–115] to better meet future trip requests. A sampling-based algorithm is also proposed to control ride-sharing fleets using predicted future trip requests [64].

However, the travel location distribution is either characterized by clusters from the geometric coordinate of locations [112, 142] or grid-based discretization [114], neither of which takes the structure of the road network into consideration. Recently, an integer programming based approach [117] is proposed for road network partition and rebalancing location selection with the maximum wait time being bounded for the Manhattan island.

Since travel demand can be characterized as a random variable on the road network, ignoring the underlying network structure can be problematic. To better describe the travel location distribution considering the structure of the road network, we propose an algorithm using multi-dimensional scaling (MDS) [143] to project the locations in the road network onto a Euclidean space. The travel locations can then be characterized by a Dirichlet Process Gaussian Mixture Model (DPGMM) [144]. The projection allows us to obtain better clustering results compared with the geometric coordinate-based methods. To utilize the demand distribution information for fleet management, we developed an idling fleet rebalancing control algorithm based on the work in [145]. We assume that the demand distribution is known, and the fleet can be controlled directly to take assigned trips and to rebalance, considering future demands. It should be noted that we do not assume the future trips are known exactly, but their probability distribution is known. In our simulations, travel demands generated by POLARIS [75], a mesoscopic agent-based transportation model, are used as prior and calibrated using data from the Safety Pilot Model Deployment (SPMD) project [76] as presented in Chapter 4. The calibration dataset consists of trip information from up to 2,800 vehicles from the city of Ann Arbor running continuously since 2012.

The main contributions of this work are: 1) a travel location clustering algorithm considering the road network structure; 2) a ride-sharing idle fleet rebalancing control policy considering future travel demand distribution is developed to reduce expected wait time of future trips.

The rest of this chapter is organized as follows: Section 5.2 presents the proposed travel location clustering algorithm. Section 5.3 presents the formulation of idle fleet rebalancing optimization. Section 5.4 presents the simulation results. Conclusions and future work are given in Section 5.5.

5.2 Road Network Partition

To characterize travel location distribution considering the road network structure, we model the road network with a Euclidean space approximation. With this approximation, we characterize the distribution of travel locations with the Lebesgue measure.

5.2.1 Multidimensional Scaling (MDS)

The MDS method is used to find the optimal Euclidian space that preserves pairwise distance in the network space. MDS can be formulated as an optimization problem defined as

$$\min_{x_1,\dots,x_N}\left(\frac{\sum_{i,j}\left(d_{ij}^{p}-\left\|x_i-x_j\right\|^{p}\right)^2}{\sum_{i,j} d_{ij}^{2p}}\right) \tag{5.1}$$

where d_{ij} is the pairwise distance between points i and j, $x_i, x_j \in \mathbb{R}^m$ are the vectors corresponding to point i and j in the projection space, and m is the dimension of the projection space, which is determined later in this chapter. p is the power transformation used by metric scaling, N is the total number of projected points. Since the projection space is Euclidian, the approximated distance is

$$\left\|x_i-x_j\right\|^2=\sum_{k=1}^{m}\left(x_{ik}-x_{jk}\right)^2 \tag{5.2}$$

When p is 1, the MDS is known as the classical MDS and can be solved with eigendecomposition by transferring the distance matrix to an inner product through double re-centering

$$G=-\frac{1}{2}\left(I_n-\frac{1}{n}\mathbf{1}\mathbf{1}^T\right)D\left(I_n-\frac{1}{n}\mathbf{1}\mathbf{1}^T\right) \tag{5.3}$$

Where $D=\{d_{ij}\}$ is the distance matrix, I_n is the identity matrix and **1** is the column vector with 1 as all its entries. With this transformation, vectors in the projection space can be obtained by eigendecomposition of G, which gives

$$x_i^*=\sqrt{\lambda_i}u_i, i=1\dots m \tag{5.4}$$

where λ_i is the i-th largest eigenvalue of G, and u_i is the corresponding eigenvector. When $p \geq 2$, the optimization problem can be solved using the steepest gradient method [143] where the solution of the classical MDS is used as the initial point for the numerical algorithm. In the following analysis, we use non-classical MDS with $p = 2$ to approximate the pairwise distance in the non-Euclidean road network space.

The distance matrix is obtained by calculating the pairwise lowest cost path distance between every pair of links in the traffic network. The traffic network is defined as a weighted directed graph with nodes associated with links of the original road network. An edge from node i to node j exists if link j is adjacent to link i and if a vehicle can travel from link i to link j (one-way roads is an example when this is not the case). The weight of the edges is defined as

$$w_{ij} = \frac{1}{2}\left(\frac{l_i}{\overline{v_i}} + \frac{l_j}{\overline{v_j}}\right) \tag{5.5}$$

where l_i, l_j are lengths of links i and j, $\overline{v_i}, \overline{v_j}$ are the travel speeds, which can be the posted speed limits of the road links if no real-time traffic information is available, or measured vehicle speed if the information is available. The graph is connected since there are no isolated links in the traffic network. The pairwise distance is solved using linear programming based on the Bellman inequality, which is the dual of Bellman-Ford algorithm [146] and can be solved efficiently with optimization solver such as Gurobi [50]

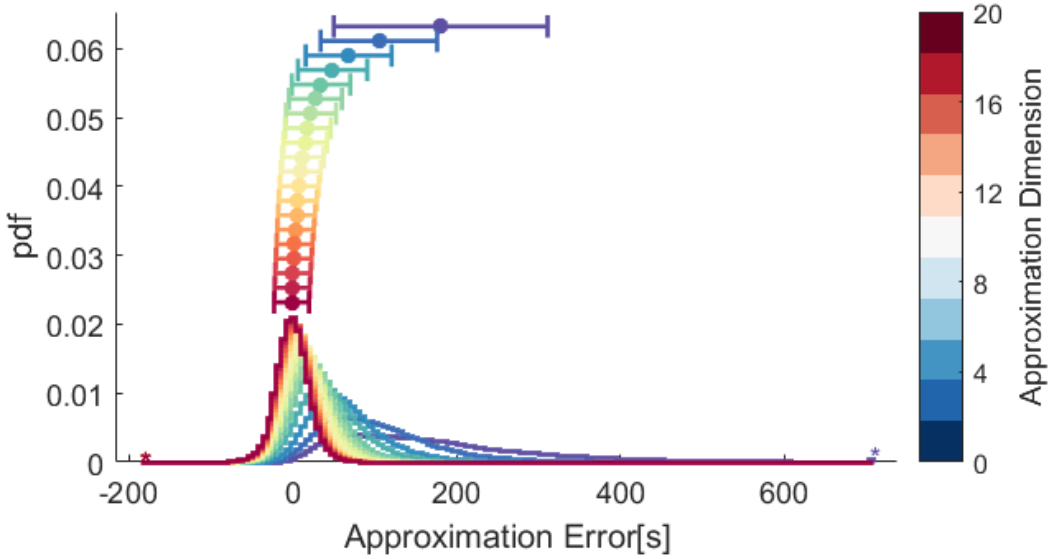


Figure 5.1 Approximation Error with Different Approximation Dimension

which we use. The approximation performance with different projection space dimension is shown in Figure 5.1, with mean and standard deviation marked using error bars.

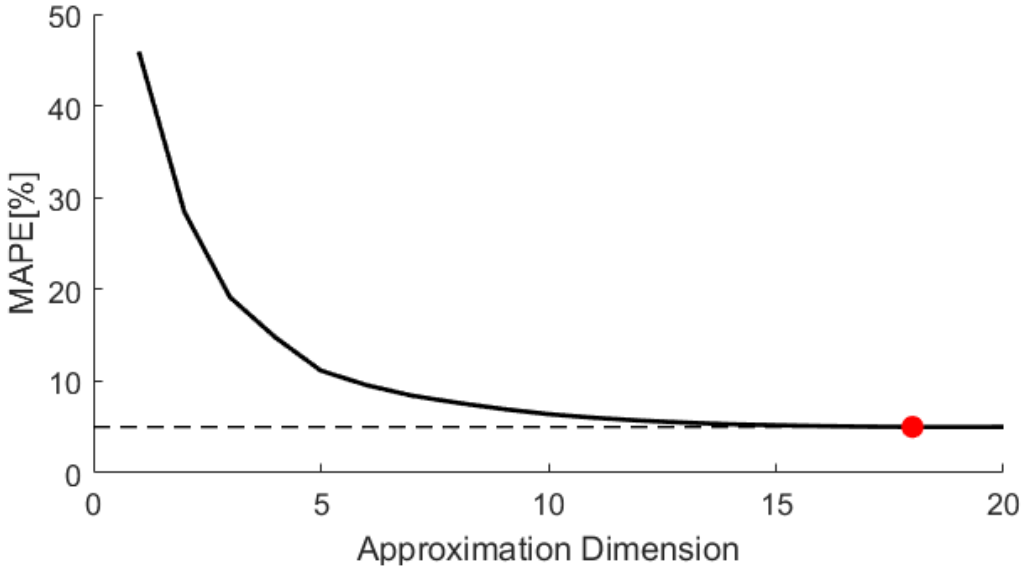


Figure 5.2 MAPE for different approximation dimension

In the following analysis, we choose 18 dimensions for the Euclidean space approximation (i.e., m=18), and the mean absolute percentage error (MAPE) is 5%, which is enough to preserve the pairwise distances of the original road network.

5.2.2 Location Distribution Characterization

In the literature, travel location distribution is frequently characterized using the Cartesian coordinate in the geometric space [112, 145, 147], which ignores road network structure information. With the approximation of link locations in a Euclidean space, we can analyze the location distribution in the approximation space using the Lebesgue measure, which preserves the original travel time in the network. In the following analysis, we assume that the origin and destination of each trip are sampled from the location distribution, and the union of origins and destinations is defined as locations of interests. Since we do not assume we know the number of clusters, we use the DPGMM to model the random variable. DPGMM is a Bayesian nonparametric extension of the Finite Gaussian Mixture Model whose probability density function can be expressed by:

$$f_X(x) = \sum_{k=1}^{K} \pi_k f_{X,k}(x) \tag{5.6}$$

where x is the random variable for travel locations, $f_X(x)$ is the overall density function, π_k is the mixing coefficient for each component, $f_{X,k}(x)$ is the density for each component, which follows a multivariate Gaussian distribution. Instead of a fixed component number K, DPGMM assumes the model consists of infinite components, i.e., $K \to \infty$ in (5.6). With this method, not only the parameters for each mixture component but also the number of mixture components can be inferred from the data. In this way, the locations of interests are modeled as the mean of each mixture component, and travel demand can be modeled as a multinomial distribution with the discretization achieved using the mixture model. The posterior of parameters of the DPGMM is inferred through collapsed Gibbs sampling, which is an approximate inference algorithm based on Markov Chain Monte Carlo (MCMC) sampling and known to be unbiased asymptotically compared with other approximate inference methods such as variational inference. The process is summarized as follows. Denote $c_i \in \{1, \dots, K\}$ as the indicator variable of the component for each data point, which is a discrete random variable parameterized by π

$$P(c_i|\pi) \sim Discrete(\pi_1, \dots, \pi_K) \tag{5.7}$$

For Dirichlet process model, $K \to \infty$. The parameters are modeled with their corresponding conjugate priors, i.e., Dirichlet distribution for π and Gaussian-Wishart distribution for mean μ and covariance Σ of each component.

$$P(\pi) = Dir(\pi|\alpha_0) = C(\alpha_0) \prod_{k=1}^{K} \pi_k^{\alpha_0 - 1} \tag{5.8}$$

$$P(\mu, \Sigma) = P(\mu|\Sigma)P(\Sigma) = \prod_{k=1}^{K} N(\mu_k|m_0, \beta_0\Sigma_k)W(\Sigma_k^{-1}|W_0, v_0) \tag{5.9}$$

where α_0 is the hyper-parameter for Dirichlet distribution, m_0, β_0, W_0, v_0 are the hyper-parameters for the Gaussian-Wishart distribution. For simplicity, we denote $\{m_0, \beta_0, W_0, v_0\}$ the set of hyperparameters for the Gaussian-Wishart distribution as γ. The hidden variables include the indicator variable c_i and the model parameters π, μ, Σ. At each step of collapsed Gibbs sampling, we sample c_i conditional on the rest of the data points and random variables from the posterior

$$p(c_i = k|c_{-i}, x, \alpha_0, \gamma) \propto p(c_i = k|c_{-i}, \alpha_0)p(x|c_{-i}, c_i = k, \gamma) \tag{5.10}$$

where $c_{-i} = \{c_j, j \neq i, j \in \mathbb{N}, 1 \leq j \leq N\}$ is the set of indicator variables for other samples except i, N is the sample size of the entire dataset. Since the prior of other parameters are

well-defined, the inference can be carried out in a closed form. Thus, no sampling is required to obtain the posterior of μ and Σ once c_i's are sampled for all data points.

The likelihood term can be obtained in a closed form from the Gaussian-Wishart distribution, and the prior term can be defined by the Chinese Restaurant Process (CRP). The resultant cluster assignment follows the pattern that the probability of a new sample belonging to a cluster is proportional to the number of samples already in the cluster.

$$p(c_i = k | c_{-i}, \alpha_0) = \begin{cases} \dfrac{N_{-i,k}}{N + \alpha_0 - 1} & If\ k \leq K \\ \dfrac{\alpha_0}{N + \alpha_0 - 1} & If\ k = K + 1 \end{cases} \tag{5.11}$$

where $N_{-i,k}$ is the sample size of data belong to cluster k for other samples except for i, K is the current number of clusters already realized. In this way, as the sample size N goes to infinity, the number of clusters can go to infinity, indicating that the model is more complex with more samples acquired. DPGMM is used to identify the number of clusters for travel locations in the approximated Euclidean space, which is used for rebalancing destination selection if no information about the cluster number is given.

The objective of partitioning the road network and rebalancing destination selection is to minimize the expected travel time for customers to be reached from the closest rebalancing station, which is the wait time for them to be picked up if the vehicles are located at the partition centers. Thus, the objective function of expected wait time minimization is defined as

$$\min_{\mu} \sum_{k=1}^{K} \sum_{i=1}^{C_k} \|x_i - \mu_k\|_2 \tag{5.12}$$

where x_i is the coordinate of sampled demand i in the MDS approximation space, C_k is the total number of demands assigned to station k, μ_k is the location of station k, K is the number of stations, which can be identified from the DPGMM model. The problem is solved using the k–medoids algorithm initialized with results from DPGMM. k–medoids algorithm has the same objective function as (5.12), and μ_k is restricted to existing data points, while in DPGMM μ_k is the expectation of each component, which is not necessary from the samples. The stations are located at the resultant cluster medoids based on the assumption that the station is located close to links.

5.3 Idling Fleet Rebalancing

Since there is a mismatch between trip origin distribution and trip destination distribution, idling vehicles tend to build up with the trip destination distribution, which would increase the expected wait time of new customers. To mitigate this effect, idling vehicles should be relocated according to the trip origin distribution to reduce the expected wait time for future travel demands. After the road network is partitioned, the trip origin distribution and idling fleet distribution can be described using random variables following categorical distribution. The objective of fleet rebalancing is to minimize the difference between these two distributions. Assuming known trip origin distribution and expected customer departure rate, the problem is formulated as an integer programming with quadratic objective function and linear constraints. To reduce the size of integer programming, we only consider the case that idling vehicles being relocated to immediate adjacent partitions.

The decision variables are defined as indicator variables of relocating trips associated with each idling vehicle, $\mathcal{T} = \{t_{ij}, \forall i \in \mathcal{V}_I, \forall j \in adj(v_i)\}$, where $\mathcal{V}_I$ is the set of idling vehicles, $adj(v_i)$ is the set of adjacent partitions of the idling vehicle v_i. $t_{ij} = 1$ if idling vehicle i is assigned to relocate to adjacent partition j and otherwise is 0. For simplicity, we assign all relocating trips to the corresponding partition center. The objective function is estimated using planning horizon T, and we only consider assignment at the current step. The objective function is defined as

$$\min_{t_{ij} \in \mathcal{T}} (1 - w_c) \sum_k \sum_{\tau=1}^{T} \left\| \frac{n_k^\tau}{N_\tau} - o_k \right\|^2 + w_c \sum_{t_{ij} \in \mathcal{T}} C_{ij} t_{ij} \tag{5.13}$$

where N_τ is the normalization constant, o_k is the density function of trip origins associated with partition k, C_{ij} is the traveling cost associated with relocating trip t_{ij} and fuel consumption is used here, w_c is weighting parameter between relocating cost and balancing objective, n_k^τ is the estimated number of available vehicles within partition k at time τ. Assuming that customers depart in each partition follow a Poisson process,

$$n_k^\tau = \sum_{t_{ij} \in \mathcal{T}} t_{ij}^{k,\tau} + d_k^\tau - \tau \gamma \lambda_k \tag{5.14}$$

where $t_{ij}^{k,\tau}$ is trajectory indicator of rebalancing trip t_{ij}, $t_{ij}^{k,\tau} = 1$ if the vehicle is located within partition k at time τ after departure, which can be estimated as a function of t_{ij} using estimated travel time, d_k^τ is the amount of arrival vehicles within partition k up to time τ, γ is a discount factor to account for the ratio of trips being shared, $\gamma \leq 1$, λ_k is the expected customer departure rate at partition k. The normalization constant is given by

$$N_\tau = \sum_k n_k^\tau = N_{idle} + \sum_k d_k^\tau - \tau\gamma \sum_k \lambda_k \tag{5.15}$$

where N_{idle} is the amount of idling vehicles. In addition to the trips to other partitions, virtual trips that the vehicle stays at the same location are also generated with the destination assigned to be the vehicle's current location. Thus the constraint is that each vehicle is assigned to one relocating trip.

$$\sum_j t_{ij} = 1, \forall i \tag{5.16}$$

Due to the large size of the problem, instead of solving the mixed integer programming exactly using the branch and bound algorithm, we solve the continuous relaxation of the original problem by relaxing t_{ij} as a real number between 0 and 1. The integer solution is then obtained by randomized rounding [148]. After solving the relaxed problem, the assigned trips are selected by random sampling using the optimum t_{ij} as the density function. The idling fleet rebalancing step is integrated with the MOD assignment framework by assigning idling fleet relocating after the reactive rebalance step to serve customers whose travel time constraints cannot be satisfied by the regular assignment. The optimization is solved with Gurobi [50]. The optimization is solved repeatedly every assignment interval based on the current status of the fleet. Assuming the fleet size is estimate accurately, i.e. all vehicles need to travel either to serve customer or rebalancing according to travel demand, then the first term in the objective function is transformed to flow conservation constraint due to the requirement of balanced operation. Thus, the formulation becomes the minimum cost flow rebalancing formulation presented in Chapter 4 of this dissertation. However, the precise fleet size is complicated to estimate when shared rides are involved and when the demand estimation is not accurate.

5.4 Results and Discussion

In this section, we present the road network partition results using the proposed algorithm. The traffic demand is generated using the algorithm presented in chapter 4. In this study, we focus on demand generated during the evening rush hour (17:00-19:00) on weekdays. However, the algorithms developed can be extended to deal with time-varying demand distribution, which can be modeled as a piece-wise constant function. We first present the results of the road network partition, then demonstrate the idling fleet rebalancing algorithm using simulations.

5.4.1 Road Network Partition

To evaluate the performance of the clustering algorithm, we used the origins of trips generated from 17:00 to 19:00 on a weekday. The travel origins heatmap generated for the evening rush hour (17:00-19:00) of a weekday is shown in Figure 5.3, and the

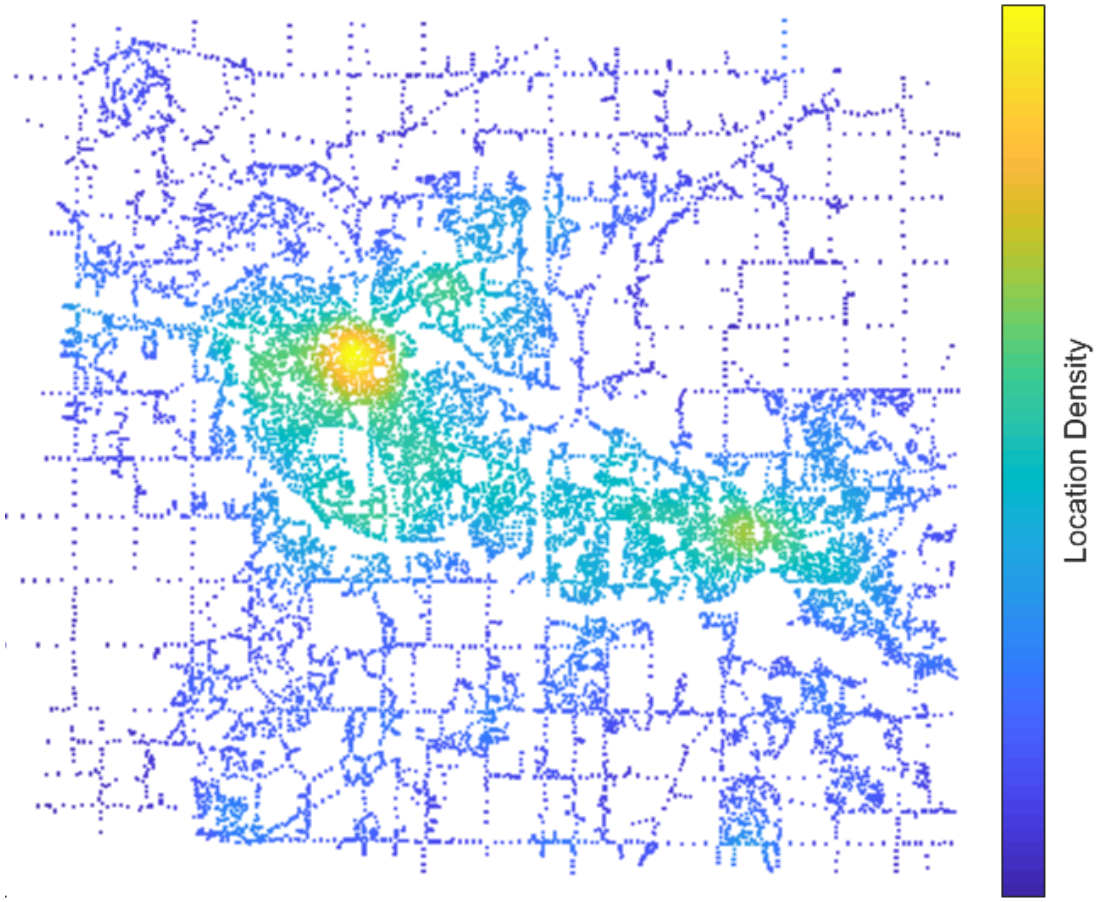


Figure 5.3 Origins heatmap generated from 17:00 to 18:00 on weekdays in Ann Arbor

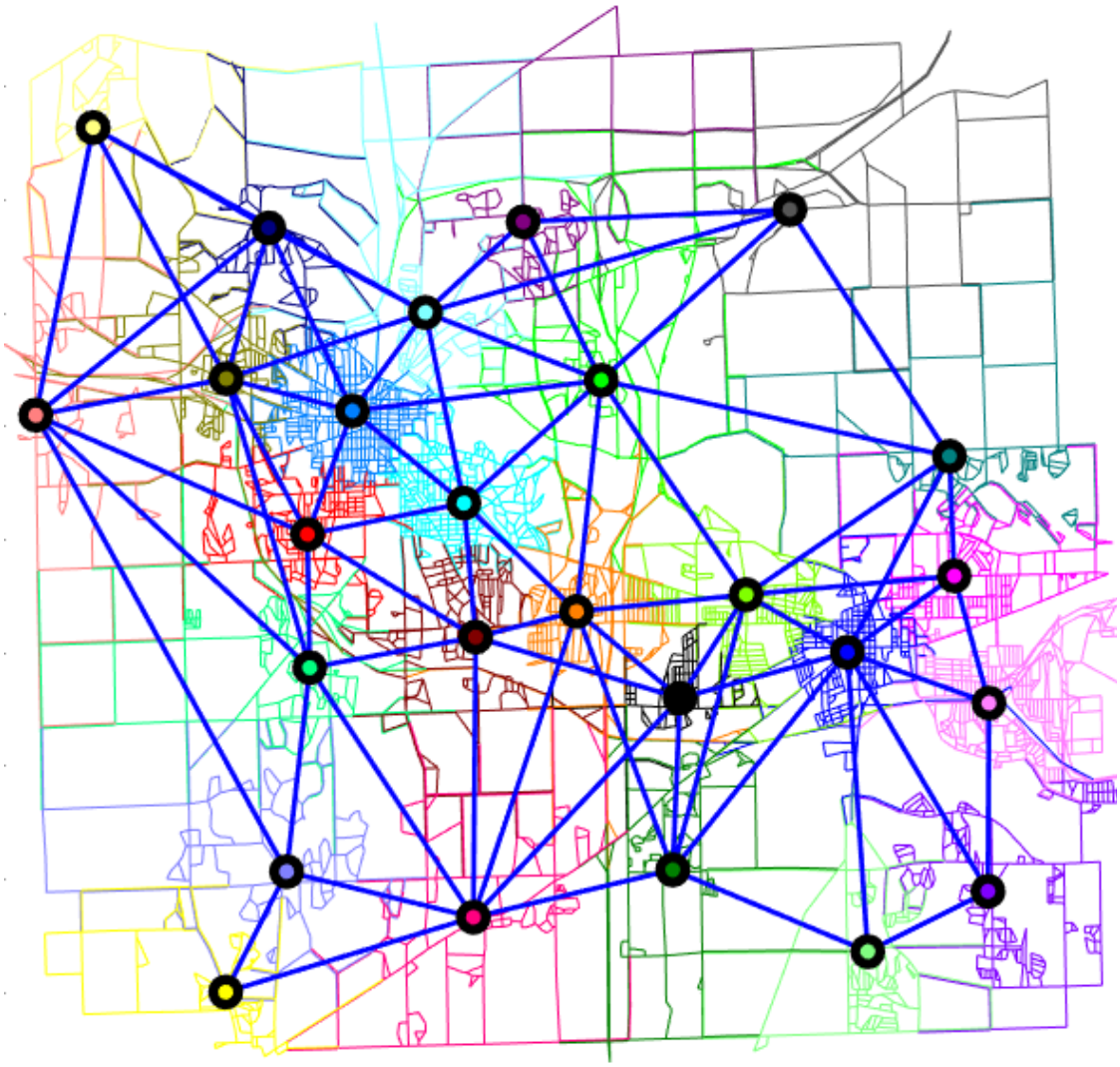


Figure 5.4 Network partition using the proposed algorithm in the approximation space for generated demand, partition centers are denoted as circles. Two partition centers are connected if the two partitions are adjacent, i.e. one can travel from one partition to the other directly without passing through another partitions

corresponding network partition results and the partition centers indicated by circles are shown in Figure 5.4. The algorithm to generate the demands are presented in Section 5. 26 components are identified using the Bayesian nonparametric algorithm, which is used to initialize the k-medoids in the approximation space. Different clusters are shown in different colors in Figure 5.4. The partition centers are represented using circles, and the adjacent relations are represented using edges. Two partition centers are connected if one can travel from one partition to the other without passing through any other partitions.

The benchmarks for traffic network partition are clustering algorithms applied in the geometric Cartesian coordinate space using the k-means algorithm [129, 142] and GMM in the approximation space, without constraining the rebalancing locations to existing links. Since the traffic network partitioning we are interested in is clustering data in the network space instead of studying the connectivity of the network itself, the

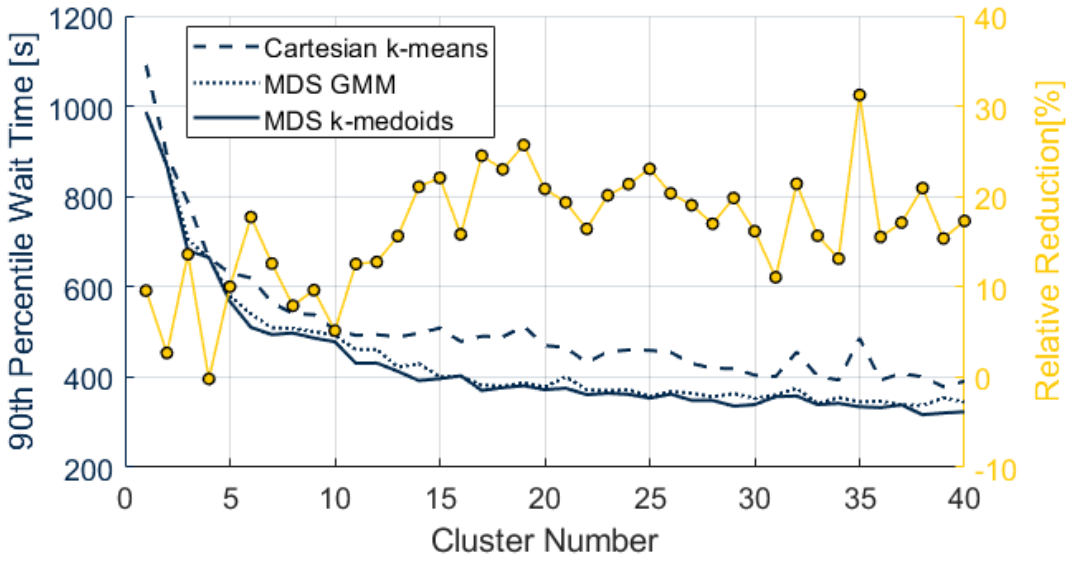


Figure 5.5 90th percentile of travel time to closest partition center for all demands using different clustering algorithm, relative reduction of MDS k-medoids compared with Cartesian k-means is shown in right axis

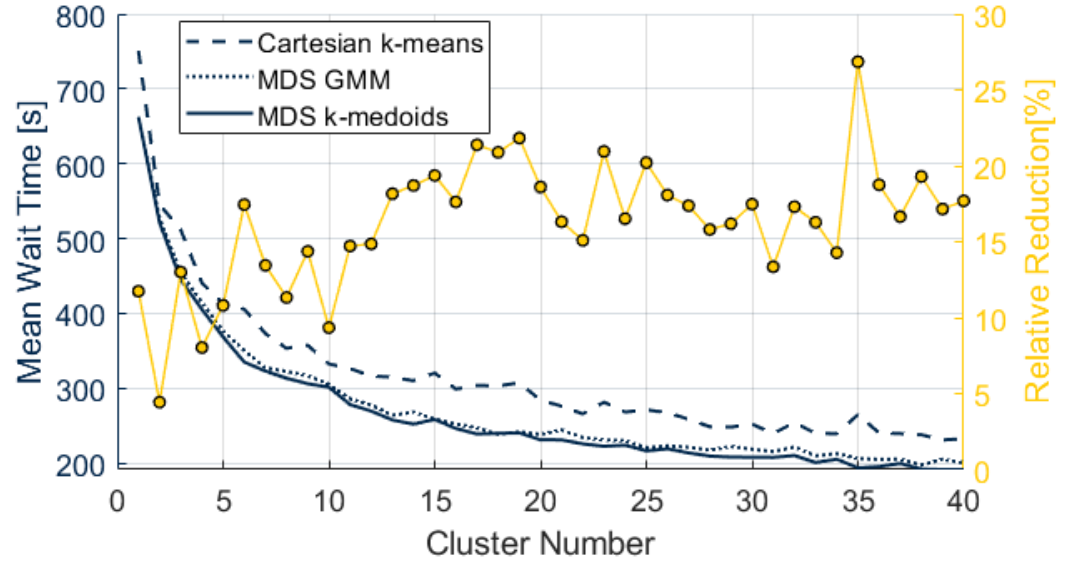


Figure 5.6 Mean travel time to closest partition center for all demands using different clustering algorithm, relative reduction of MDS k-medoids compared with Cartesian k-means is shown in right axis

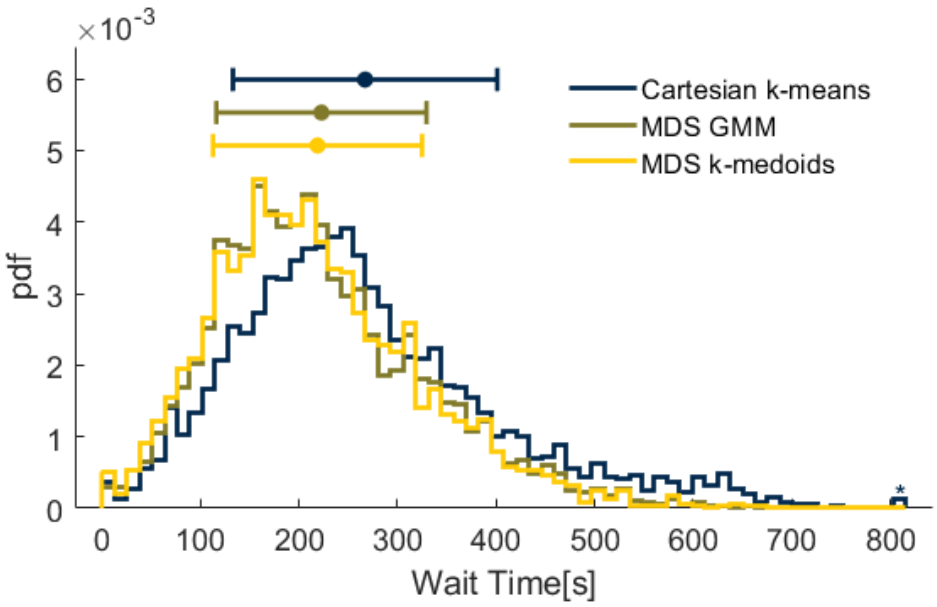


Figure 5.7 Wait time histogram with 26 partitions

Table 5.1 Statistics of wait time with 26 partitions

	Mean [s]	Std.Dev [s]	90th percentile [s]
Cartesian k-means	267	134	455
MDS GMM	223	105	365
MDS k-medoids	218	106	359

community detection algorithms [149] and grid-based discretization [116] are out of scope for our evaluation. Since the component number needs to be specified for the k-means algorithm, to make the evaluation a fair comparison, instead of using Bayesian nonparametric algorithm to identify the component number for GMM and our proposed algorithm, we use the Expectation-Maximization (EM) algorithm to identify parameters for the mixture models. Considering the objective of partition is to select the rebalancing stations, the performance metric we selected is mean value and the 90th percentile of travel time to cluster center, with travel time for each road section generated from SUMO during the studied hour. The 90th percentile and average travel time to cluster center (wait time) for different clustering algorithms are shown in Figure 5.5 and Figure 5.6 respectively, with the right axis showing relative reduction with our algorithm compared with k-means in yellow.

As shown in Figure 5.5 and Figure 5.6, with the right number of mixture component, our algorithm can reduce expected wait time by more than 15% and the 90th percentile of wait time by more than 20% compared with clusters generated using the k-means algorithm in geometric Cartesian coordinate. By considering the constraint that the rebalancing station can only locate close to existing links, the k-medoid algorithm further reduces the expected wait time by 4% and 90th percentile of wait time by 2% compared with GMM in the MDS approximation space directly. The wait time histogram of all travel demands in weekday evening rush hour (17:00~18:00) from different clustering algorithms using 26 partitions are shown in Figure 5.7, and the statistics are summarized in Table 5.1. The number of partition, 26, is identified using DPGMM in the MDS approximation space. As shown in the histogram and table, the proposed algorithm can reduce the mean wait time as well as the standard deviation, indicating that the partitions are more uniform in terms of in-cluster wait time and the benefit is road network-wide applicable.

5.4.2 Parametric Study for Fleet Size

Numerical simulations using the traffic simulator presented in Chapter 4 are used to demonstrate the effect of idling fleet rebalancing in this section. We fix the demand ratio served by the MOD fleet at 4% of the total demand during the weekday evening rush hour (17:00 ~ 19:00). The simulated data from 18:00 ~ 19:00 is used for evaluation after the system reaches steady state. In the trip assignment framework, fuel oriented assignment

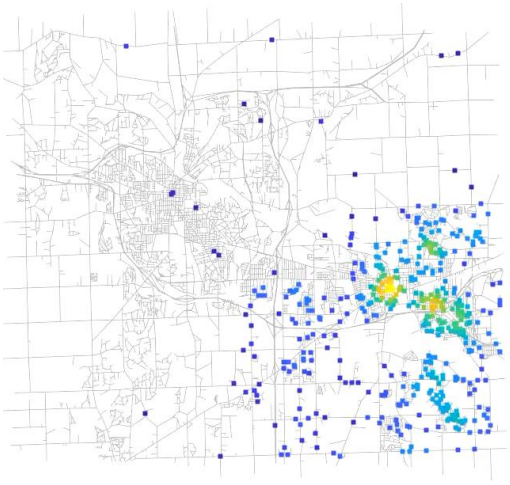


Figure 5.8 Idling fleet with 1,500 vehicles location at 19:00, without rebalancing

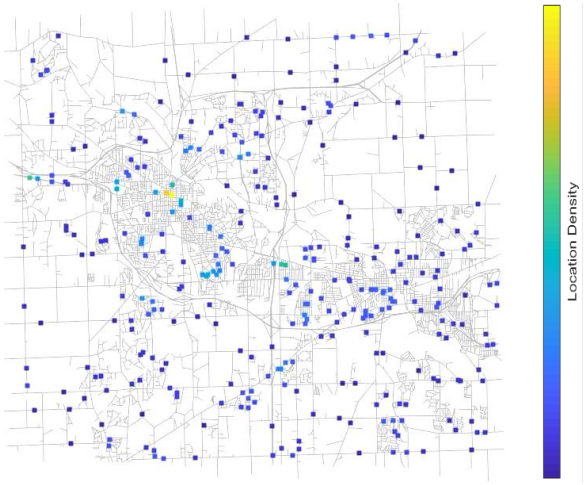


Figure 5.9 Idling fleet with 1,500 vehicles location at 19:00, with rebalancing

(configuration 8) is applied. For the idling rebalancing trips, the eco-routing strategy is applied. The simulation is performed for fleet size 1,200 to 1,500. Customers' maximum wait time and delay time are 5 minutes. In this section, a small weight is chosen in (5.13) for fuel consumption (0.0005). The locations of the idling fleet with and without rebalancing trips using 1,500 vehicles are shown in Figure 4.14 and Figure 4.15. If the fleet is not rebalanced, the idling vehicles will follow the trip destination distribution, and thus the expected wait time of the future trips is increased. With the rebalancing algorithm, we can relocate the idling vehicles according to the origin distribution as shown in Figure 4.15, thus reduce the expected wait time for all trips.

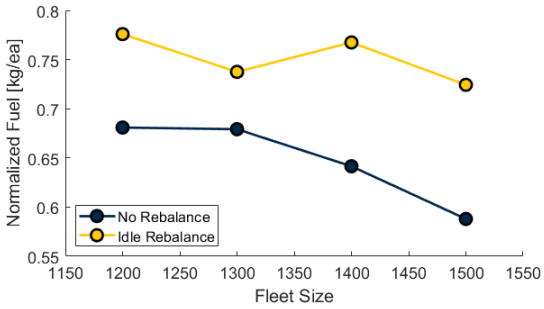


Figure 5.10 Fuel consumption per served customer

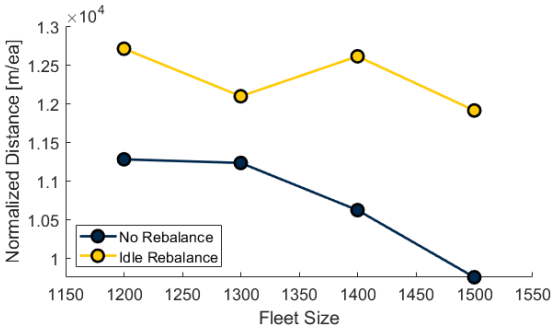


Figure 5.11 Travel distance per served customer

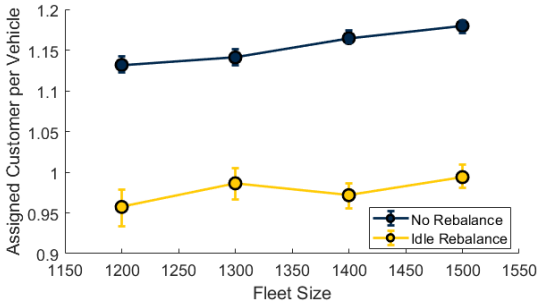


Figure 5.12 Average assigned customer per traveling vehicle

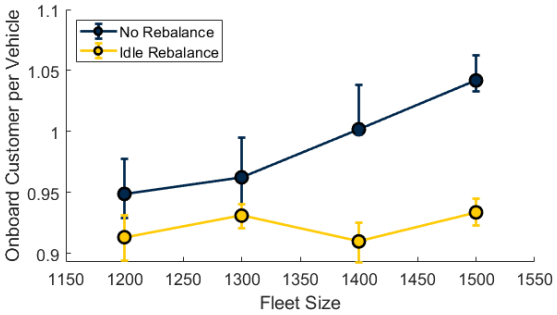


Figure 5.13 Average onboard customer per traveling vehicle

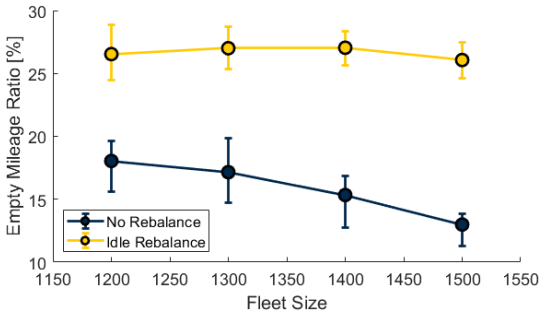


Figure 5.14 Empty vehicle travel mileage ratio (pickup and rebalance)

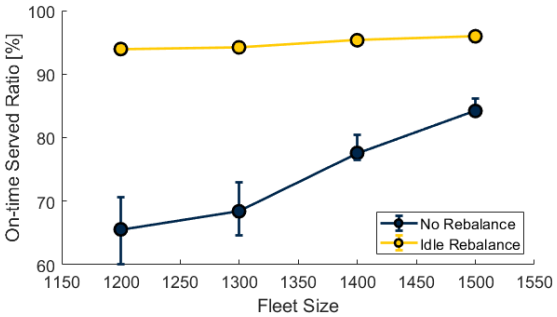


Figure 5.15 Customer served within travel time constraints

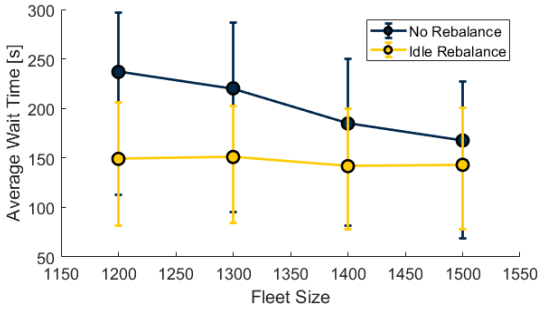


Figure 5.16 Average wait time

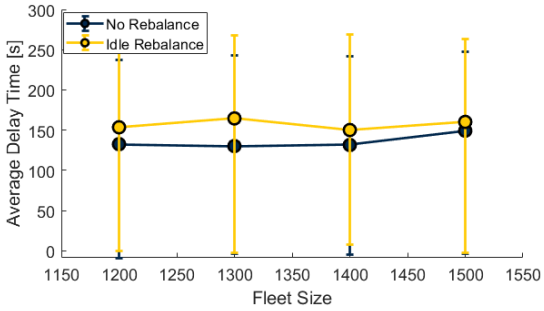


Figure 5.17 Average delay time

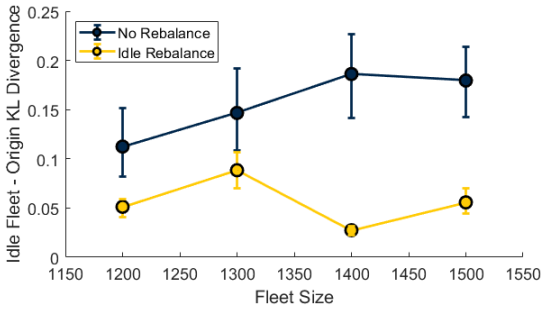


Figure 5.18 KL divergence between idling fleet location distribution and origin distribution

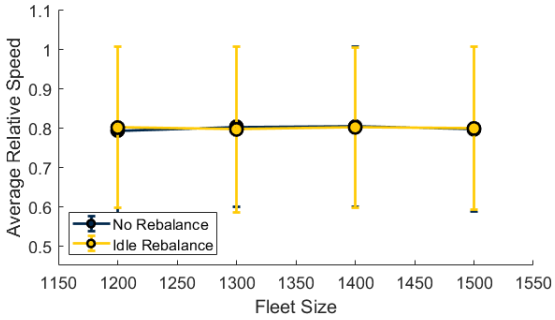


Figure 5.19 Road network average relative speed

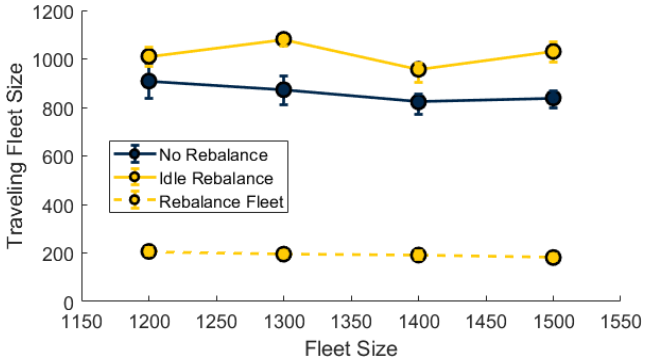


Figure 5.20 Number of MOD vehicles traveling on road

The system performance comparison using different fleet sizes are summarized from Figure 5.10 to Figure 5.20. After applying the rebalancing strategy, the empty vehicle travel mileage increases to more than 25% for all fleet sizes, which lead to an increase in the total travel distance and thus the total fuel consumption to provide the service. Also, due to the extra rebalancing traveling vehicles, the average number of customers per vehicle is reduced. However, the balanced fleet can increase the service quality in terms of the wait time with the rebalancing strategy. The fleet can serve more than 90% of the customers within their travel time constraints with 1,200 vehicles, while only less than 60% of customers‘ constraints are satisfied if the fleet is not rebalanced. The mean wait time is

reduced by more than 37% for 1,200 vehicles and 15% for 1,500 vehicles. The performance improvements are due to closer matching between the idling fleet distribution and the trip origin distribution as indicated by the KL divergence shown in Figure 5.18. With the rebalancing strategy, the Kullback–Leibler (KL) divergence [150] between the idling fleet location distribution and the demand distribution is reduced by 40% to 85% depending on the fleet size. The influence on traffic flow is shown in Figure 5.19 and the amount of traveling vehicles are shown in Figure 5.20. As shown in the figures, the MOD vehicle accounts for 5% of the running vehicles in the road network, which is not large enough to cause a significant change in relative travel speed in the road network as indicated by Figure 5.19.

5.4.3 Parametric Study for Fuel Weight

A parametric study is conducted for the weight between fleet balance and fuel consumption, w_c in (*5.13*). The fleet size is 1,200 for all simulations. The results are shown in Figure 5.21 to Figure 5.29. The baseline is the case of which the fuel weight is 1 (balance

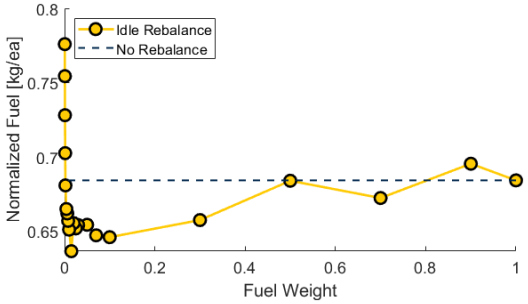


Figure 5.21 Fuel consumption per served customer

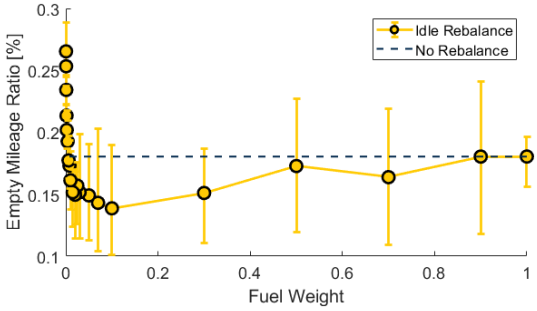


Figure 5.22 Empty vehicle travel mileage ratio (pickup and rebalance)

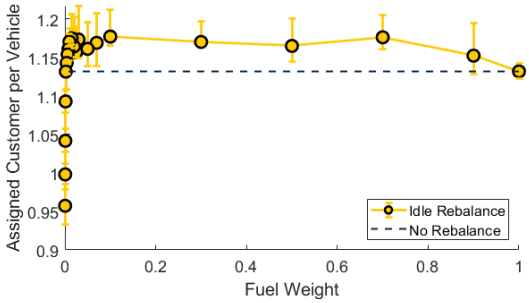


Figure 5.23 Average assigned customer per traveling vehicle

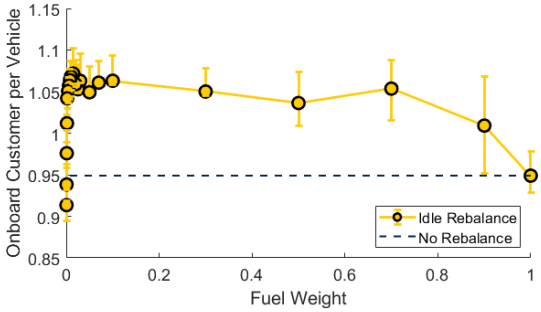


Figure 5.24 Average onboard customer per traveling vehicle

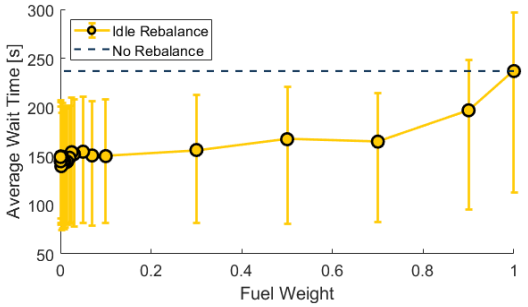


Figure 5.25 Average wait time

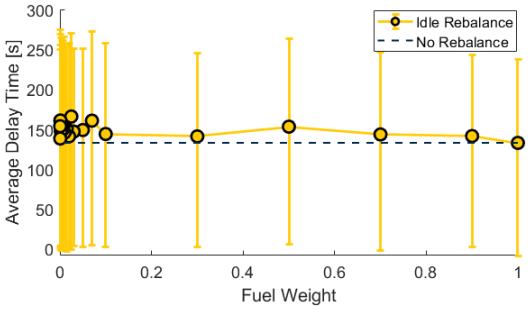


Figure 5.26 Average delay time

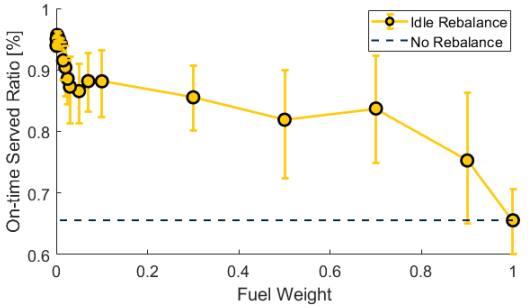


Figure 5.27 Customer served within travel time constraints

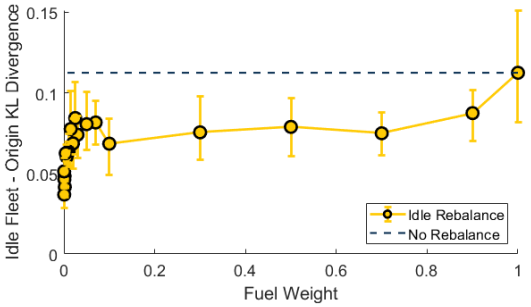


Figure 5.28 KL divergence between idling fleet location distribution and origin distribution

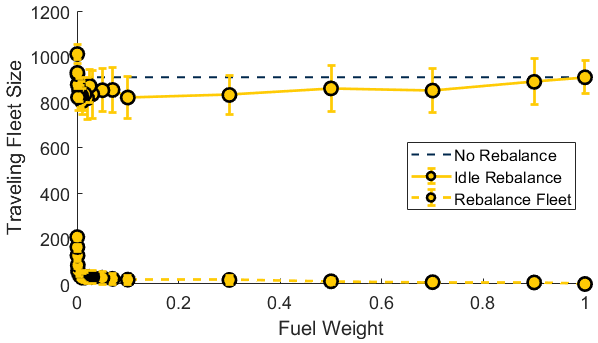


Figure 5.29 Number of MOD vehicles traveling on road

weight 0), corresponding to the case where idling vehicles are not assigned for rebalancing trips. With the decreasing fuel weight, more idling vehicles are assigned for rebalancing trips as shown in Figure 5.29, thus reducing the KL divergence between idling fleet location distribution and origin distribution (Figure 5.28) and increasing the ratio of customers served within travel time constraints (Figure 5.27). However, the shareability and fleet fuel efficiency would first decrease with the fuel weight due to the balanced fleet and then increase due to the additional rebalancing trips as shown in Figure 5.21, Figure 5.23 and Figure 5.24. With more than 90% of customers served within their travel time constraints, the fuel consumed per customer can be reduced by 5% compared with the case when no rebalance trip is assigned. The wait time of customers first decreases with the

decrease of fuel weight but doesn’t show significant changes when the fuel weight is smaller than 0.1. The results indicate that if the fleet size can be accurately estimated, then there exists an optimal fuel weight that can minimize the fleet fuel consumption as well as customers‘ wait time as shown in the existing works where shared rides are not considered [68, 69, 151].

5.5 Conclusion and Future Works

We proposed a road network partition algorithm using the Multidimensional Scaling (MDS) method. The partition is used to discretize the travel demand distribution and select idling rebalancing fleet destinations. An idling fleet rebalancing algorithm is developed using the partition results. With the idling fleet rebalanced according to the trip origin distribution, the expected wait time of customers is reduced, and more customers can be served within the travel time constraints. However, the increased empty vehicle traveling also could increase the total fuel consumption of the fleet with a small value of fuel consumption weight.

CHAPTER 6
Conclusions and Future Works

6.1 Conclusions

The objective of this research is to optimize fuel consumption of connected automated vehicle in an urban environment, including speed trajectory optimization at intersections, data-drive fuel consumption model development, eco-routing algorithm, and eco-MOD (mobility-on-demand) fleet assignment. Chapter 2 introduced a speed trajectory optimization algorithm at signalized intersections taken the additional cost due to turning at intersections into consideration. Chapter 3 introduced a mesoscopic fuel consumption model, and an eco-routing algorithm was developed using the model. Chapter 4 integrated the eco-routing algorithm with a MOD with ride-sharing trip assignment framework. The benefits from trip assignment level and routing strategy level are discussed. Chapter 5 introduced a traffic network partitioning algorithm minimizing the expected waiting time. Also, an idling fleet rebalancing algorithm using the partitioned network is developed.
The contributions include

- A speed trajectory optimization algorithm at signalized intersections with speed and acceleration limit due to left and the right turn is developed. The algorithm can be extended to multiple intersections and multiple vehicles.
- A data-driven fuel consumption model based on real-world driving data and Autonomie fuel consumption simulation turn is developed and analysis of trade-off between travel time and fuel consumption of different routing strategies including fastest route, shortest distance route, eco-route, and travel time constrained eco-route is performed.
- A framework for eco-MOD combining eco-routing strategy and MOD fleet assignment with ride-sharing is developed, and extensive simulation studies are

performed, showing the importance to include fuel consumption in the assignment algorithm to avoid increment in emission.

- An integrated MOD control/simulation framework calibrated using SPMD database being able to recreate Ann Arbor evening rush hour average speed. SUMO is used as the traffic simulator, and Matlab is used as the fleet and demand management component for MOD fleet assignment strategy development and validation
- A traffic network partition algorithm is developed to minimize the expected in-cluster travel time for MOD idling fleet rebalancing.
- An idling fleet rebalancing algorithm for MOD fleet to better meet future travel demands using the traffic network partition results

6.2 Future Works

This dissertation has investigated fuel-saving potentials of connected automated vehicle technologies, from microscopic speed trajectory optimization to macroscopic routing and mobility-on-demand fleet optimization. We showed that the fuel-saving effects of MOD fleet are promising when the fleet is operated properly, which requires further investigations to improve the system performance. The following are several potential directions for future study:

Link-level fuel consumption model should be further developed to use real-time measurement to reduce model parameter uncertainties. The model developed in Chapter 3 is under the assumption that a representative vehicle is used. However, when being applied for routing of MOD fleet or delivery trucks, the powertrain efficiency and characteristics can vary significantly. In addition, for trucks, the vehicle load can vary considerably. Also, different numbers of onboard passengers and weather conditions can affect the fuel consumption of the MOD fleet vehicles. Currently, the model is developed under the Bayesian framework. Thus, instead of point estimation of the parameters, the posterior distributions are obtained. Given the posterior distributions of parameters, the fuel consumption is estimated as conditional expectation. Thus, given real-time measurements

of fuel consumption, the posterior of parameters can be updated using inference algorithms such as MCMC [129], to address model uncertainties in route optimization.

Traffic simulator should be further developed to improve accuracy. The traffic simulator developed can recreate average speed in Ann Arbor. However, there is a significant error in the flow and speed estimation in downtown Ann Arbor. The limitations are due to the approximations made during the model calibration process as well as data availability. The traffic simulator developed for MOD performance evaluation assumes all vehicles belonging to the same class, and route choice is optimized to recreate the average speed in Ann Arbor road network. Thus the influence of bus stops, pedestrian traffic, and traffic signals on traffic flow is not captured by the model. Namely, the simulator can be extended to be multi-modal to improve its accuracy. Since only floating vehicle measurement is available from the SPMD database, the traffic flow is estimated using a simplified data-driven polynomial model of the fundamental diagram using data from SUMO. Thus, the model performance can be further improved by combining other data sources to get the flow rate measurement for multi-modal transportation. More sophisticated link model also has the potential to improve flow estimation.

MOD fleet assignment algorithm should be further developed to consider expected future cost function. The fleet assignment algorithm shows great potential in improving fleet operation efficiency and reducing fuel consumption. However, the assignment strategy only considers the travel demand generated at the current time. The expected future travel demand is addressed by assigning idling fleet to rebalance trips according to the trip origin distribution. One potential approach to account for expected future demands in the assignment stage by modeling the system as a Markov Decision Process (MDP), and the main difficulty is the curse of dimensionality. Recently, model-free MDP and value approximation shows great potential in controlling high dimensional system [152–155], which could be applied to MOD fleet assignment for expected value function estimation.

BIBLIOGRAPHY

[1] L. M. Clements and K. M. Kockelman, "Economic Effects of Automated Vehicles," *Transp. Res. Rec. J. Transp. Res. Board*, 2017.

[2] S. A. E. International, "Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles." SAE International Warrendale, PA, 2016.

[3] C. Leins, "Say Goodbye to Owning Your Own Car: Some industry experts say autonomous vehicles and ride-sharing will lead people to stop buying their own cars.," *U.S. News & World Report News*, 2018.

[4] "Autopilot | Tesla." [Online]. Available: https://www.tesla.com/autopilot. [Accessed: 02-Jun-2019].

[5] "GIVING YOU THE FREEDOM TO GO HANDS-FREE." [Online]. Available: https://www.cadillac.com/world-of-cadillac/innovation/super-cruise. [Accessed: 02-Jun-2019].

[6] "Pilot Assist." [Online]. Available: http://volvo.custhelp.com/app/answers/detail/a_id/9731/~/pilot-assist-ii. [Accessed: 02-Jun-2019].

[7] M. P. Christiansen, M. S. Laursen, R. N. Jørgensen, S. Skovsen, and R. Gislum, "Ground vehicle mapping of fields using LiDAR to enable prediction of crop biomass," *arXiv Prepr. arXiv1805.01426*, 2018.

[8] I. H. S. Automotive, "Emerging technologies: Autonomous cars-not if, but when," *IHS Automot. study, http//press. ihs. com/press-release/automotive/self-driving-cars-moving-industrys-drivers-seat*, 2014.

[9] T. Litman, "Autonomous Vehicle Implementation Predictions: Implications for Transport Planning," *Transp. Res. Board Annu. Meet.*, no. 2014, pp. 36–42, 2014.

[10] P. Bansal and K. M. Kockelman, "Forecasting Americans' long-term adoption of connected and autonomous vehicle technologies," *Transp. Res. Part A Policy Pract.*, vol. 95, pp. 49–63, 2017.

[11] X. Huang, Di. Zhao, and H. Peng, "Empirical Study of DSRC Performance Based on Safety Pilot Model Deployment Data," *IEEE Trans. Intell. Transp. Syst.*, vol. 18,

no. 10, pp. 2619–2628, 2017.

[12] M. Shen, D. Zhao, J. Sun, and H. Peng, “Improving Localization Accuracy in Connected Vehicle Networks Using Rao-Blackwellized Particle Filters: Theory, Simulations, and Experiments,” *Ieee Trans. Intell. Transp. Syst.*, 2016.

[13] I. G. Freedman, E. Kim, and P. A. Muennig, “Autonomous vehicles are cost-effective when used as taxis,” *Inj. Epidemiol.*, vol. 5, no. 1, 2018.

[14] P. M. Bösch, F. Becker, H. Becker, and K. W. Axhausen, “Cost-based analysis of autonomous mobility services,” *Transp. Policy*, vol. 64, no. 4, pp. 76–91, May 2018.

[15] Morgan Stanley, “Shared Mobility on the Road of the Future,” *Morgan Stanley*, 2016. [Online]. Available: https://www.morganstanley.com/ideas/car-of-future-is-autonomous-electric-shared-mobility.

[16] M. Furuhata, M. Dessouky, F. Ordóñez, M. E. Brunet, X. Wang, and S. Koenig, “Ridesharing: The state-of-the-art and future directions,” *Transp. Res. Part B Methodol.*, vol. 57, pp. 28–46, 2013.

[17] L. D. Burns, W. C. Jordan, and B. A. Scarborough, “Transforming personal mobility,” *Earth Inst.*, vol. 431, p. 432, 2013.

[18] S. Shaheen and E. Martin, “Impacts of car2go on Vehicle Ownership, Modal Shift, Vehicle Miles Traveled, and Greenhouse Gas Emissions,” pp. 0–25, 2016.

[19] E. Martin and S. Shaheen, “The impact of carsharing on household vehicle ownership,” 2011.

[20] L. T. Truong, C. De Gruyter, G. Currie, and A. Delbosc, “Estimating the trip generation impacts of autonomous vehicles on car travel in Victoria, Australia,” *Transportation (Amst).*, vol. 44, no. 6, pp. 1279–1292, 2017.

[21] T. D. Chen and K. M. Kockelman, “Carsharing’s life-cycle impacts on energy use and greenhouse gas emissions,” *Transp. Res. Part D Transp. Environ.*, vol. 47, pp. 276–284, 2016.

[22] A. Alessandrini, A. Campagna, P. D. Site, F. Filippi, and L. Persia, “Automated vehicles and the rethinking of mobility and cities,” *Transp. Res. Procedia*, vol. 5, pp. 145–160, 2015.

[23] B. Schoettle and M. Sivak, “A survey of public opinion about connected vehicles in the U.S., the U.K., and Australia,” in *2014 International Conference on Connected Vehicles and Expo, ICCVE 2014 - Proceedings*, 2014.

[24] T. S. Stephens, J. Gonder, Y. Chen, Z. Lin, C. Liu, and D. Gohlke, “Estimated

Bounds and Important Factors for Fuel Use and Consumer Costs of Connected and Automated Vehicles Estimated Bounds and Important Factors for Fuel Use and Consumer Costs of Connected and Automated Vehicles," no. November, 2016.

[25] EIA, "Annual Energy Outlook 2017 with projections to 2050," pp. 1–64, 2017.

[26] A. Vahidi and A. Sciarretta, "Energy saving potentials of connected and automated vehicles," *Transp. Res. Part C Emerg. Technol.*, no. September, pp. 1–22, 2018.

[27] K. Y. Liang, J. Mårtensson, and K. H. Johansson, "Heavy-Duty Vehicle Platoon Formation for Fuel Efficiency," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 4, pp. 1051–1061, 2016.

[28] H. Xia *et al.*, "Field operational testing of ECO-approach technology at a fixed-time signalized intersection," in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 2012.

[29] X. Huang and H. Peng, "Speed Trajectory Planning at Signalized Intersections Using Sequential Convex Optimization," pp. 2992–2997, 2017.

[30] S. Tsugawa, S. Kato, and K. Aoki, "An automated truck platoon for energy saving," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011.

[31] M. P. Lammert, A. Duran, J. Diez, K. Burton, and A. Nicholson, "Effect of Platooning on Fuel Consumption of Class 8 Vehicles Over a Range of Speeds, Following Distances, and Mass," *SAE Int. J. Commer. Veh.*, vol. 7, no. 2014-01–2438, pp. 626–639, 2014.

[32] K. McDonough, I. Kolmanovsky, D. Filev, D. Yanakiev, S. Szwabowski, and J. Michelini, "Stochastic dynamic programming control policies for fuel efficient in-traffic driving," in *2012 American Control Conference (ACC)*, 2012.

[33] S. Li, K. Li, R. Rajamani, and J. Wang, "Model predictive multi-objective vehicular adaptive cruise control," *IEEE Trans. Control Syst. Technol.*, 2011.

[34] S. E. Li, R. Li, J. Wang, X. Hu, B. Cheng, and K. Li, "Stabilizing Periodic Control of Automated Vehicle Platoon With Minimized Fuel Consumption," *IEEE Trans. Transp. Electrif.*, vol. 3, no. 1, pp. 259–271, Mar. 2017.

[35] S. E. Li, "Effect of Pulse-and-Glide Strategy on Traffic Flow for a Platoon of Mixed Automated and Manually Driven Vehicles," vol. 00, pp. 1–14, 2015.

[36] J. Larson, K. Y. Liang, and K. H. Johansson, "A distributed framework for coordinated heavy-duty vehicle platooning," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 419–429, 2015.

[37] D. J. Fagnant and K. Kockelman, “Preparing a nation for autonomous vehicles: Opportunities, barriers and policy recommendations,” *Transp. Res. Part A Policy Pract.*, vol. 77, pp. 167–181, 2015.

[38] A. Stevanovic, J. Stevanovic, K. Zhang, and S. Batterman, “Optimizing Traffic Control to Reduce Fuel Consumption and Vehicular Emissions,” *Transp. Res. Rec. J. Transp. Res. Board*, 2009.

[39] L. Wu, Y. Ci, J. Chu, and H. Zhang, “The influence of intersections on fuel consumption in urban arterial road traffic: A single vehicle test in Harbin, China,” *PLoS One*, 2015.

[40] J. Y. K. Luk, “TWO TRAFFIC-RESPONSIVE AREA TRAFFIC CONTROL METHODS: SCAT AND SCOOT.,” *Traffic Eng. Control*, 1984.

[41] J. Misener, S. Shladover, and S. Dickey, “Investigating the potential benefits of broadcasted signal phase and timing (SPAT) data under IntelliDriveSM,” in *ITS America Annual Meeting, Houston, Texas*, 2010.

[42] L. Evans, “DRIVER BEHAVIOR EFFECTS ON FUEL CONSUMPTION IN URBAN DRIVING.,” *Hum. Factors*, 1979.

[43] B. Asadi and A. Vahidi, “Predictive cruise control: Utilizing upcoming traffic signal information for improving fuel economy and reducing trip time,” *IEEE Trans. Control Syst. Technol.*, 2011.

[44] H. Rakha and R. K. Kamalanathsharma, “Eco-driving at signalized intersections using V2I communication,” in *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, 2011, pp. 341–346.

[45] G. Mahler and A. Vahidi, “An optimal velocity-planning scheme for vehicle energy efficiency through probabilistic prediction of traffic-signal timing,” *IEEE Trans. Intell. Transp. Syst.*, 2014.

[46] X. He, H. X. Liu, and X. Liu, “Optimal vehicle speed trajectory on a signalized arterial with consideration of queue,” *Transp. Res. Part C Emerg. Technol.*, 2015.

[47] E. Ozatay, G. Rizzoni, S. Onori, and U. Ozguner, “ANALYTICAL SOLUTION TO THE MINIMUM FUEL CONSUMPTION OPTIMIZATION,” no. 1, pp. 1–10, 2016.

[48] A. Hofleitner, R. Herring, and A. Bayen, “Arterial travel time forecast with streaming data: A hybrid approach of flow modeling and machine learning,” *Transp. Res. Part B Methodol.*, vol. 46, no. 9, pp. 1097–1122, Nov. 2012.

[49] J. Schulman, J. Ho, A. Lee, I. Awwal, H. Bradlow, and P. Abbeel, “Finding Locally

Optimal, Collision-Free Trajectories with Sequential Convex Optimization," in *Robotics: Science and Systems IX*, 2013.

[50] G. O. Inc., "Gurobi Optimizer reference manual," *Www.Gurobi.Com*, vol. 6, p. 572, 2014.

[51] K. Boriboonsomsin, M. Barth, S. Member, W. Zhu, and A. Vu, "ECO-Routing Navigation System based on Multi- Source Historical and Real-Time Traffic Information," *Network*, vol. 13, no. 4, pp. 1694–1704, 2012.

[52] K. Ahn and H. A. Rakha, "Network-wide impacts of eco-routing strategies: A large-scale case study," *Transp. Res. Part D Transp. Environ.*, vol. 25, pp. 119–130, 2013.

[53] J. Sun and H. X. Liu, "Stochastic Eco-routing in a Signalized Traffic Network," *Transp. Res. Procedia*, vol. 7, pp. 110–128, 2015.

[54] L. Guo, S. Huang, and A. W. Sadek, "An Evaluation of Environmental Benefits of Time-Dependent Green Routing in the Greater Buffalo–Niagara Region," *J. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 18–30, 2013.

[55] A. Rousseau, P. Sharer, and F. Besnier, "Feasibility of Reusable Vehicle Modeling: Application to Hybrid Vehicles," *SAE Tech. Pap.*, no. 2004-01–1618, p. 12, 2004.

[56] J. Kwon, A. Rousseau, and P. Sharer, "Analyzing the uncertainty in the fuel economy prediction for the EPA MOVES binning methodology," *SAE Int.*, 2007.

[57] M. Kubicka *et al.*, "Performance of current eco-routing methods," *IEEE Intell. Veh. Symp. Proc.*, vol. 2016-Augus, pp. 472–477, 2016.

[58] W. Zeng, D. Candidate, T. Miwa, and T. Morikawa, "Application of machine learning and heuristic k- shortest path algorithm to eco-routing problem with travel time constraint," pp. 1–18, 2016.

[59] M. Masikos, M. Theologou, K. Demestichas, and E. Adamopoulou, "Machine-learning methodology for energy efficient routing," *IET Intell. Transp. Syst.*, 2014.

[60] Y. Chen, L. Zhu, J. Gonder, S. Young, and K. Walkowicz, "Data-driven fuel consumption estimation: A multivariate adaptive regression spline approach," *Transp. Res. Part C Emerg. Technol.*, vol. 83, pp. 134–145, 2017.

[61] H. G. Sung, "Gaussian Mixture Regression and Classification," Rice University, 2004.

[62] "Shared Mobility On The Road Of The Future." [Online]. Available: https://www.forbes.com/sites/morganstanley/2016/07/20/shared-mobility-on-the-road-of-the-future/#a3a6c851cae8. [Accessed: 20-Mar-2019].

[63] C. D. Porter, A. Brown, J. DeFlorio, E. McKenzie, W. Tao, and L. Vimmerstedt, “Effects of travel reduction and efficient driving on transportation: Energy use and greenhouse gas emissions,” 2013.

[64] J. Alonso-mora, A. Wallar, and D. Rus, “Predictive Routing for Autonomous Mobility-on-Demand Systems with Ride-Sharing,” pp. 3583–3590, 2017.

[65] J. Alonso-Mora, S. Samaranayake, A. Wallar, E. Frazzoli, and D. Rus, “On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment,” *Proc. Natl. Acad. Sci.*, vol. 114, no. 3, pp. 462–467, Jan. 2017.

[66] F. Miao *et al.*, “Taxi Dispatch with Real-Time Sensing Data in Metropolitan Areas: A Receding Horizon Control Approach,” *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 463–478, 2016.

[67] F. Rossi, R. Zhang, and M. Pavone, “Congestion-Aware Randomized Routing in Autonomous Mobility-on-Demand Systems,” *Transp. Res. Part B Methodol.*, vol. 99, pp. 1–29, Sep. 2016.

[68] R. Zhang, F. Rossi, and M. Pavone, “Analysis , Control , and Evaluation of Mobility-on-Demand Systems : a Queueing-Theoretical Approach,” pp. 1–10, 2018.

[69] K. Treleaven, M. Pavone, and E. Frazzoli, “Models and Asymptotically Optimal Algorithms for Pickup and Delivery Problems on Roadmaps (submitted),” *Decis. Control (CDC), 2012 51st IEEE Conf.*, pp. 5691–5698, 2012.

[70] M. Čáp, S. Vajna, and E. Frazzoli, “Fleet Sizing in Vehicle Sharing Systems with Service Quality Guarantees,” 2018.

[71] A. Wallar, J. Alonso-mora, and D. Rus, “Optimizing Vehicle Distributions and Fleet Sizes for Mobility-on-Demand.”

[72] M. M. Vazifeh, P. Santi, G. Resta, S. H. Strogatz, and C. Ratti, “Addressing the minimum fleet problem in on-demand urban mobility,” *Nature*, vol. 557, no. 7706, pp. 534–538, May 2018.

[73] R. Tachet *et al.*, “Scaling Law of Urban Ride Sharing,” *Sci. Rep.*, vol. 7, p. 42868, 2017.

[74] X. Huang and H. Peng, “Eco-Routing based on a Data Driven Fuel Consumption Model,” 2018, no. i.

[75] J. Auld, M. Hope, H. Ley, V. Sokolov, B. Xu, and K. Zhang, “POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations,” *Transp. Res. Part C Emerg. Technol.*, vol. 64, pp. 101–116, 2016.

[76] D. Bezzina and J. Sayer, “Safety pilot model deployment: Test conductor team report,” *Rep. No. DOT HS*, vol. 812, no. June, p. 171, 2014.

[77] D. Krajzewicz and J. Erdmann, “Recent Development and Applications of SUMO–Simulation of Urban MObility,” *Int. J. …*, vol. 5, no. 3, pp. 128–138, 2012.

[78] M. Sánchez, J. C. Cano, and D. Kim, “Predicting traffic lights to improve urban traffic fuel consumption,” in *ITST 2006 - 2006 6th International Conference on ITS Telecommunications, Proceedings*, 2007.

[79] M. Barth, S. Mandava, K. Boriboonsomsin, and H. Xia, “Dynamic ECO-driving for arterial corridors,” in *2011 IEEE Forum on Integrated and Sustainable Transportation Systems, FISTS 2011*, 2011.

[80] B. HomChaudhuri, A. Vahidi, and P. Pisu, “A fuel economic model predictive control strategy for a group of connected vehicles in urban roads,” in *Proceedings of the American Control Conference*, 2015.

[81] B. HomChaudhuri, R. Lin, and P. Pisu, “Hierarchical control strategies for energy management of connected hybrid electric vehicles in urban roads,” *Transp. Res. Part C Emerg. Technol.*, 2016.

[82] M. Miyatake, M. Kuriyama, and Y. Takeda, “Theoretical study on eco-driving technique for an electric vehicle considering traffic signals,” in *Proceedings of the International Conference on Power Electronics and Drive Systems*, 2011.

[83] E. Ozatay, U. Ozguner, D. Filev, and J. Michelini, “Analytical and numerical solutions for energy minimization of road vehicles with the existence of multiple traffic lights,” in *Proceedings of the IEEE Conference on Decision and Control*, 2013.

[84] G. De Nunzio, C. C. De Wit, P. Moulin, and D. Di Domenico, “Eco-driving in urban traffic networks using traffic signals information,” *Int. J. Robust Nonlinear Control*, 2016.

[85] E. Yao and Y. Song, “Study on Eco-Route Planning Algorithm and Environmental Impact Assessment,” *J. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 42–53, 2013.

[86] H. M. A. Aziz, S. V Ukkusuri, H. M. Abdul Aziz, and S. V Ukkusuri, “Exploring the trade-off between greenhouse gas emissions and travel time in daily travel decisions: Route and departure time choices,” *Transp. Res. Part D Transp. Environ.*, 2014.

[87] W. Li, L. Yang, L. Wang, X. Zhou, R. Liu, and Z. Gao, “Eco-reliable path finding in time-variant and stochastic networks,” *Energy*, vol. 121, pp. 372–387, 2017.

[88] C. C. Lu, J. Liu, Y. Qu, S. Peeta, N. M. Rouphail, and X. Zhou, "Eco-system optimal time-dependent flow assignment in a congested network," *Transp. Res. Part B Methodol.*, vol. 94, pp. 217–239, 2016.

[89] S. Sugawara and D. Niemeier, "How Much Can Vehicle Emissions Be Reduced?: Exploratory Analysis of an Upper Boundary Using an Emissions-Optimized Trip Assignment," *Transp. Res. Rec. J. Transp. Res. Board*, 2002.

[90] Nagurney.A and Dong.J, "A multiclass, multicriteria traffic network equilibrium model with elastic demand," *Transp. Res. B*, 2001.

[91] M. Levin W, M. Duell, and S. Waller Travis, "Effect of Road Grade on Networkwide Vehicle Energy Consumption and Ecorouting," *Transp. Res. Rec. J. Transp. Res. Board*, no. 2427, p. pp 26–33, 2014.

[92] Y. Nie and Q. Li, "An eco-routing model considering microscopic vehicle operating conditions," *Transp. Res. Part B Methodol.*, vol. 55, pp. 154–170, 2013.

[93] A. Elbery, H. Rakha, M. Y. ElNainay, W. Drira, and F. Filali, "Eco-routing: An Ant Colony based Approach," *Proc. Int. Conf. Veh. Technol. Intell. Transp. Syst.*, no. June, pp. 31–38, 2016.

[94] C. F. Minett, A. M. Salomons, W. Daamen, B. Van Arem, and S. Kuijpers, "Eco-routing: Comparing the fuel consumption of different routes between an origin and destination using field test speed profiles and synthetic speed profiles," in *2011 IEEE Forum on Integrated and Sustainable Transportation Systems, FISTS 2011*, 2011.

[95] H. a. Rakha, K. Ahn, and K. Moran, "INTEGRATION Framework for Modeling Eco-routing Strategies: Logic and Preliminary Results," *Int. J. Transp. Sci. Technol.*, vol. 1, no. 3, pp. 259–274, 2012.

[96] K. Ahn and H. Rakha, "The effects of route choice decisions on vehicle energy consumption and emissions," *Transp. Res. Part D Transp. Environ.*, vol. 13, no. 3, pp. 151–167, 2008.

[97] D. J. Bertsimas and D. Simchi-Levi, "A New Generation of Vehicle Routing Research: Robust Algorithms, Addressing Uncertainty," *Oper. Res.*, vol. 44, no. 2, pp. 286–304, 1996.

[98] E. Lam and P. Van Hentenryck, "A branch-and-price-and-check model for the vehicle routing problem with location congestion," *Constraints*, vol. 21, no. 3, pp. 394–412, 2016.

[99] R. W. Bent and P. Van Hentenryck, "Scenario-Based Planning for Partially Dynamic Vehicle Routing with Stochastic Customers," *Oper. Res.*, vol. 52, no. 6,

pp. 977–987, 2004.

[100] A. Maheo, P. Kilby, and P. Van Hentenryck, “Benders Decomposition for the Design of a Hub and Shuttle Public Transit System,” no. January 2018, 2015.

[101] G. Laporte, “The vehicle routing problem: An overview of exact and approximate algorithms,” *Eur. J. Oper. Res.*, 1992.

[102] N. Agatz, A. Erera, M. Savelsbergh, and X. Wang, “Optimization for dynamic ride-sharing: A review,” *Eur. J. Oper. Res.*, vol. 223, no. 2, pp. 295–303, 2012.

[103] W. Herbawi and M. Weber, “A Genetic and Insertion Heuristic Algorithm for Solving the Dynamic Ridematching Problem with Time Windows,” *Gecco 2012*, pp. 385–392, 2012.

[104] D. Teodorović and M. Dell ’orco, “Bee Colony Optimization – A cooperative learning approach to complex transportation problems,” *Adv. OR AI Methods Transp.*, 2015.

[105] N. Masoud and R. Jayakrishnan, “A decomposition algorithm to solve the multi-hop Peer-to-Peer ride-matching problem,” *Transp. Res. Part B Methodol.*, vol. 99, no. May, pp. 1–29, May 2017.

[106] H. Hosni, J. Naoum-Sawaya, and H. Artail, “The shared-taxi problem: Formulation and solution methods,” *Transp. Res. Part B Methodol.*, vol. 70, pp. 303–318, 2014.

[107] P. Santi, G. Resta, M. Szell, S. Sobolevsky, S. H. Strogatz, and C. Ratti, “Quantifying the benefits of vehicle pooling with shareability networks.,” *Proc. Natl. Acad. Sci.*, vol. 111, no. 37, pp. 13290–4, 2014.

[108] T. Zhou, C. Osorio, and E. Fields, “A data-driven discrete simulation-based optimization algorithm for large-scale two-way car-sharing network design.”

[109] A. Prorok and V. Kumar, “Privacy-Preserving Vehicle Assignment for Mobility-on-Demand Systems,” 2017.

[110] M. W. Levin, “Congestion-aware system optimal route choice for shared autonomous vehicles,” *Transp. Res. Part C Emerg. Technol.*, vol. 82, pp. 229–247, 2017.

[111] R. Zhang, F. Rossi, and M. Pavone, “Model Predictive Control of Autonomous Mobility-on-Demand Systems,” *Int. Conf. Robot. Autom.*, pp. 1382–1389, 2016.

[112] M. H. Hasan, P. Van Hentenryck, C. Budak, J. Chen, and C. Chaudhry, “Community-Based Trip Sharing for Urban Commuting,” 2018.

[113] S. Han, U. Topcu, and G. J. Pappas, “Quantification on the efficiency gain of automated ridesharing services,” *2017 Am. Control Conf.*, pp. 3560–3566, 2017.

[114] F. Miao, S. Han, A. M. Hendawi, M. E. Khalefa, J. A. Stankovic, and G. J. Pappas, “Data-driven distributionally robust vehicle balancing using dynamic region partitions,” *Proc. 8th Int. Conf. Cyber-Physical Syst. - ICCPS ’17*, pp. 261–271, 2017.

[115] X. Chen and G. J. Pappas, “Hierarchical Data-Driven Vehicle Dispatch and Ride-sharing,” no. October, 2017.

[116] J. Thai, C. Yuan, and A. M. Bayen, “Resiliency of Mobility-as-a-Service Systems to Denial-of-Service Attacks,” *IEEE Trans. Control Netw. Syst.*, vol. PP, no. 99, 2016.

[117] A. Wallar, M. van der Zee, J. Alonso-Mora, and D. Rus, “Vehicle Rebalancing for Mobility-on-Demand Systems with Ride-Sharing.”

[118] A. Galip Ulsoy, H. Peng, and M. Çakmakcı, *Automotive control systems*. 2012.

[119] M. D. L. Paul R. Windover, Russel J. Owens, Terry M. Levinson, “Stop and Restart effects on modern vehicle starting system components,” *Argonne Natl. Lab.*, 2015.

[120] S. E. Li and H. Peng, “Strategies to minimize the fuel consumption of passenger cars during car-following scenarios,” *Proc. Inst. Mech. Eng. Part D J. Automob. Eng.*, 2012.

[121] C. Ericson, B. Westerberg, and R. Egnell, “Transient Emission Predictions With Quasi Stationary Models,” in *SAE Powertrain & Fluid Systems Conference & Exhibition*, 2005.

[122] Y. H. Lee and W. Deng, “A simulation study of vehicle curve speed control system,” in *ASME 2006 International Mechanical Engineering Congress and Exposition*, 2006, pp. 163–171.

[123] R. Akcelik and D. C. Biggs, “Acceleration Profile Models for Vehicles in Road Traffic,” *Transp. Sci.*, 1987.

[124] J. Wang, K. Dixon, H. Li, and J. Ogle, “Normal Deceleration Behavior of Passenger Vehicles at Stop Sign-Controlled Intersections Evaluated with In-Vehicle Global Positioning System Data,” *Transp. Res. Rec. J. Transp. Res. Board*, 2005.

[125] R. H. Wortman and J. S. Matthias, *An evaluation of driver behavior at signalized intersections*. Arizona Department of Transportation Phoenix, Arizona, 1983.

[126] A. Richards and J. P. How, “Aircraft trajectory planning with collision avoidance

using mixed integer linear programming," in *Proceedings of the 2002 American Control Conference (IEEE Cat. No.CH37301)*, 2002.

[127] D. K. Smith and A. Schrijver, "Theory of Linear and Integer Programming.," *J. Oper. Res. Soc.*, 2006.

[128] E. A. B. da Silva and G. V. Mendonca, "Digital Image Processing," in *The Electrical Engineering Handbook*, 2005.

[129] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[130] D. P. Bertsekas, "Dynamic Programming and Optimal Control 3rd Edition , Volume II by Chapter 6 Approximate Dynamic Programming Approximate Dynamic Programming," *Control*, vol. II, pp. 1–200, 2010.

[131] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "Optics: Ordering points to identify the clustering structure," *ACM Sigmod Rec.*, pp. 49–60, 1999.

[132] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," *Proc. 2nd Int. Conf. Knowl. Discov. Data Min.*, pp. 226–231, 1996.

[133] C. Bron and J. Kerbosch, "Algorithm 457: finding all cliques of an undirected graph," *Commun. ACM*, vol. 16, no. 9, pp. 575–577, 1973.

[134] S. Ma, Y. Zheng, and O. Wolfson, "Real-Time City-Scale Taxi Ridesharing," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 7, pp. 1782–1795, 2015.

[135] J. Auld and A. K. Mohammadian, "Activity planning processes in the Agent-based Dynamic Activity Planning and Travel Scheduling (ADAPTS) model," *Transp. Res. Part A Policy Pract.*, 2012.

[136] M. Maciejewski, "A Comparison of Microscopic Traffic Flow Simulation," *Transp. Probl.*, vol. 5, no. 4, 2010.

[137] K. Ballantyne *et al.*, "Toward a Systematic Approach to the Design and Evaluation of Automated Mobility-on-Demand Systems : A Case Study in Singapore," pp. 0–16, 2014.

[138] M. Pavone, S. L. Smith, E. Frazzoli, and D. Rus, "Robotic load balancing for mobility-on-demand systems The," pp. 0–25, 2012.

[139] D. K. George and C. H. Xia, "Fleet-sizing and service availability for a vehicle rental system via closed queueing networks," *Eur. J. Oper. Res.*, vol. 211, no. 1, pp. 198–207, 2011.

[140] R. Zhang and M. Pavone, "Control of robotic mobility-on-demand systems: A queueing-theoretical perspective," *Int. J. Rob. Res.*, vol. 35, no. 1–3, pp. 186–203, 2016.

[141] F. J. Massey, "The Kolmogorov-Smirnov Test for Goodness of Fit," *J. Am. Stat. Assoc.*, 1951.

[142] R. Zhang, F. Rossi, and M. Pavone, "Routing Autonomous Vehicles in Congested Transportation Networks: Structural Properties and Coordination Algorithms," *Proc. Robot. Sci. Syst.*, 2016.

[143] J. B. Kruskal and M. Wish, *Multidimensional Scaling*, vol. 31. 1978.

[144] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian Data Analysis*. 2004.

[145] R. Zhang, K. Spieser, E. Frazzoli, and M. Pavone, "Models, algorithms, and evaluation for autonomous mobility-on-demand systems," *Proc. Am. Control Conf.*, vol. 2015-July, pp. 2573–2587, 2015.

[146] R. Bellman, "On a Routing Problem," *Q. Appl. Math.*, vol. 16, pp. 87–90, 1958.

[147] F. Rossi, R. Zhang, Y. Hindy, and M. Pavone, "Routing autonomous vehicles in congested transportation networks: structural properties and coordination algorithms," *Auton. Robots*, May 2018.

[148] P. Raghavan and C. D. Tompson, "Randomized rounding: A technique for provably good algorithms and algorithmic proofs," *Combinatorica*, vol. 7, no. 4, pp. 365–374, Dec. 1987.

[149] M. E. J. Newman, "Spectral methods for network community detection and graph partitioning," 2013.

[150] Kullback. S and Leibler. R. A, "ON INFORMATION AND SUFFICIENCY," *Source Ann. Math. Stat.*, 1951.

[151] M. Pavone, "Autonomous Mobility-on-Demand Systems for Future Urban Mobility," in *Autonomous Driving*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 387–404.

[152] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-End Training of Deep Visuomotor Policies," vol. 17, pp. 1–40, 2015.

[153] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-Dimensional Continuous Control Using Generalized Advantage Estimation," pp. 1–14, 2015.

[154] M. P. Deisenroth and C. E. Rasmussen, "PILCO: A Model-Based and Data-Efficient

Approach to Policy Search," *Proc. Int. Conf. Mach. Learn.*, pp. 465–472, 2011.

[155] M. Li *et al.*, "Efficient Ridesharing Order Dispatching with Mean Field Multi-Agent Reinforcement Learning," 2019.