Congestion-Aware Operation of Coordinated Autonomous Mobility-on-Demand Systems

Master Thesis

Author(s):
Lu, Chengqi

Publication date:
2019-08

Permanent link:
https://doi.org/10.3929/ethz-b-000365682

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute for
Dynamic Systems and Control
IDSC
Institut für Dynamische Systeme
und Regelungstechnik

陆承骐 Chengqi Lu

Congestion-Aware Operation of Coordinated Autonomous Mobility-on-Demand Systems

Master's Thesis

Institute for Dynamic Systems and Control
Swiss Federal Institute of Technology (ETH) Zurich

Supervision

Claudio Ruch
Sebastian Hörl
Prof. Dr. Emilio Frazzoli
Prof. Dr. Kay Axhausen

August 2019

IDSC-EFCR-MT-05

Abstract

With the advent of self-driving cars, Autonomous Mobility on Demand (AMoD) may become a very popular mode of transport in the near future because of its low price, convenience and other benefits. This study will explore the congestion effect of introducing a large fleet of autonomous vehicles to cities on a realistic simulation environment and will develop congestion-aware operation strategies for the fleet decision-making process. Refuting some previous results, our results show that replacing private car trips with AMoD rides can improve the overall traffic condition if a good operating strategy is used. Furthermore, our results also indicate that traffic can still be improved when only a fraction of the private car trips are replaced by AMoD rides. The higher the replacement rate is, the better the overall traffic will be. The study concludes with the identification of multiple future research directions that may bring us one step closer to the mobility of future.

Keywords: Autonomous Mobility-on-Demand (AMoD), Dynamic Vehicle Routing Problem, Congestion-aware, Coordinated Operation, Interaction between Autonomous Vehicles and private cars.

Contents

Chapter 1

Introduction

With the rising of autonomous driving technology, Autonomous Mobility on Demand (AMoD) has become a popular topic nowadays and is considered as an alternative to conventional private vehicles. Because autonomous vehicles can drive between places on their own, a vehicle can be shared among several users and therefore the upfront cost of AMoD can be significantly lower than private cars. Similar to the taxi and ride hailing services that are currently available, users can simply request for a ride and then a car will arrive to them and take them to the destination conveniently. But unlike the taxi or ride hailing available today, AMoD can offer the same or even better service at a fraction of the cost, because no human drivers are needed. According to a study, up to 88% of the fare collected are used to pay the driver[1]. In addition, a well-designed autonomous driving system can drive in a more energy efficient way than human beings and this further reduce the cost. Based on the findings in [1], it is possible that the fare to take an autonomous taxi can be comparable or even slightly cheaper than the cost of driving private vehicle (e.g., fuel, maintenance, depreciation, insurance, parking). As a result, it is likely that many people will switch from driving private cars to using AMoD as their mode of transport in the near future when the self-driving technology is fully mature. With that, an interesting question arises: how will AMoD influence the traffic in our cities? Will it improve the traffic, or will it intensify the congestion problem that is already bothering residents in many cities around the globe?

Congestion is a headache for many cities around the world and it is getting worse due to increasing travel demands as a result of on-going urbanization process. For example, it is estimated that around 20 billion US dollar may be lost on the traffic jam each year in New York City [2]. In addition, traffic congestion also increases the amount of harmful gas emitted by vehicles, which is undermining human's health. According to WHO, one third of deaths from stroke, lung cancer and heart disease are due to air pollution [3]. Improving traffic and tackling congestion is therefore one of the major tasks for the engineers and city planners who design the future of mobility.

AMoD is among the list of the future mobility and it is considered as an effective way to solve the traffic congestion problem because we can greatly reduce the total number of vehicles if people are using AMoD instead of owning private cars. This has been mentioned as the benefit of AMoD to the traffic in many studies on the autonomous driving topic. Reasonable as it sounds to be, this is not as straightforward. While total number of vehicles is reduced, the number of vehicles running on the road at any given moment, however, cannot be reduced if same travel demands are to be fulfilled. In fact, there will be even more vehicles running on the road and more Vehicle Miles Traveled (VMT) because of empty repositioning drives between travel demands. This can be a major issue during rush hours, as adding vehicles to traffic jam can severely intensify the congestion and that is exactly opposite to the hope of improving traffic condition with autonomous vehicles.

Fortunately, however, AMoD does possess advantage that can potentially turn the tide to its favor. With private vehicles, the traffic tends to converge to Dynamic User Equilibrium (DUE) or Wardrop's Equilibrium [4], which is a Nash Equilibrium and is usually far from the system optimum. Under such a circumstance, the social benefits are low, and congestion can still happen even if the system has adequate flow capacity. This means building new roads and enlarging capacity of current network may not be enough to remove congestion. A fleet of autonomous vehicles, on the other hand, can improve current situation by behaving more cooperatively and moving the whole traffic system from the Nash Equilibrium towards a social equilibrium, in which the overall benefits of the users can be greatly improved. Apart from routing aspect, the dispatching and rebalancing algorithm also plays an active part in reducing congestion. According to a commonly used traffic model developed by Bureau of Public Road (BPR) [5], an additional car to a congested road will add much more to the congestion than that on an empty road. As a result, flattening the peak to some extent will also be effective in tackling congestion. With AMoD, this can be achieved by a good operating policy that can decide which request to serve first and whether to serve a request immediately or deliberately delay the service by a short period of time.

Will the positive effects brought by the AMoD overweight those negative counterparts? In this study, we will look at how a congestion-aware AMoD system will influence the traffic and examine if AMoD can be introduced to cities at a large scale. In particular, we will focus on the effect of replacing private car trips with AMoD rides. Before reaching the conclusion, we will also discuss some potential further research directions on this topic, such as interaction between AMoD and public transport and the influence of induced travel demands by AMoD.

Chapter 2

Literature Review

Fundamentals of AMoD Systems

In recent years, many studies have been conducted on large scale Autonomous Mobility-on-Demand Systems. One focus area of the study is the fundamentals of AMoD systems, where theoretical performance of the AMoD fleet are explored and vehicle positioning problems are studied. Imbalance between the distribution of origins and destinations of travel demand is one of the key factors that limit the performance of an autonomous vehicle fleet and this is also the reason why a good vehicle positioning strategy is very important in AMoD operation. In the article [6], this kind of imbalance is analyzed in a 2-D Euclidean space and has been characterized by the Earth Mover's distance (or Wasserstein Distance, more details can be found in [7]). Identifying the imbalance is very important when choosing the fleet size of AMoD. As indicated in article [6] and [8], failing to take imbalance into consideration will lead to poor estimation of the AMoD fleet size and service quality (i.e., high accessibility and low waiting time). Because of this imbalance, empty vehicles need to travel from popular travel destination regions to the popular origin (i.e., the starting point) regions in order to make the AMoD system stay in balance. Therefore, this process is often referred to as rebalance process in the literature. In order to generate a good rebalancing strategy, several studies use fluidic models to optimize the rebalance process. For example, in article [9] a station based model for the city-scale network is constructed and linear programming is performed to generate optimal rebalance plan under the constant demand pattern. This article also introduced a real time rebalancing strategy, Adaptive Real-Time Rebalancing Policy, that can respond to the time-varying demand pattern. In another article [10] based on the same idea of [9], time-varying demand pattern is included in a different way that allows an optimal rebalancing plan to be calculated under continuously changing travel demands. In addition to the fluidic models, another method, Queueing-Theoretical approach, is also commonly used in the literature. In article [11], queueing-theoretical approach based on a closed Jackson network is used to model the AMoD system. The model is also station-based, like in [9], but instead of relying on fluidic assumption, vehicles are considered as discrete object. Because of the discrete nature of vehicles in this model, the author of article [11] has also carried out simulations with real world data and has estimated the number of autonomous vehicles needed to serve cities like New York (Manhattan) and Hangzhou without having passengers waiting for too long. Studies on more advanced rebalancing strategies are still on-going and many interesting strategies, such as adding predictive element [12] or using a model-free policy [13], are available. Studies on fundamentals of the AMoD systems provide a general guideline for the operation of an autonomous vehicle fleet. These studies, however, do not take congestion into consideration or treat it as an exogenous effect. Therefore, for a city scale application of AMoD system, especially in heavily populated cities, results from these studies may not be fully applicable.

Treating Congestion as an Endogenous Effect

When congestion effect is treated as an endogenous effect, the problem becomes much more complex and interesting results arise from different studies. One common assumption in studies on the topic of endogenous congestion effect of AMoD operation is that all vehicles running in the network are controlled by the operator (see Figure 2.1). In one study [14], the author has found out that the performance of an autonomous vehicles system is very sensitive to the fleet size when congestion effect is considered as an endogenous effect. Not only the waiting time will be influenced by the fleet size, network traffic condition will also vary under different fleet sizes. In particular, when reducing the fleet size from an adequate value, the traffic situation will become worse, which is contrary to the common belief that the less vehicle we have the less congestion there will be. The complexity of the problem is also reflected in the computational workload. Linear programming (integer linear programming) is one of the most frequently used tools in the traffic assignment problem for AMoD system. In [15], space-time-state networks is used to model the traffic and several techniques are used to formulate and solve the linear programming problem. In another article [16], the author developed an optimal routing algorithm based on link transmission model and use linear programming to solve the problem. Although these algorithms solve for the optimal solution, as the size of the network and vehicle fleet grows, the complexity of the problem increase exponentially. As a result, these algorithms are only feasible to run on simplified and very small network with limited number of vehicles. In a recent study [17], a rolling-horizon optimization strategy that combines both offline and online calculation is developed to increase the tractability of the problem and it is feasible to run the algorithm on Manhattan's network with thousands of vehicles.

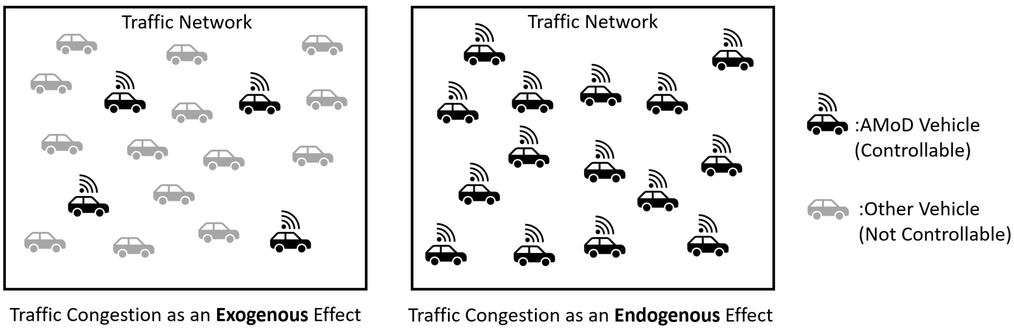

Figure 2.1: Exogenous and Endogenous Effects of Congestion

In addition to the complexity, contradictory results exist on the question whether AMoD will improve or worsen the traffic. In fact, people already have different opinions on the existing ride-sharing service operated by companies like Uber and Lyft (which can be treated as a kind of mobility-on-demand service). For example, two studies, [18] and [19], yield completely different results on the effect of ride-sharing service on traffic congestion. Similarly, such disagreement also happens to AMoD. In article [20], the author shows that, in a capacity symmetric network, with an appropriate algorithm, rebalancing vehicles will not lead to any increase in the congestion, given that there exists a congestion free customer flow. In a follow up study [21], the author further introduced a randomized congestion-aware routing algorithm to improve the speed of calculation without compromising the performance of the algorithm. In other words, as long as the flow capacity of the network is large enough for the travel demand and a good routing algorithm can find an congestion free arrangement for all the travel demand, then those additional empty drive because of the rebalance process will not cause any congestion under a good rebalancing strategy. These studies demonstrate a positive outlook for the AMoD system. The simplified traffic model and reliance on the constant travel demand, however, are the limitations of these studies. On the other hand, a study on the AMoD with a more realistic simulation [22] has pointed out that given the same network capacity, introducing AMoD service in cities will only aggravate the traffic congestion problem we are facing today. The reason for the additional congestion is exactly due to the additional drive brought by the re-positioning vehicles.

Other Factors

Furthermore, there are other factors, although outside the focus of this study, worth noting when studying the endogenous congestion effect of AMoD. As AMoD is a new transportation mode, it is possible that additional travel demands may be induced by this new way to travel. Study [1] suggests that autonomous vehicles are likely to add more burden to current transportation network by inducing more travel demand. Because of its convenience and low price, AMoD will attract more people, such as public transport users and cyclists, to call for a ride and this can further deteriorate the traffic condition. To simulate how people will choose their mode of transport, discrete mode choice model [23] and a cost structure based on Charypar-Nagel Utility Function [24] are used in this study. Apart from this, competition between different AMoD operators [25] and vehicle-level interaction among self-driving cars, such as lane reserving technique [26] and intelligent traffic control at intersection [27], will also play a role in the traffic network.

Summary of Literature Review

To conclude the literature review, congestion is a very important aspect to consider when designing the AMoD systems and it is a very complex problem. Whether AMoD will bring more congestion to the road or improve the traffic remains as an open question. In addition, due to the high complexity of the traffic assignment problem in a city scale network, most studies either rely on the simplified and very small traffic network and model (such as [16], [20] and [21]) or use heuristic and empirical methods (such as [1] and [22]) to tackle the problem.

Contribution of this work: This work will bridge the gap between the theoretical research on advanced dispatching algorithms and realistic simulation based studies by developing various online routing and dispatching algorithms for AMoD system. The algorithms are based on the theory from theoretical research and are adapted to the realistic simulation environments. The results of this work may serve as a small effort in mitigating the disagreement on the potential influence of AMoD system among different studies.

Chapter 3

Development of Congestion-Aware Operating Strategies for AMoD Systems

3.1 Objectives of Fleet Operation

A good AMoD operating strategy should provide a satisfying service quality while keeping the operational cost low. Before developing any strategy, we need to setup performance criteria to evaluate whether a strategy is good or not. We will reference to the commonly used evaluation metrics in the literature and develop our performance criteria for the congestion-aware operating strategy for AMoD systems.

Waiting time and on-board traveling time (sometimes also referred to as drive time) are commonly used in the literature to evaluate the service quality of AMoD Systems. For studies on the fundamental of AMoD operation, such as [8], [9] and [13], waiting time is the major criteria to evaluate the service quality of an operating strategy. In these studies, congestion is either ignored or treated as exogenous effect and vehicles will simply choose the shortest path when traveling and on-board time will not be influenced by the operating strategy. On the other hand, when congestion is considered as an endogenous effect of the AMoD vehicles, on-board travel time will also be influenced by the operating strategy. If all the vehicles choose the shortest path based on free flow travel speed of the network, congestion is likely to form on some of the roads and the on-board travel time will increase significantly. In addition to routing of autonomous vehicles, a good operating strategy can also influence the traffic flow by controlling the departure rate of travel demands in different areas. For example, the operating strategy can deliberately delay the service for some travel demand by a short period of time in order to reduce traffic flow during peak hours. Therefore, on-board time is also a very important criterion for the congestion-aware operating strategy.

Then, we will adapt these two commonly used service quality related criteria to our study. One of the advantages of the AMoD over private vehicles is that instead of spending a lot of time in the traffic jam, people can relax at home or in the office until autonomous vehicles arrive at their doorstep. Apparently, spending time in traffic is not as desirable as resting at home or in the office. Indeed, it is common practice in many simulation-based studies, such as [1] and [28], to associate negative utility (i.e., cost) to the time spent on traveling and positive utility to the time spent at a facility (e.g., home, workplace, shop). Besides, with a fleet that is large enough to influence the traffic, the average on-board time provide a good indication to the overall traffic condition. In general, given the same travel demand, a longer average on-board time indicates a worse overall traffic condition. Because of the above mentioned reasons, on-board time will be the major performance criterion for this study. Meanwhile, waiting time should not be ignored, as it is also not desirable to have the passenger wait for a ride for too long. To define the term “too long” is difficult as different people will possess different opinions on this definition and it can vary significantly among different groups of people. However, in general, if the waiting time for a ride is comparable to the time that needed for other available modes of transport, such as the time needed for a private car user to go to the parking lot or garage to reach the car and then drive the car to the road or the time it takes a public transport user to reach a bus or train station and wait for the vehicle to come, then it will not discourage most of the people from using the AMoD service. In this work, we define **5 minutes** as the critical value for mean and median of waiting time and **15 minutes** for 95 percentile waiting time. If the waiting time statistics are below these critical values, then we will define the service quality as satisfying in terms of waiting time. Note that this is a general estimation and the tuning of these values may be needed in real world applications.

Apart from providing a high quality service, reducing operational cost is another objective for the operating strategy. Vehicle Miles Traveled (VMT) is closely related to the operational cost of the AMoD system. The more distance traveled, the more fuel will be consumed, and maintenance cost will also be higher. An operating strategy can reduce VMT by better routing the vehicles and reduce unnecessary re-positioning process of vehicles. Thus, we will also include VMT into our performance criteria. At this stage, however, operational cost is not the major focus and therefore VMT will have a lower priority than the service quality related criteria.

To conclude, in this study, on-board time will be the major criteria for the evaluation of an operating strategy. Waiting time statistics will serve as an additional check to determine if a strategy yield a satisfying performance. VMT will also be included in the comparison process, but it will only be used to demonstrate the potential difference in operational cost associated to each operating strategy.

3.2 Development Environment

In this work, an activity-based multi-agent simulation framework MATSim[29] is used for the traffic simulation. MATSim uses microscopic modeling of traffic and can efficiently simulate a city scale network with a large number of agents in a reasonably amount of time. Thanks to the extension friendly feature of MATSim, extensions with different objectives are readily available. To allow effective implementation of more advanced dispatching algorithm and detailed analysis of the AMoD fleet performance, AMoDues [30] add-on is used in this study. In addition to the improved Autonomous Vehicles Module, AMoDeus also contains several basic unit capacity and high capacity (ride-sharing) dispatching strategy, which serve as an ideal starting point for this study.

As the traffic system is very complex, we need to make some assumptions to reduce the complexity of the problem. The development of the congestion-aware operating strategy in this study relies on the following two assumptions:
Assumption 1. *All private car trips are replaced by the AMoD rides and all the autonomous vehicles running on the road are controlled by the operating strategy.*
Assumption 2. *Only car trips are replaced by AMoD rides and there is only one AMoD operator in the simulated city.*

Although these two assumptions are a little bit strong, they serve as a good starting point to study the endogenous congestion effect of the AMoD operation by excluding many complex elements in the traffic system. Besides, similar assumptions are also commonly used in many relevant studies (such as [14] [16] and [20]).

3.3 Congestion-Aware AMoD Operating Strategy

The Congestion-Aware Operating Strategy (CAOS) mainly consists of two parts: Routing Algorithm (RA) and Vehicle Operating Strategy (VOS). VOS can further be divided into Dispatching algorithm and Rebalancing algorithm. There are interactions between RA and VOS, but we consider them independent in a first approximation. The overall relationship among the elements of the complete operating strategy is shown in figure 3.1.

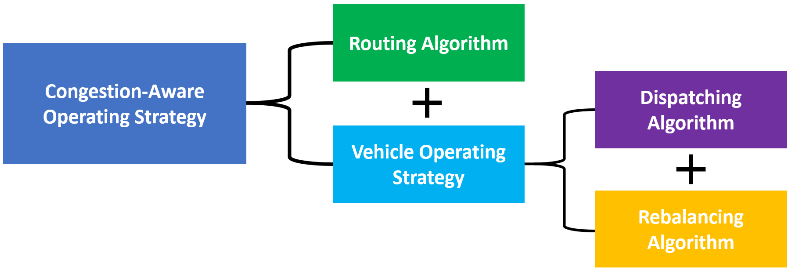

Figure 3.1: Congestion-aware AMoD Operating Strategy

3.3.1 Routing Algorithm

The routing algorithm is an important element in the congestion-aware operation of the AMoD system. With full control over the whole autonomous vehicles fleet, we can assign vehicles in a way such that the congestion in the traffic network is minimized. Due to the large size of the city scales network we are working on in this study, optimal routing plan for a large AMoD fleet is too complex to compute real time (for example, in [16], even with a new technique on modeling the problem and with powerful modern computers, the problem is still not possible to be solved in large scale). Therefore, a heuristic router is a more suitable choice for this study.

There exists an online travel time module called Within-Day Travel Time (WDTT) [31] in the MATSim extension and can be used to construct an online AMoD router. The WDTT module can return the travel time of a given link by averaging the travel time of vehicles that has traversed that link in the last 15 minutes. If no vehicle has passed that link for the last 15 minutes, free flow travel time will be returned. An online routing algorithm can then be constructed by combing a shortest path algorithm (e.g., Dijkstra algorithm) with the WDTT real time travel time module. The cost of a path is set to be the travel time or a linear combination of travel time and distance. This router is working relatively fast and can diverge the traffic flow to some extent. However, limitations present when using this router and that will deteriorate the performance of the AMoD system.

The first limitation of the existing router is lack of updated travel time information on the links. When the router plans the route, it will use the current travel time information on the links and once the route has been planned, the vehicle will follow that route until it reaches destination. As traffic network is a highly dynamic system, congestion can form and relieve within minutes and travel time information of a link can become obsolete in very short time. During a trip, especially those longer trips, the "optimal" route generated by the router at the beginning of the trip (top part of figure 3.2) may no longer be a good choice after some time (bottom part of figure 3.2). The second limitation of the existing router is that it will only diverge the traffic when congestion has already formed, and no extra penalty is given to the congestion. This will keep the whole system stay near the Wardrop's equilibrium (Nash Equilibrium in traffic assignment problem, see [4] for a detailed description) instead of the social equilibrium (a desired, optimal assignment).

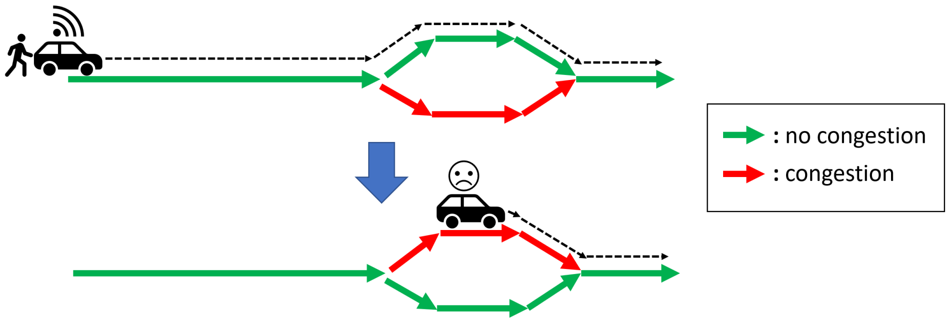

Figure 3.2: Limitation of the online router based on WDTT module

Batch Online Rerouting

In order to overcome the first limitation, Batch Online Rerouting process is introduced. For every time period t_r, a fraction x of the running vehicles (randomly chosen) will re-calculate the route based on the current traffic situation. The variable t_r and x are parameter to be tuned. There is not yet formalized theory or study on the choice of these parameters but in general x should not be too large such that the traffic flow is diverged from the critical links, instead of moving the congestion from one place to another. For example, in the extreme case of x being 1 (i.e., 100 percent), all the vehicles in a region will choose to take the links that are currently less congested and that will bring congestion to those links. In this case, congestion will not be really resolved, but will simply translate back and forth in the network. We demonstrate this with a simple example illustrated in Figure 3.3. On the left of the figure, Route A is congested, and Route B has a good traffic condition. If we reroute all incoming vehicles based on this traffic information, then all vehicles will be assigned to Route B. By doing so, it is likely that congestion will begin to form on Route B and traffic condition on Route A will be improved. Then we will end up in the situation shown on the right of the figure. If we reroute all incoming vehicles again, all vehicles will be assigned to Route A and we will move back to the situation shown on the left of the figure. With that, a loop is formed and congestion is translating in the network instead of being resolved.

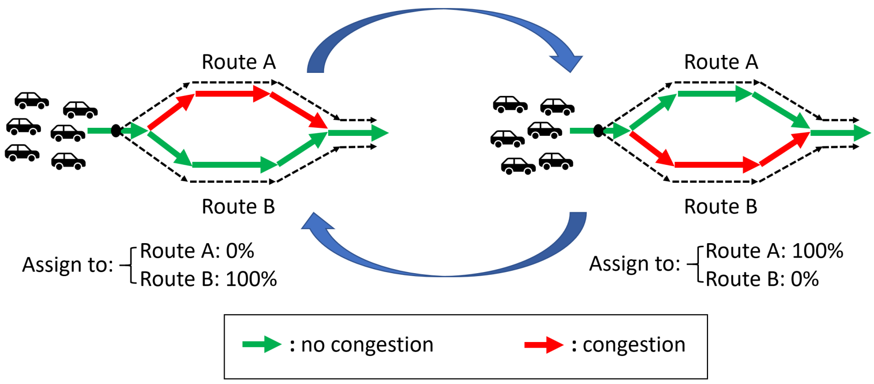

Figure 3.3: Congestion translates back and forth in network

The choice of t_r depends on the value of x as well as the computational power. To reroute some vehicles, shortest path problem will be solved and the more frequently reroute is performed the more computational power it will consume. For a large AMoD system, a high reroute frequency can lead to heavy computational workloads. On the other hand, with a smaller x, a higher reroute frequency (i.e., a smaller t_r) is desirable so that the probability of a vehicle's route is based on updated travel time information is high. In specific, we can calculate the probability of a vehicle whose current route is updated. First, we denote the travel time of a link l at time t to be $\tau_l(t)$. We then define the travel time information to be updated if it is generated at most Δt time before (i.e., $\tau_l(t)$ is updated if and only if $t \geqslant t_{now} - \Delta t$, and t_{now} is the current time), and a route is updated if it is generated by the router based on the updated travel time information. Let M be the event that a vehicle's route is updated then we can calculate the probability of M as shown in equation 3.1. In this study, we set the value of x to be 0.3 (i.e., reroute 30% of the vehicles each time) and t_r to be 30 seconds. Furthermore, we set the Δt to be 3 minutes (180 seconds), which means a travel time information is considered as updated if it is generated within 3 minutes. In this case, we can calculate the probability of a vehicle's route being updated by plugging in these parameters into equation 3.1 and the result is 0.882. This suggests that most of the vehicles are traveling on updated routes.

$$P(M) = 1 - (1 - x)^{(\Delta t / t_r)} \tag{3.1}$$

The benefits of this reroute process are illustrated in Figure 3.4. First, it can relieve the congestion in an area by diverging the traffic flow and not sending additional vehicles to the congested link (see the blue vehicle in the middle in figure 3.4). In this case, not only are the vehicles being rerouted benefited from this process, but also the remaining vehicles in the congested area are benefited, because less vehicles will be presented in that area and the traffic condition can be improved. Second, reroute process can avoid unnecessary detour due to the outdated travel information (see the green vehicle on the right in figure 3.4). Similar to the formation of the congestion, the congestion can be resolved in short time as well. An outdated route may consist unnecessary detour that will only increase the travel time and distance. With the rerouting process, most of the vehicles will be traveling on updated routes and the probability of having such detours will be greatly reduced.

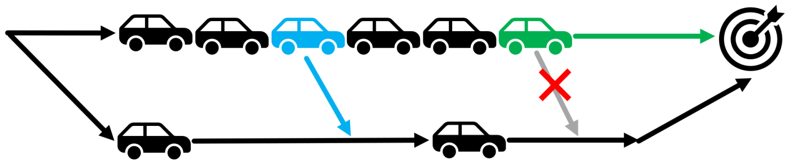

Figure 3.4: Benefits of reroute process

Real Time Travel Time Module

To tackle the second limitation of the existing simple routing algorithm, a new Real Time Travel Time (RTTT) module is constructed. The module penalizes the route that will lead to the congestion by adding cost to links whose occupation rate and flow rate are close to the capacity or are already overloaded. In that way, the router will discourage vehicles to travel on those very crowded links. When the flow rate of a link is well below its capacity, free flow travel time on that link will be returned and no additional cost will be associated to the link. The new real time travel time module estimates the travel time of a link by counting the number of vehicles on the link. Based on the MATSim traffic model, given the flow capacity C_l, the length L_l and free flow travel speed V_l of link l, the maximum number of vehicles $N_l^{freeflow}$ on the l under free flow condition can be calculated by equation 3.2. If there are more vehicles on the link that that value, then free flow condition will no longer be met, and the travel time on the link will increase.

$$N_l^{freeflow} = C_l \cdot \frac{L_l}{V_l} \tag{3.2}$$

The RTTT module is developed based on the literature on traffic assignment problems [32], in which a simple model is used to demonstrate the difference between Wardrop's Equilibrium and Social Equilibrium. Consider a simple network with a direct route through town center and a bypass (see Figure 3.5). Taking direct route under free flow condition will lead to a shorter journey than the detour. But the direct route has smaller capacity limit than the bypass and when the capacity of a road is reached, the travel time will increase as the number of vehicles traveling on that road increase. In a conventional traffic network, a driver will choose to take the bypass when the direct route and bypass have the same cost. Usually, when this point is reached (Wardrop's Equilibrium), the travel time in the direct route is so long, such that the overall benefits (or social benefits) of the network is not the optimum. In order to achieve the social equilibrium where social benefits are maximized, the marginal cost of all routes are equal and minimized. This usually leads to more vehicles taking the bypass rather than direct route. Although the social benefits are maximized, for the drivers who take the bypass, there exist incentive for them to choose the direct route as this will reduce the cost for them. The conventional way to solve this problem is to charge drivers who take the direct route a toll in order to encourage more people to use the detour. In the AMoD case, we have full control over the fleet and artificial cost can be associated to the link that is overcrowded such that the system can move from a Wardrop's Equilibrium towards a Social Equilibrium.

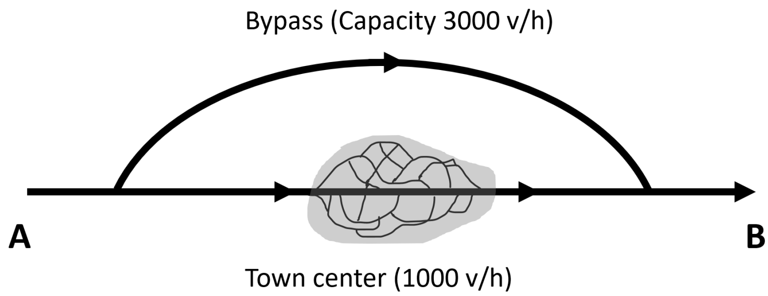

Figure 3.5: Town served by a bypass and a town center route (adapted from [32])

In this study, a cost structure with capacitated network structure with linear penalization on congested link is used by RTTT module for route calculation. The cost structure also distinguishes the passenger carrying vehicles and empty drive vehicles. The cost $C_l^v(t)$ of traveling on a link l for vehicle v at time t is a linear combination of the augmented travel time $\tau_l^v(t)$ (i.e., including penalty for congestion) and the length of the link L_l, and it is shown in equation 3.3. The augmented travel time is calculated by equation 3.4, where τ_l^0 is the free flow travel time of the link l and N_l is the current number of vehicles on the link. $\chi \in \{0, 1\}$ is the Boolean variable representing if the vehicle is carrying passenger with 1 indicating vehicle is driving with customer and 0 meaning the vehicle is in empty drive (see equation 3.5). r_1 and r_2 are the discount rate on the free flow capacity of the link for passenger carrying vehicles and empty vehicle respectively. p_1 and p_2 are the penalty terms that associates more cost for traveling on congested links for passenger carrying vehicles and empty vehicles respectively. $t_l^{addition}$ is the additional travel time for each additional vehicle on the congested link l under the MATSim traffic model and it is simply the inverse of the capacity of the link as shown in equation 3.6. Normally, passenger carrying vehicles have higher priority than the empty vehicles and therefore in general r_1 is larger than or equal to r_2 and p_1 is smaller than or equal to p_2. Due to the complexity of the network and highly dependency of the equilibrium point on the specific network, it is very difficult to calculate the exact additional cost to associate to the congested links in order to reach the social equilibrium, thus parameter tuning need to be performed in each network.

$$C_l^v(t) = f(\tau_l^v(t), L_l) = \alpha \cdot \tau_l^v(t) + (1-\alpha) \cdot L_l \quad with\ \alpha \in [0,1] \tag{3.3}$$

$$\tau_l^v(t) = \tau_l^0 + (N_l - N_l^{freeflow} \cdot (\chi r_1 + (1-\chi) r_2)) \cdot (\chi p_1 + (1-\chi) p_2) \cdot t^{addition} \tag{3.4}$$

$$\chi = \begin{cases} 1, & \text{if } v \in V_{DriveWithPassenger} \\ 0, & \text{otherwise} \end{cases} \tag{3.5}$$

$$t_l^{addition} = \frac{1}{C_l} \tag{3.6}$$

Smoothed Link Travel Time Estimation (SLTTE)

In addition, in the case where AMoD vehicles take up the great majority of all the vehicles running on the roads, the router can be further improved by using Smoothed Link Travel Time Estimation (SLTTE). With SLTTE, a vehicle will not occupy one unit of capacity of the link it is currently on. Instead, occupation of that one unit capacity is distributed among the links the vehicle has just left and the links the vehicle will enter soon (see Figure 3.6). This is possible because we can track the routes of AMoD vehicles. SLTTE will increase the performance of the router by reducing the effect of abrupt changes on the occupation rate of links due to the discrete nature of vehicles.

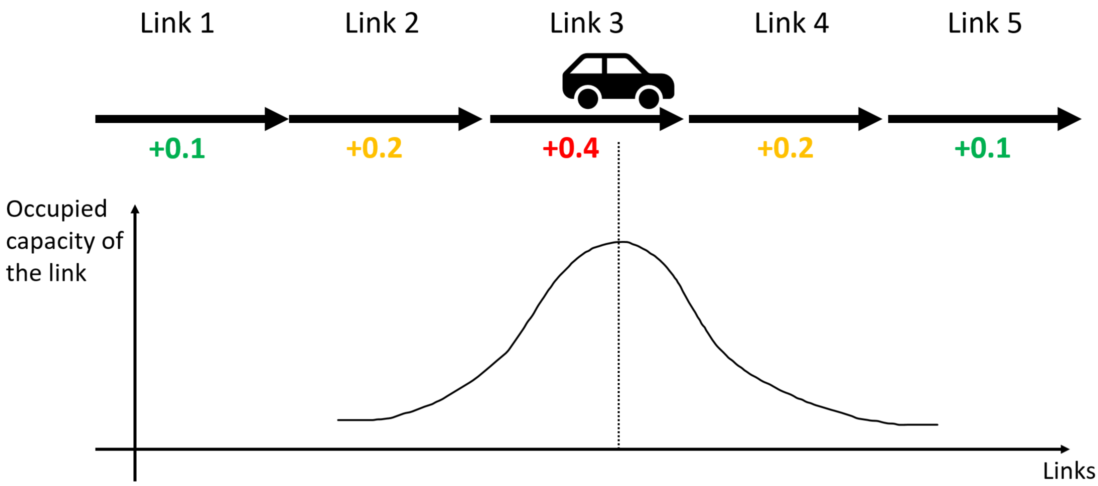

Figure 3.6: Illustration of Smoothed Link Travel Time Estimation (SLTTE)

In this study, we will distribute the occupied capacity of a vehicle across the five links as shown in the Figure 3.6: 2 links the vehicle has just left, the link the vehicle is currently on and 2 links the vehicle will enter thereafter. We will associate weight to each link in the format of $\{w_1; w_2; w_3; w_4; w_5\}$, with $w_i \geqslant 0$. The absolute values of each weight can be randomly chosen, and the distribution is decided by the relative values of the weights. When calculating the occupied capacity of a vehicle on one of the five links, we will normalize the values such that they sum up to one (i.e., a vehicle still occupies one unit of capacity in the network). For example, the weight $\{1; 2; 4; 2; 1\}$ will lead to the distribution shown in the Figure 3.6 and the occupied capacity of a vehicle is distributed as $\{0.1; 0.2; 0.4; 0.2; 0.1\}$ among the five links after the normalization. The weight $\{0; 0; 1; 0; 0\}$ will lead to the occupied capacity of a vehicle concentrating on the link the vehicle is currently on, and this is equivalent to the case in which SLTTE is not applied. For some vehicles, the trajectory may not contain the all of the five links (e.g., at the beginning of a trip or approaching the end of a trip), in that case, only the links covered by the trajectory will be included in the distribution and occupied capacity will be normalized among those included links. As SLTTE is an empirical technique based on MATSim traffic model, pre-simulations are necessary to obtain the most suitable distribution function for SLTTE.

3.3.2 Vehicle Operating Strategy

As shown in Figure 3.1, the Vehicle Operating Strategy (VOS) for the AMoD system can be divided into two parts: Dispatching algorithm, where vehicles are assigned to the travel demands, and Rebalancing algorithm, where vehicles are assigned to drive to some locations without actual requests in order to prepare for future travel demands in that area. Similar to the approach in the existing literature, the two parts of the VOS operate in collaboration but in two separate loops. This enables different combinations of dispatching algorithms and rebalancing algorithm to form new VOSs.

In this section, we will develop three different VOSs. As VOS is part of the Congestion-Aware Operating Strategy, we also take congestion effect into consideration when developing VOSs, which differs our VOSs from most of existing ones in the literature. We will first go through some background information and key ideas for the development of VOS. Then, we will introduce each VOS in detail.

Background Information

In this study, bipartite matching algorithm will be used to form the base of the dispatching and rebalancing algorithm, which provides an efficient match for the whole system. As a classical problem, there are several well formalized methods to solve the bipartite matching problem. The Hungarian method [33] is one of the efficient algorithms to find an optimal bipartite match and will be used to solve the bipartite matching problem. In order to solve the matching problem, we need to define the cost of each potential match. In this study, the cost of a potential match between an available vehicle and a not yet served request is defined as the Euclidean distance between the vehicle and the request. Note that using Euclidean distance as cost function will not lead to an optimal solution in terms of total network distance or travel time as traffic network consists of discrete links and is not a continuous space. Nevertheless, it is the most suitable choice because of following two reasons: first, using network distance or travel time as cost function requires computation of shortest path for every potential match which is extremely time consuming in a large AMoD system; second, using Euclidean distance as cost function can lead to good approximation of the optimal match in most road networks.

When developing rebalancing algorithm, we will divide the network into small regions (also known as virtual nodes). The region should be small enough so that travel demand within each region is relatively uniformly distributed. As the first step, we divide the full network into rectangular regions of the same size. More advanced techniques, such as requests clustering with k-mean algorithm (proposed in article [34]), can be used to divide the network. By doing so, we can implement some of the well-formalized rebalancing algorithms in the literature, which are usually station-based (such as [8] and [9]). Each region in our network can be considered as a station. Sending a rebalancing vehicle from one station to another can then be interpreted as sending one vehicle rebalance from one region to another region. With a small enough region, we can simply pick a random link in the destination region as the target of the rebalance trip. When there are multiple rebalancing vehicles and multiple destinations, bipartite matching can be used to generate the optimal match. In addition, we will slightly abuse the term “rebalance” to include all the empty vehicle operations that do not directly drive toward any actual request and with the purpose of improving the service quality.

Key Idea for Congestion Awareness

The development of a congestion-aware VOS is based on the key idea of limiting departure rate of the requests. One of the major reasons for congestion to form in traffic network is the concentrated traffic flow on some links. When the traffic flow on those links go beyond the capacity of those links, vehicles are going to accumulate on those links due to flow capacity limitation, and congestion will then begin to form and propagate backward. In addition, when congestion is formed, the flow rate of the congested links will further reduce, making congestion to become even worse, and a vicious cycle is formed. One practical way to solve this problem is to limit the departure rate of the travel demand. With a controlled departure rate, overloading of the network is reduced and the overall traffic can be improved. This may lead to an increase in the waiting time for those both spatially and temporally concentrated requests, but the on-board time can be reduced for the whole system.

As we have divided the full network into small regions, we can use them to limit the departure rate. For each region, limited number of requests enter the request pool of the system where they will be matched to available vehicles. Requests beyond the limit are temporarily ignored and will wait for the next dispatching period. When selecting requests in each region to enter the request pool, several methods can be used. In this study, one of the simplest methods, First Come First Serve (FCFS) is used, that means the first N_r requests in each region will enter the system requests pool for matching, where N_r is the departure rate limit for the region.

Vehicle Operating Strategy 1 (VOS1)

To develop the dispatching algorithm for the first Vehicle Operating Algorithm (VOS1), we implement the key ideas of limiting departure rate mentioned above. To pair unserved requests with available vehicles, we will use bipartite matching scheme. As some requests may be deliberately ignored for a short period of time by the dispatching algorithm in exchange for a better overall traffic condition, we name this dispatching algorithm as Tactical Request Ignoring Dispatching Algorithm (TRIDA). The complete process of TRIDA is shown in Algorithm 3.1.

Algorithm 3.1 Tactical Request Ignoring Dispatching Algorithm (TRIDA)

Result: Generate matching between unassigned requests and unassigned vehicles
initialization
$P \leftarrow \varnothing$ ▷ Initialize Request Pool P
$V \leftarrow$ all available unassigned vehicles
for *each region* **do**
    $L_r \leftarrow$ all unmatched requests in region r ▷ L_r is a list structure
    Sort L_r in term of requests submission time in ascending order
    $m = min(N_r, |L_r|)$ ▷ $|L_r|$ is the size of L_r
    $L'_r \leftarrow$ first m requests in L_r
    $P \leftarrow P \bigcup L'_r$
end
perform bipartite matching between V and P

Then, we will develop a rebalancing algorithm for VOS1. Based on the TRIDA above, one obvious way to rebalance the vehicles is to send available vehicles to the area where there are temporarily ignored requests. In the next dispatching period, some of those ignored requests will enter the pool and thus sending vehicles to those area will improve the service quality. Since ignored requests in a region are caused by high departure rate in that region, which will lead to a very busy traffic, it is not desirable to send vehicles directly to the region. Instead, rebalancing vehicles are sent to neighboring areas of the busy region. In order to quantify this, we introduce the limit of incoming traffic which upper bound the number of vehicles that is assigned to travel towards any region in the network. As we can treat rebalance trip as pickup drive to a virtual request, it is reasonably to set this limit same as departure rate limit N_r. With this regulation term, those busy regions with many incoming pickup vehicles will not accept any rebalancing in vehicles which will prevent them from being overloaded. Available vehicles will be sent to neighbors of those busy regions and it will take those vehicles less time to reach the customer in the next dispatching period. To generate the rebalance plan, we randomly choose corresponding number of links in the destination regions as the targets for rebalancing trips. Then, we use bipartite matching scheme to send the unassigned available vehicles to those targets of rebalancing trips. As this rebalancing algorithm is responding to the artificial "imbalance" created by the TRIDA, we refer to it as Pseudo Rebalancing Algorithm (PRA). Algorithm 3.2 shows the full process of PRA.

Algorithm 3.2 Pseudo Rebalancing Algorithm (PRA)

Result: Generate rebalancing plan
initialization
$L \leftarrow \varnothing$ ▷ Initialize rebalance destinations set
$S \leftarrow$ all regions with ignored requests
$V \leftarrow$ all available unassigned vehicles
for *each region* r_i *in* S **do**
    $N \leftarrow$ all neighboring region of r_i
    $A \leftarrow \sum_{r_j \in N} a_j$ ▷ a_j is the current available incoming vehicles quota of region r_j
    $x \leftarrow$ number of ignored requests in the region r_i
    while $x > 0$ *AND* $A > 0$ **do**
        pick a random region $r_k \in N$
        if $a_k > 0$ **then**
            pick a random link l in region k
            $L \leftarrow L \bigcup \{l\}$
            $a_k \leftarrow a_k - 1$
            $x \leftarrow x - 1$
            $A \leftarrow A - 1$
        else
            remove region k from N
        end
    end
end
perform bipartite matching between V and L

Combing the TRIDA and PRA, we get the complete Vehicle Operating Strategy. During the operation process, dispatching algorithm and rebalancing algorithm will be called at same frequency. TRIDA will be called first to generate match between available vehicles and requests and based on the matching results, PRA will then be executed immediately after to produce rebalancing plans.

Vehicle Operating Strategy 2 (VOS2)

The second Vehicle Operating Strategy (VOS2) uses the same dispatching algorithm, TRIDA, as VOS1. But, instead of PRA, a more active rebalancing algorithm will be used to work with TRIDA. We develop the new rebalancing algorithm based on the Adaptive Real-time Rebalancing Algorithm (ARRA), originally proposed in article [9].

The original ARRA use the real time vehicle and travel demand information and spread the available vehicles evenly across all regions (interpreted from the term "stations" in [9]) in the network. During each rebalancing period, the number of available unassigned vehicles (staying vehicles) will be counted and the desired number of vehicles in each region $N_{desired}$ is calculated by dividing that value by the number of active regions (i.e., regions inside the AMoD service coverage). The number of vehicles possessed by a region is the sum of the staying vehicles inside a region, rebalancing in vehicles and vehicles arriving with passengers. If the number of possessed vehicles by a region is greater than the desired value, then the region is referred to as a surplus region; If the number of possessed vehicle of a region is smaller than desired value, then it is labeled as a deficit region. The rebalancing algorithm will then send staying vehicles in the surplus regions into deficit regions.

In order to adapt the original ARRA to this study, some modifications are made. First, a buffer β is added to the categorization of regions. A region is considered as deficit or surplus only if the difference between the number of possessed vehicles by the region and the desired value $N_{desired}$ is greater than a threshold. Otherwise, the region is treated as “balanced” and no vehicles will be rebalanced to or from this region. In the original work of ARRA flow model is used, and the travel demand are considered as a piece-wise constant flow [9]. In our simulation environment, dynamic discrete travel demand is used and thus a region can fluctuate between deficit and surplus due to the newly entering requests and this will lead to unnecessary back and forth rebalance assignment. Obviously, the back and forth movement of empty vehicles will significantly intensify the congestion and should be avoided. By introducing the buffer, such problem can be solved and only the region with non-trivial imbalance will be balanced by the algorithm. Second, when generating a rebalance plan, a limit of incoming and outgoing flow of vehicles N_{max} for all the regions are introduced. This is similar to the idea in the dispatching algorithm and the first rebalancing algorithm above, and it will prevent the network from overloading. Third, we modified the rebalancing plan generating process. As mentioned in the Background information above, we use small regions in our network to represent the station concept in the original ARRA. Instead of using linear programming to find the optimal rebalancing flow, we use bipartite matching scheme to generate rebalancing plans.

With above mentioned modifications, the new rebalancing algorithm based on ARRA, General Rebalancing Algorithm (GRA), is formulated in Algorithm 3.3.

By combing TRIDA with GRA, we get the complete process of the second Vehicle Operating Strategy (VOS2). Same as in VOS1, the dispatching algorithm and rebalancing algorithm are called at the same frequency. TRIDA will match unserved requests with available vehicles and then GRA will produce rebalancing plans based on the matching results.

Algorithm 3.3 General Rebalancing Algorithm (GRA), based on ARRA [9]

Result: Generate rebalancing plan

initialization
$L \leftarrow \varnothing$, $V \leftarrow \varnothing$ ▷ Initialize rebalance destinations and rebalance vehicles set
$N_d \leftarrow \varnothing$, $N_s \leftarrow \varnothing$ ▷ Initialize deficit and surplus regions sets N_d and N_s
$V^s \leftarrow$ all staying vehicles
$M \leftarrow$ all active regions
$N_{desired} \leftarrow \frac{|V^s|}{|M|}$ ▷ $|\cdot|$ is the number of elements in the set
for *each region* r_i *in* S **do**
  $D_i \leftarrow$ unmatched travel demands originating from region r_i
  $X_{ji} \leftarrow$ arriving with customer vehicles to r_i coming from r_j
  $Y_{ji} \leftarrow$ rebalancing vehicles to r_i coming from r_j
  $n_i = |V_i| + \sum_{j \neq i} (|X_{ji}| + |Y_{ji}|) - |D_i|$ ▷ calculate the number of vehicles possessed by r_i
  if $n_i \leqslant (1-\beta)N_{desired}$ **then**
    $d = N_{desired} - n_i$ ▷ deficit value
    $l = min(d, N_{max})$
    $L_i \leftarrow$ randomly generate l destination links in region r_i
    $L \leftarrow L \bigcup L_i$
  end
  if $n_i \geqslant (1+\beta)N_{desired}$ **then**
    $s = n_i - N_{desired}$ ▷ surplus value
    $v = min(min(d, N_{max}), V_i^s)$ ▷ V_i^s is the set of staying vehicles in region r_i
    $V_i \leftarrow$ randomly choose v staying vehicles in region r_i
    $V \leftarrow V \bigcup V_i$
  end
end
perform bipartite matching between V and L

Vehicle Operating Strategy 3 (VOS3)

The third Vehicle Operating Strategy (VOS3) is a more advanced VOS that includes the concept of departure rate limit, pre-calculated feed-forward signals and a real time feedback loop. The full process of VOS 3 is illustrated in Figure 3.7. With this advanced VOS, we can increase the efficiency of the AMoD fleet and same travel demands can be served by a smaller fleet without compromising the service quality.

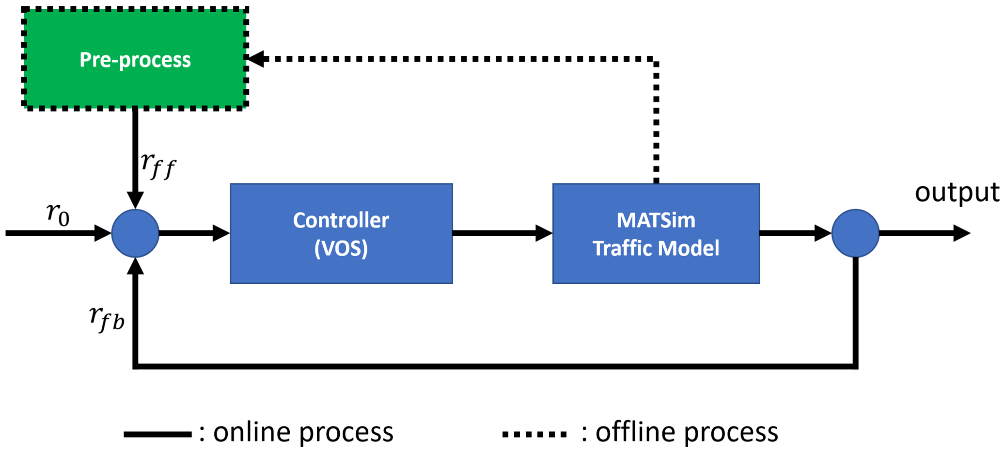

Figure 3.7: Illustration for VOS3

The dispatching algorithm of VOS3 is similar to TRIDA but matching between unserved requests and available vehicles only happens within each region, instead of across the network. This is because the rebalancing algorithm of VOS3, which will be introduced below, relies more on the concept of region than previous rebalancing algorithms. Because of this region-based matching characteristic, the dispatching algorithm is named as Region-based Assignment Dispatching Algorithm (RADA). Algorithm 3.4 shows the process of the RADA. Same as in TRIDA, N_r is the departure rate limit for each region.

Algorithm 3.4 Region-based Assignment Dispatching Algorithm (RADA)

Result: Generate matching between unassigned requests and vehicles

initialization

for *each region* r_i **do**

  $R \leftarrow$ all unmatched requests in region r_i

  $V \leftarrow$ all staying vehicles in region r_i

  Sort R in terms of requests submission time in ascending order

  $n = min(|R|, N_r)$

  $P \leftarrow$ first n elements in R

  Perform bipartite matching between P and V

end

The Advanced Rebalancing Algorithm (ARA) of the VOS3 consists of two parts: feed-forward part and feedback part. While feed-forward signal will be calculated offline, the feedback process will be carried out online during the operation.

The feed-forward part of the VOS3 is based on the idea of an advanced rebalancing algorithm, Feed-forward Fluidic Optimal Rebalancing Algorithm (FFORA), in the literature [9]. The original FFORA is developed in a station-based network with constant travel demand flow. It uses travel demand flow among stations to calculate an optimal rebalancing flow (can be think as the counter flow) such that the total empty drive distance is the least. The algorithm calculates the optimal rebalancing flow offline based on the available travel demand information (e.g., past data) and generate a master rebalancing plan before the online operation. During the online operation, the rebalancing algorithm will read from the pre-calculated master plan and execute corresponding rebalance assignments.

To implement the idea from FFORA in our study, we again use the region in our network to represent the station in original FORS. We divide a day into 15-minutes intervals and within each interval the travel demand flow can be considered as constant. We then use available travel demand flow data to calculate the optimal rebalancing flow among regions for each time interval and store it in a file. During the online operation, the file will be read by the VOS to execute feed-forward part of the rebalancing assignments. To assign a vehicle to rebalance from one region to another, we will randomly choose an available vehicle in the rebalancing out region and send it to a randomly selected link in the destination region.

The feed-forward part of the Advanced Rebalancing Algorithm (ARA) is shown in Algorithm 3.5. $\alpha_{ij}(t)$ is the theoretical optimal number of to rebalance from region i to region j at time t. To acquire $\alpha_{ij}(t)$, we can read the optimal rebalancing rate from region i to region j at time t from the pre-calculated file and then multiply that rebalancing rate with the time period between two consecutive feed-forward part of the ARA is called.

Algorithm 3.5 Advanced Rebalancing Algorithm (ARA), Feed-forward part (based on [9])

Result: Generate feed-forward rebalancing assignment
initialization
for *each region* r_i **do**
  $V_r \leftarrow$ staying vehicles in region r_i
  Calculate $\alpha_{ij}(t)$ from pre-calculated rebalancing rate data for $\forall j \in \{1, 2, ..., N_R\}$
  for *each region* r_j **do**
    if $\alpha_{ij}(t) > 0$ *AND* $|V_r| > 0$ **then**
      $n \leftarrow min(floor(\alpha_{ij}(t)), |V_r|)$
      Send n vehicles from r_i to n random links in r_j
    end
  end
end

Finally, we introduce the feedback term in the ARA. Because the feed-forward rebalancing plan is calculated beforehand, based on projected travel demand, it is very likely that there will be some differences between the predicted travel demand and the real one. Introducing a feedback term can eliminate the error caused by the difference between projected demands and real demands. Similar idea will be used as in the general rebalance strategy introduced in section 3.3.2. We calculate the deficit or surplus of each region by taking the difference of the number of vehicles possessed by the region and the current unmatched requests originating in the region. Unlike the general rebalancing strategy, we do not simply spread the surplus vehicles evenly across the network, we send more vehicles to the popular departure regions so that the possibility of those region running out of vehicles will be reduced. For the rest regions, a bare minimum number of surplus vehicles will be the reference signal to be tracked by the rebalancing algorithm. The feedback part of the ARA is shown in Algorithm 3.6.

Algorithm 3.6 Advanced Rebalancing Algorithm (ARA), Feedback part

Result: Generate feedback rebalancing plan

initialization
$V \leftarrow$ all staying vehicles
$R^h \leftarrow$ high demand regions; $R^n \leftarrow$ the rest regions
$L \leftarrow \Phi$, $V \leftarrow \Phi$
for *each region* r_i **do**
    $D_i \leftarrow$ unmatched travel demands originating from region r_i
    $X_{ji} \leftarrow$ arriving with customer vehicles to r_i coming from r_j
    $Y_{ji} \leftarrow$ rebalancing vehicles to r_i coming from r_j
    if $r_i \in R^h$ **then**
        $N_i^d = N^h$
    else
        $N_i^d = N^n$
    end
    $n_i = |V_i| + \sum\limits_{j \neq i} (|X_{ji}| + |Y_{ji}|) - |D_i|$
    if $n_i \leqslant (1-\beta)N_i^d$ **then**
        $d = N_{desired} - n_i$
        $l = min(d, N_{max})$
        $L_i \leftarrow$ randomly generate l destination links in region r_i
        $L \leftarrow L \bigcup L_i$
    end
    if $n_i \geqslant (1+\beta)N_i^d$ **then**
        $s = n_i - N_{desired}$
        $v = min(min(d, N_{max}), V_i^s)$
        $V_i \leftarrow$ randomly choose v staying vehicles in region r_i
        $V \leftarrow V \bigcup V_i$
    end
end
Perform bipartite matching between V and L

Combining the RADA with ARA, the third Vehicle Operating Strategy (VOS3) is complete. As the dispatching algorithm in VOS3 generate matching only between vehicles and requests within each region, the workload is lower than the bipartite matching across the whole network. Therefore, RADA can be called at a higher frequency than the dispatching algorithm in the previous two VOS. Similarly, the feed-forward part of the ARA only consists of reading data from the pre-calculated file and simple calculation and the computational workload is also relatively light compared to the feedback part of the ARA. As a result, feed-forward part of the ARA can be called more frequently than the feedback part.

Summary of VOS

In this section, we have introduced many algorithms for different Vehicle Operation Strategies. To make it easier for readers to distinguish three VOSs introduced above, we summarize them in Table 3.1.

As mentioned in the beginning of this chapter, the complete Congestion-Ware Operating Strategy (CAOS) for AMoD system consists of Routing Algorithm (RA) and Vehicle Operating Strategy (VOS). Each VOS can be pair up with the newly developed RA in section 3.3.1 to form a complete CAOS (i.e., CAOS1=RA+VOS1; CAOS2=RA+VOS2; CAOS3=RA+VOS3).

VOS	**Dispatching Algorithm**	**Rebalancing Algorithm**
VOS1	TRIDA (Algorithm 3.1)	PRA (Algorithm 3.2)
VOS2	TRIDA (Algorithm 3.1)	GRA (Algorithm 3.3)
VOS2	RADA (Algorithm 3.4)	ARA (Algorithm 3.5 + 3.6)

Table 3.1: Summary of three Vehicle Operation Strategies

3.4 Simulations and Results

3.4.1 Simulation Setup

Simulation Scenario: Sioux Falls

Sioux Falls is one of the suitable scenarios for the development and testing of AMoD operating strategies. Sioux Falls is a city in the South Dakota, US (see Figure 3.8) and is a commonly used scenario for the traffic study. According to [35], this scenario combines fully dynamic demand fitted with realistic socio-economic and demographic attributes with a small scale network. Although the Sioux Falls Scenario is not created with the goal to exactly replicate the real world of the Sioux Falls city, it is a very suitable test bed. For more information on the Sioux Falls scenario and the data used to create the original scenario, readers may refer to [35].

Some small modifications are needed to adapt the Sioux Falls scenario to this study. The original network of the Sioux Falls scenario is rather coarse and does not include the roads in the surrounding areas of the city center. This limit the route choice option for the AMoD system. Without adequate routing options, the potential of AMoD operating strategies cannot be fully realized. Besides, a coarse network also undermines the door-to-door concept of the AMoD system. With the smaller links inside the residential and business area missing, commuters can only be picked up or dropped off on major roads instead of at their doorsteps. As a result, to make the Sioux Falls scenario more suitable for this AMoD study, we regenerate a more detailed network that also includes roads in the immediate surroundings of the Sioux Fall city based on the data available on Open Street Map [36]. The newly generated network is shown in Figure 3.9. The travel demands are then adapted to the new network.

In this work, the public transportation is not analyzed. All public transport trips are converted to walk and will be excluded from the statistics. This will not have any significant effect on the traffic condition as public transport trips takes up only a very small percentage of the total travel demand in Sioux Falls scenario and the small scale public transport network in the scenario has very limited effect on the congestion. Besides, agents who have activities outside the scenario boundary are also removed. After the scenario preparation, there are in total around 60,000 agents whose activities throughout the day are within the boundary of the scenario. An agent in the simulation may make several trips during the day, such as going to work, going shopping and returning home and that leads to a total of 120,320 trips throughout the day. Around 90% of the all the trips are car trips and those trips are the suitable ones to be replaced by AMoD rides. The rest 10% are walking (including those converted from public transport trips) or bike trips. Those trips do not influence the traffic in MATSim simulations and will be excluded from the statistics in this study.

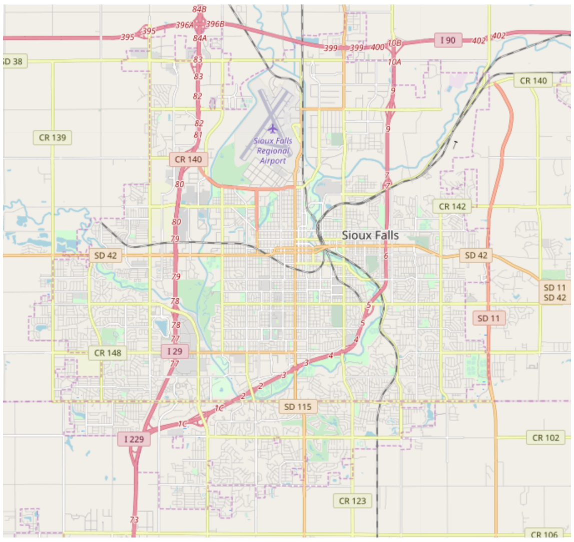

Figure 3.8: Map of the Sioux Falls on Open Street Map

Figure 3.9: Prepared network for the Sioux Falls Scenario

As a common practice in MATSim simulations, a sub-population is used to speed up the process. In our simulations, a sub-population of 20,000 randomly chosen from the full population is used. This corresponds to around 30% of the full population and thus the network capacity (flow capacity and storage capacity) is shrunk down to 30% of the original value. Among those 20,000 chosen agents, around 18,000 owns a car and there are in total 36,148 car trips throughout the day. Based on Assumption 1 and Assumption 2 in section 3.2, all those private car trips will be converted to AMoD rides and will be served by our AMoD fleet.

To decide the initial AMoD fleet size to serve those travel demands, we refer to the findings in the literature. According to various studies, such as [37] and [38], the fleet size required for AMoD to serve all the personal mobility demand in a city can be somewhere between 10 percent and one third of the number of existing private cars in the city. As a starting point, we use adequate number of vehicles to test our newly developed CAOS and later in Chapter 4 we will perform stress test by gradually reducing the fleet size. We choose 6,000 as the initial fleet size to serve the all travel demands converted from private car trips, which corresponds to one third of the 18,000 existing private cars in Sioux Falls Scenario.

CAOS setup

Parameters used for the Routing Algorithm are listed in the table 3.2 and values of those parameters chosen tuned based on the results of pre-simulations. Vehicle Operating Strategy related parameters are listed in table 3.3. The dispatching and rebalancing frequency for VOS1 and VOS2 are set to be 30 seconds, which are commonly used in the literature. For VOS3, as mentioned above, a higher frequency will be used for dispatching algorithm and feed-forward part of the rebalancing algorithm. The other parameters depend on the scenario and the network. Values for those parameters are acquired by running pre-simulations.

Parameter Name	**Value**
Reroute part	
reroute proportion x	0.3
reroute frequency m	30 seconds
Link travel cost part	
time-distance α	1.0
discount ratio r_1	0.9
discount ratio r_2	0.9
penalty term p_1	2
penalty term p_2	4
SLTTE	
occupation distribution	{0; 3; 6; 3; 2}

Table 3.2: Routing Algorithm parameters

Parameter Name	**VOS1**	**VOS2**	**VOS3**
Dispatching algorithm			
dispatching frequency [second]	30	30	10
departure rate limit per region N_r	20	20	6
Rebalancing algorithm			
rebalancing frequency [second]	30	30	FF:10; FB:30
buffer β	N/A	0.1	N/A
rebalance rate limit N_{max}	N/A	20	20
normal region reference N^n	N/A	N/A	1
popular region reference N^h	N/A	N/A	10

Table 3.3: Vehicle Operating Strategy parameters

3.4.2 Benchmarks

In order to evaluate the performance of the new algorithm, the results are compared to benchmarks. Two benchmarks are used in this study to examine the performance of the new algorithm. The first benchmark is derived from the Wardrop's Equilibrium of all the private car trips (i.e., no AMoD replacement) and the second benchmark is the performance of an existing operating strategy in the literature.

Benchmark 1

To acquire the first benchmark performance, the conventional MATSim iterative approach is applied. The detailed introduction to this iterative approach can be found at the MATSim documentation [29]. The same sub-population is used under the same scenario setup as described in section 3.4.1. The only difference is that all private car trips remain unchanged. As the population travel plan is synthesized to represent real world data, the mode choice and departure time are fixed during iterations and agents are only allowed to change the route choice during the re-planning process between two iterations. The strategy of re-planning is set as follow: 0.75 change exponential beta and 0.25 reroute (note: this is from the MATSim re-planning process and is different from the reroute term in our Routing algorithm). With adequate iterations, the scenario will reach the Wardrop's Equilibrium (also referred to as Dynamic User Equilibrium in MATSim). We use 50 iterations with innovation process (i.e., reroute) turned off after the 45th iteration. The plots of iteration scoring and travel distance indicate that user equilibrium has been reached well before the last iteration. From this equilibrium we extract the average travel time and total travel distance of all the 36,148 car trips. In the private car case, the travel time equals to the on-board time (drive time) as the time needed to get to the car and drive the car to the network are not included. Also, for private car users, the term waiting time is trivial as users are always considered to have access to their own cars.

Benchmark 2

The second benchmark is derived from an existing Operating Strategy in [22], where congestion effect of the AMoD fleet is studied. The Operating Strategy consists of the Demand Supply Balancing Dispatching Algorithm [39] and an online router based on Within-Day Travel Time module [31] in MATSim. The structure of this existing Operating Strategy is a little bit different from the ones we have constructed above. As there is no rebalancing algorithm, dispatching algorithm is the only element in the Vehicle Operating Strategy.

The Demand Supply Balancing Dispatching Algorithm (DSBDA) is a heuristic dispatching algorithm that can perform matching between travel demands and available vehicles in very large scale. DSBDA divides the AMoD system into two situations: oversupply and under supply. In the oversupply case, requests are assigned to the closest available vehicle. In under supply case, where all vehicles are busy, the algorithm will wait until one (or more) vehicle becomes free and that vehicle will be assigned to pick up the request that is closest to it. Because of its simplicity, DSBDA can effectively operate in a city scale network (such as Berlin) and yield satisfying matching. The online router based on Within-Day Travel Time (WDTT) module has been introduced at the beginning of the section 3.3.1. By combing the online router with the DSBDA, a complete Operating Strategy is formed.

3.4.3 Results

The results of the newly developed Congestion-Aware Operation Strategies for AMoD system are shown in figure 3.10. The benchmarks are also included in the same figure for better comparison.

In addition to the numerical results, simulation results are also visualized to provide a more intuitive comparison between the results. Visualization process is carried out on the Via application from Simunto [40]. One snapshot of the visualization is showed in figure 3.11. On the left is the AMoD case with CAOS1; on the right is the Benchmark 1 (private cars case). Each triangle represents a single vehicle (either conventional private car or self-driving car) and the color of the car represents the relative speed to the free flow speed of the link the vehicle is currently on. A triangle in green represents the vehicle is traveling at free flow speed of the link while red indicates that the vehicle is traveling at a very low speed (i.e., congestion is happening). The colors in between (e.g., lime, yellow and orange) represent the situations between free flow and congestion. The snapshot is taken during the evening rush hours, when the travel demands in the Sioux falls reach the peak value of the day. It can be observed that the traffic condition is better in the AMoD case (on the left) than the private car case (on the right) even though there are actually more vehicles running on the road in the AMoD case. The full video of the visualization process is available online (see appendix).

	Mean Drive time [s]	Waiting Time [s]			Mean Total Journey Time* [s]	VMT [km]
		Mean	50%	95%		
Benchmark 1 (Private vehicles)	328	N/A			328 + X	117,666
Benchmark 2 (Existing Operation Strategy [22])	373	230	74	907	603	130,587
CAOS1 (RA+VOS1)	264	160	67	592	424	127,829
CAOS2 (RA+VOS2)	271	126	56	425	387	168,200
CAOS3 (RA+VOS3)	267	135	46	518	402	151,535
Lower Bound (Infinite Network Capacity)	198	Values depend on VOS				

*: X represents the time required for private car users to reach the car and to drive the car from parking place to the road.

Figure 3.10: Results

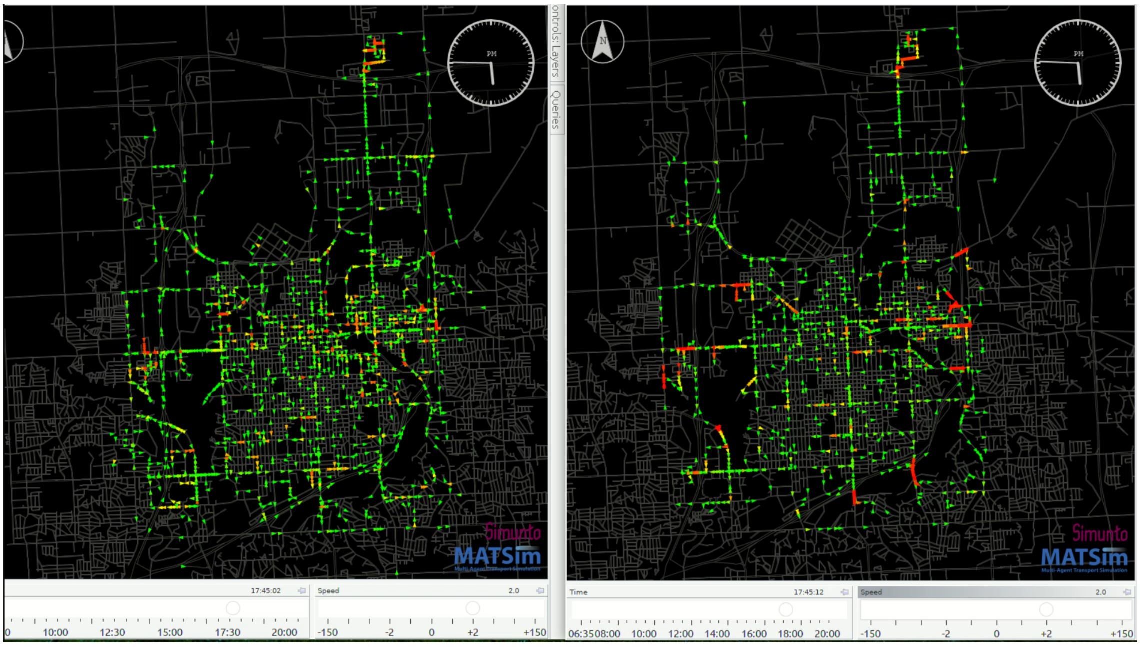

Figure 3.11: A snapshot of visualization process

3.5 Analysis

The first interesting observation is the comparison between the two benchmarks: the private car case and the AMoD case with existing operating strategy. The drive time (i.e., on-board traveling time) of Benchmark 2 is longer than that of Benchmark 1. This means if AMoD system operated by the strategy in Benchmark 2 (i.e., existing Operating Strategy) is introduced to replace all private cars trips in the city, it will end up with a making the traffic condition even worse. This confirms the conclusion in article [22] that introducing AMoD will lead to more congestion if network capacity remains the same. But this is not the end of the story, because the original DSBDA is not designed for a congested network and the existing online router has obvious limitations (as discussed in section 3.3.1 above), and as a result, it is not too surprising that the performance of the operating strategy is not very impressive in a congested network.

With the implementation of our newly developed CAOSs, which consists of both improved routing algorithm and vehicle operation strategies developed for congestion tackling purpose, the situation is different. For both congestion-aware operation strategies, the drive time are reduced by considerable amount compared to both benchmarks. This suggests that passengers spend less time traveling on the road and the overall traffic condition is improved when congestion-aware AMoD system is introduced to the city. This can also be confirmed by the visualization of the simulations (see figure 3.11). In addition, the visualization process also shows that using the congestion-aware AMoD system, autonomous vehicles will spread across the network and detours are actively taken when congestion begins to form in some regions. Such active decision maintains a good traffic flow throughout the network even during peak hours. With the improved online routing algorithm and the vehicle operating strategy working collectively, the negative effect on traffic due to the additional vehicles running on the road and additional mileage traveled can be overweighed by the positive effect of the better coordination of the running vehicles and better controlled travel demands. This implies that AMoD will not increase the congestion, and can even improve the traffic, under the same network capacity, as long as a suitable strategy is used to operate the system. This is indeed an optimistic outlook for the AMoD system.

Apart from the drive time and the congestion, the waiting time and traveling distance should also be taken into consideration when analyzing the AMoD performance. From the results, it can be observed that the waiting time statistics (mean, median and 95 percentile) of all three newly developed CAOSs meet the criteria in section 3.1. This suggests that changing from driving private cars to riding AMoD will not lead to any significant change in daily plans of the people and the service quality is satisfying in term of waiting time.

The total Vehicle Miles Traveled (VMT), increases to some extent when AMoD is used to replace private car journeys. When a more active rebalancing strategy is used, the VMT increases even more. Unfortunately, the increase in the VMT is inevitable for the AMoD system because autonomous vehicles have to cover the distance between two travel demand they serve consecutively. Although we can avoid causing additional congestion with the extra VMT by implementing good operating strategy, it may still lead to some concern on the energy consumption and the potential fare increase. With improvements in vehicle propulsion technology (e.g., better electric motors, energy recuperation system) and self-driving algorithm, autonomous vehicles can become more energy efficient and this concern can be greatly relieved. The details on vehicle propulsion technology and self-driving algorithm is beyond the scope of this study and therefore will not be included.

Chapter 4

Testing and Extending the Congestion-Aware Operating Strategies

Simulations in Chapter 3 show that replacing all the private car trips in Sioux Falls scenario with congestion-aware AMoD system can improve the overall traffic condition. The results demonstrate a positive outlook for AMoD systems. In this Chapter, we will carry out some tests to examine the performance of our newly developed CAOSs under stress and whether the good results in Chapter 3 can be generalized to different scenarios. We will also extend our CAOS to a more realistic setting, the mixed traffic case, where only some of the private car trips are replaced by AMoD rides and both autonomous vehicles and private cars present in the traffic network. The three directions we will work on in this Chapter are summarized in the Figure 4.1.

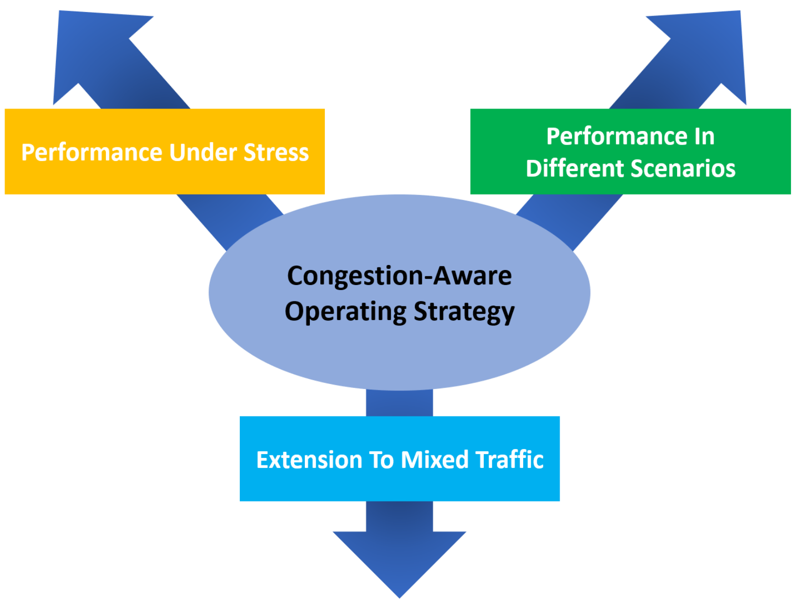

Figure 4.1: Testing and Extending the CAOS

4.1 Fleet Size Stress Test

The term stress test is a common procedure in many fields, such as in material science, finance and computer engineering, where a system is exposed to heavy load to examine whether the system can still work properly or fail to function. In this section, we will perform stress test on the newly developed CAOSs for AMoD systems. By doing so, we can examine how good the newly developed CAOSs respond to heavy load and explore the possibility of serving the same travel demands with a smaller fleet without compromising the service quality.

To increase the load on AMoD system, we can increase the travel demands or reduce the fleet size (or do both). In this study, we will perform stress test on our CAOSs by reducing the fleet size while maintaining the same travel demands. By keeping the travel demands unchanged, we do not need to adjust the capacity of the network, as our network is scaled down based on the size of the sub-population. In addition, increasing travel demand will lead to more computational workload and longer simulation time. As a result, reducing the fleet size while keeping travel demands unchanged is a preferable way to impose more load on the AMoD system.

We use the Sioux Falls scenario to perform stress test on the three CAOSs. The scenario settings and operating strategies settings remain the same as in Chapter 3 and the only difference is the fleet size. We gradually reduce the fleet size from 6,000 to 2,000 with a step size of 1,000. By reducing the fleet size, the waiting time is expected to grow and if waiting time does not meet the criteria, we have developed in Section 3.1, then we will define the service quality as unsatisfying. Based on simulation results, when fleet size is reduced to 2,000, none of the CAOS can provide satisfying service quality in terms of waiting time. Furthermore, obvious shortage of vehicles can be observed. Therefore, it does not make sense to further reduce the fleet size and we stop at fleet size of 2,000 for the stress test. The results of stress test are shown in Figure 4.2 to 4.6.

From the results, it can be seen that three CAOSs react differently to the stress test. There is a minor decrease in the mean drive time as the we reduce the fleet size. The waiting time related statistics, on the other hand, experience some significant changes when different loads are imposed on the AMoD system. This is expected as it is obvious that the less vehicles there are, the longer the AMoD users have to wait for a ride. The total VMT for CAOS1 and CAOS3 increase slightly as the fleet size reduces while VMT for CAOS2 behaves a little bit differently.

The resilience of a CAOS against stress can be revealed by the waiting time statistics. With an adequate fleet size, the three CAOSs all provide a satisfying service quality in terms of waiting time (mean, median and 95 percentile). When we gradually reduce the fleet size, it can be observed that CAOS3 has a relatively higher resilience to stress compared to the other two CAOSs. At the fleet size of 3,000, the mean waiting time and 95 percentile waiting time of the CAOS3 is clearly better than the other two CAOSs. This indicates that CAOS3 is capable of providing a more satisfying service quality when serving the same travel demands with a smaller fleet size. Indeed, as shown in Figure 4.3, 4.4 and 4.5, all three waiting time statistics of CAOS3 are well below the criteria we have developed in Section 3.1, which is marked by the red horizontal dotted line in the figures. This suggests that with a more advanced rebalancing algorithm, the AMoD system can be less sensitive to the stress and the same travel demands can be served by a smaller fleet size without compromising the service quality.

On the other hand, with a smaller fleet size, the VMT become larger for CAOS1 and CAOS3. This is not surprising because with a smaller fleet size, vehicles need to cover more extra distance between two trips due to the imbalance of the travel demands. This can be viewed as a tradeoff between fleet size and service quality. While a smaller fleet size reduces the upfront investment for the AMoD operator, the induced longer VMT will increase the operational cost. In addition, throughout the stress test, the VMT of CAOS3 is always larger than CAOS1. This suggests that AMoD operator also need to tradeoff between the good service quality and longer VMT brought by a more advanced rebalancing algorithm at the same time.

For CAOS2, the different behavior in the VMT is because of the structure of the operating strategy. With adequate vehicles, there will be more rebalance trips to spread all the available vehicles across the network. When fleet size is reduced, the rebalance process will be reduced significantly as there are less available vehicles to spread across the network. Meanwhile, vehicles need to cover more distance between two trips. These two factors are having opposite effect on the VMT and therefore CAOS2 generates a more complex VMT plot in the stress test.

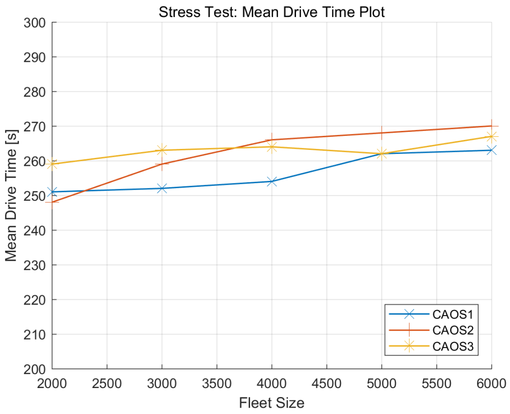

Figure 4.2: Stress Test: Mean Drive Time

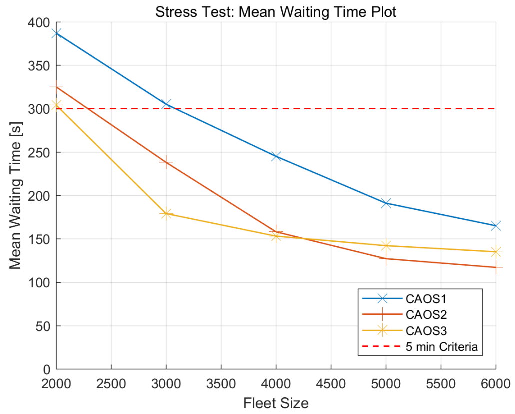

Figure 4.3: Stress Test: Mean Waiting Time

Figure 4.4: Stress Test: Median Waiting Time

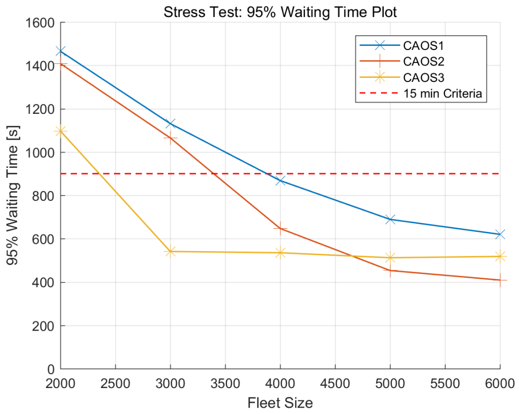

Figure 4.5: Stress Test: 95% Waiting Time

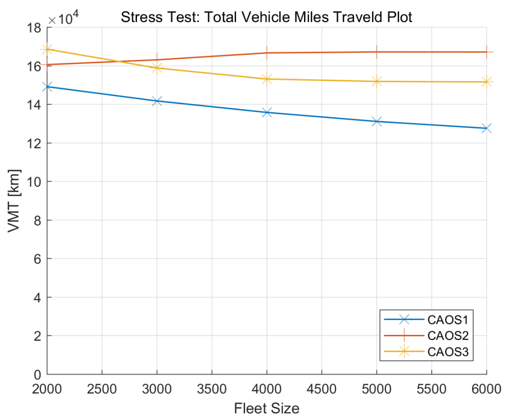

Figure 4.6: Stress Test: Total VMT

4.2 Testing on Different Scenarios

To examine the general applicability of the newly developed congestion-aware operation strategies for AMoD systems, we have tested them in several different scenarios. In this study, Berlin and Zürich scenarios are used. The setup of the scenarios and results are shown in the subsections below.

Berlin Scenario

Berlin scenario is adapted from the Open Berlin Scenario [41] which is available online. The original scenario is modified such that all the car trips will be served by the AMoD vehicles. The modified version of the scenario can be found on the AMoDeus website. The scenario contains the Berlin city and its surrounding area (see Figure 4.7 and 4.8). Due to the very large size, the scenario is scaled down to 12% of the original size and network capacity is also scaled down proportionally. After the scale down process, the population size is 357,506 and there are in total 509,825 AMoD trips throughout the day. The network is divided into 298 square regions of the same size to implement newly developed CAOSs. Parameters of the Routing algorithm for Berlin Scenario remain the same as in the Sioux Falls Scenario, which are listed in Table 3.2 in Section 3.4.1. The Vehicle Operating Strategy setup for the Berlin Scenario is shown in Table 4.1.

Parameter Name	**VOS1**	**VOS2**	**VOS3**
Dispatching algorithm			
dispatching frequency [second]	30	30	10
departure rate limit per region N_r	50	50	15
Rebalancing algorithm			
rebalancing frequency [second]	30	30	FF:10; FB:30
buffer β	N/A	0.1	N/A
rebalance rate limit N_{max}	N/A	50	50
normal region reference N^n	N/A	N/A	1
popular region reference N^h	N/A	N/A	10

Table 4.1: Berlin Scenario Vehicle Operating Strategy parameters

Figure 4.7: Berlin Scenario in Open Street Map

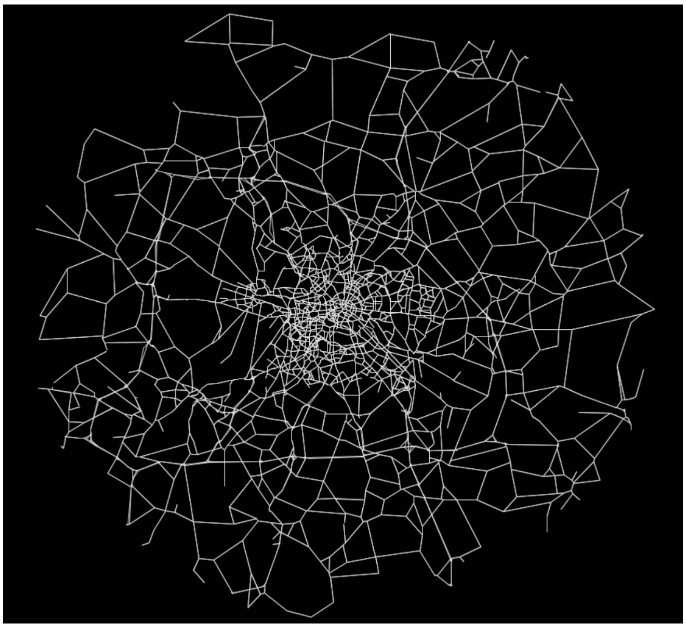

Figure 4.8: Berlin Scenario: prepared network

Simulation results of the Berlin Scenario are shown in Figure 4.9. Based on the stress test results, CAOS3 has higher resilience to stress. Thus, we have carried out additional test for CAOS3 with smaller fleet size and the result is also included in the table.

	Drive time [s]	Waiting Time [s]			Mean Total Journey Time [s]	VMT [km]	Fleet Size
		Mean	50%	95%			
Benchmark 1 (private cars)	1131	N/A			1131 + X	5,394,906	N/A
Benchmark 2 (based on [22])	2106	795	260	3040	2901	8,305,155	30,000
CAOS1	1113	93	40	390	1206	7,035,421	30,000
CAOS2	1051	1943	410	9300	2994	8,152,834	30,000
CAOS3	1050	83	20	410	1134	7,059,939	30,000
CAOS3	1049	102	20	510	1151	7,013,642	25,000
Lower Bound	855	Values depend on VOS					

*: X represents the time required for private car users to reach the car and to drive the car from parking place to the road.

Figure 4.9: Berlin Scenario Simulation Results

From the results, it can be seen that all of the newly constructed CAOSs can reduce the mean driving time in the Berlin Scenario. This means replacing private car trips with AMoD rides can improve the traffic even though there are more vehicles driving on the road and additional VMT. The comparison between the newly constructed CAOSs and existing work based on [22] reveals the importance of a good operating strategy for AMoD system.

We can also observe from the results that the waiting time for CAOS2 is very long. This is because of the uneven distribution of the travel demands and the very large size of the Berlin Scenario.

Most of the travel demands are concentrated in the city center area (i.e., in the center of the network in Figure 4.8). Therefore, spreading all available evenly across the network will lead to shortage of vehicles in the city center area while many vehicles are waiting for jobs outside the city. In addition, due to the large size of the map, sending vehicles from outer areas of the network to the inner areas will take very long time and the waiting time for the travel demands increases as a consequent. As a result, spreading the available vehicles evenly across the network is not suitable for a large scenario with unevenly distributed travel demands. In order to improve the performance of CAOS2 in such kind of scenarios, we need to adapt the rebalancing algorithm to the specific scenario. When generating rebalancing plans, we need to adjust the number of available vehicles to send to each region based on the distribution of travel demands over the network.

Zürich Scenario

The Zürich scenario is created by extracting data from the travel information of Switzerland provided by Sebastian Hörl from Institut für Verkehrsplanung und Transportsysteme (IVT). The data is highly representative to the reality. We cut out the network for city of Zürich and its surrounding area from the full data. To do so, we draw a circle centering at the geometric center of the Zürich city with a radius of 10 kilometers and keep all the network inside the circle (see Figure 4.10 and 4.11). And then we extract the population whose activities throughout the day are inside that circle. In order to compensate for the trips made by the population we have cut out due to having activities outside the circle, we generate artificial travel demands based on the distribution of the existing ones. The detailed process of artificial demands generation is included in the Appendix.

With this modification, the pattern of the augmented travel demands may diverge from that of the real Zürich region. Nevertheless, it is still a very suitable generality test for the newly developed operating strategies because the network of Zürich and the augmented travel demands are very different to Sioux Falls. The modified Zürich Scenario have a total of 126,634 travel demands throughout the day. The network has been divided into 324 rectangular regions of same sizes. Finally, the capacity of the network is scaled down to 0.18 based on the sub-population size. The parameter of Routing algorithm remains the same as in Sioux Falls Scenario (see Table 3.2 in Section 3.4.1). Parameters for the Vehicle Operating Strategies are listed in Table 4.2.

Parameter Name	**VOS1**	**VOS2**	**VOS3**
Dispatching algorithm			
dispatching frequency [second]	30	30	10
departure rate limit per region N_r	10	10	3
Rebalancing algorithm			
rebalancing frequency [second]	30	30	FF:10; FB:30
buffer β	N/A	0.1	N/A
rebalance rate limit N_{max}	N/A	10	10
normal region reference N^n	N/A	N/A	1
popular region reference N^h	N/A	N/A	10

Table 4.2: Zürich Scenario Vehicle Operating Strategy parameters

Simulation results are shown in Figure 4.12. Note that with the artificially generated travel demands, the private car case benchmark (i.e., Benchmark 1) is no longer accurate (see Appendix for details) and will not serve as a good reference point. Because when we generate the artificial travel demands, we only include the origin and destination locations and departure times. Other information, such as activities and attributes, are excluded as it requires more advanced techniques and data to generate which is beyond the scope of this study. Therefore, for Zürich Scenario, we will only compare the performance of the new CAOSs to the existing operating strategy derived from [22]. In addition, same as in Berlin Scenario, we carried out additional test for CAOS3 with a reduced fleet size to show the resilience of the AMoD system against stress.

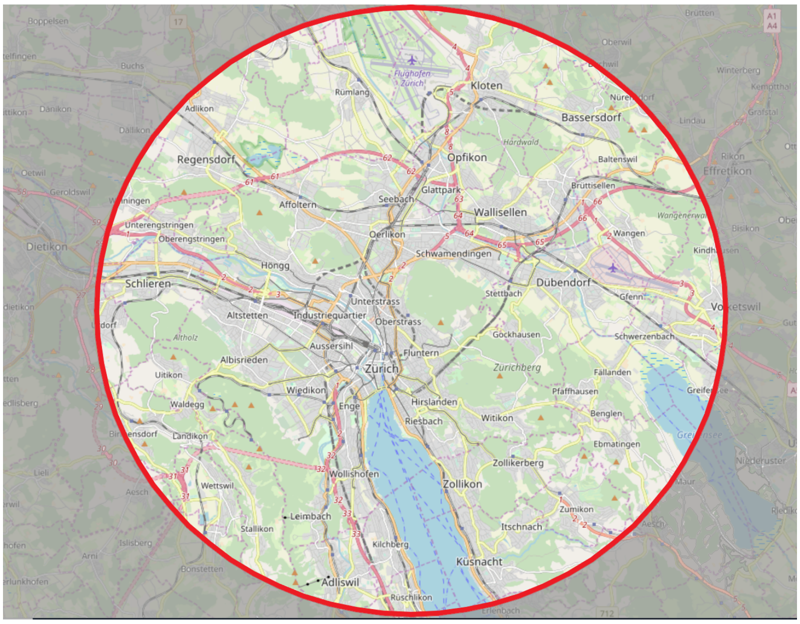

Figure 4.10: Zürich Scenario in Open Street Map

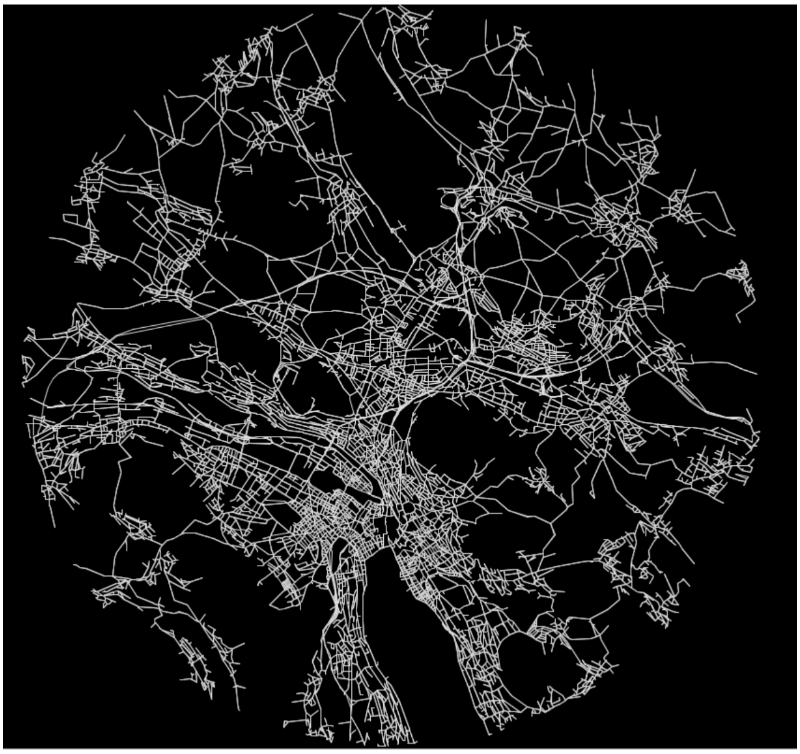

Figure 4.11: Zürich Scenario: prepared network

	Drive time [s]	Waiting Time [s]			Mean Total Journey Time [s]	VMT [km]	Fleet Size
		Mean	50%	95%			
Benchmark 2 (based on [22])	1536	749	104	2262	2285	822,719	10,000
CAOS1	611	237	87	897	848	708,652	10,000
CAOS2	608	393	79	1628	1001	820,248	10,000
CAOS3	711	169	64	623	880	823,080	10,000
CAOS3	708	200	81	768	908	832,935	7,500
Lower Bound	434	Values depend on VOS					

Figure 4.12: Zürich Scenario Simulation Results

From the results above, we can observe that our newly developed operating strategies can significantly reduce the time people spend on the roads compared to the existing operating strategy (Benchmark 2 based on [22]). This means the congestion awareness in our CAOSs are also effective in the Zürich Scenario. Similar to Berlin Scenario, the Zürich network is large, and the distribution of the travel demands are not even across the network. Due to these factors, the waiting time for the CAOS2 is not satisfying. As a result, adaptation of the CAOS2 to the network is needed to improve the service quality.

Summary of Testing Results

Simulation results in Berlin and Zürich Scenarios have shown that the good performance and congestion relieving effect of the CAOS1 and CAOS3 can be generalized to different scenarios. For CAOS2, although traffic congestion can still be relieved in different scenarios, the good service quality cannot be maintained due to the excessively long waiting time. This is because the rebalancing algorithm of CAOS2 is not suitable for large size scenarios with unevenly distributed travel demands. In order to improve the performance of CAOS2 in such scenarios, some adaptations are necessary.

In addition, due to the large scale of the Berlin and Zürich scenario, parameter tuning process is extremely time-consuming. As this study is not focused on developing an AMoD operating strategy for a specific scenario, we haven't carry out detailed parameter tuning. In the development process for the AMoD system in a specific city, studies on the pattern of travel demands and fine tuning of the parameters can be conducted. This will further improve the performance of the AMoD operating system.

4.3 Extension to Mixed Traffic Case

Up to now, we have relied on Assumption 1 in Section 3.2 that all private cars are converted to AMoD rides to carry out simulations. This is a good starting point for the study on the congestion effect of AMoD system. In reality, however, it is not very likely that all the private cars on the streets are replaced by centrally controlled autonomous vehicles. In this section, we will move one step closer to the reality by removing Assumption 1. By doing so, we extend our study to include the mixed traffic case, where both AMoD vehicles and private cars are used by the people in the city to travel.

As mixed traffic case is in general very complex, we need to simplify our system to some extent. We will keep the Assumption 2 that only private car trips are considered as the potential trips to be replaced by AMoD rides and there is only one operator in the city. Besides, in reality, people will choose their mode of transport based on many factors, such as cost and convenience. Modeling the choice of human beings is a very complex process and it is outside the scoop of this study. Therefore, we will carry out a sequence of simulations with different fraction of private car trips replaced by AMoD rides. Before each simulation, we set the replacement rate of AMoD and then randomly choose corresponding number of agents to switch from driving private cars to using AMoD. In addition, we will return to the Sioux Falls scenario to carry out the simulations on the mixed traffic case, because of its small size and concentrated travel flow during peak hours.

Since private vehicles are involved in mixed case simulation loops, MATSim iterative approach will be used. As we have fixed the mode choice at the beginning of the simulation, the iteration process for private cars is similar to the one we have carried out in Section 3.4.2. Adequate iterations are carried out with re-planning strategy of 0.75 change exponential beta and 0.25 reroute (innovation process of MATSim re-planning). Reroute will be turned off after 90% of the iterations. AMoD vehicles will also be included during the iteration process. Unlike private cars which will do the re-planning after each iteration, AMoD vehicles are operated by CAOS and will react to the travel demand and traffic during each iteration. Based on the pre-simulation results, 50 iterations are enough for the system to converge to the equilibrium for every AMoD replacement rate.

In order to reduce the effect of the randomness in the mode choice between private car and AMoD ride, for each replacement rate, multiple random seeds are used for the selection process of agents to change travel mode from driving private cars to taking AMoD rides. This will lead to multiple different population files. Simulations will be carried out based on the different population files and the results will be averaged.

When replacing the private cars with AMoD vehicles, we maintain the total mobility supply at the same level by applying the conversion rate between private cars and AMoD vehicles. As the first step moving toward mixed traffic case, we provide adequate AMoD vehicles to serve the fraction of the population who will use the service. We use the same ratio to replace existing private cars with AMoD vehicles as in Section 3.4.1: one autonomous vehicle will replace roughly three private cars. In the 100% replacement rate case, there will be 6,000 autonomous vehicles. For other cases, the number of autonomous vehicles used can simply be calculated by multiplying replacement rate to 6,000. The simulation points and setups for the mixed traffic study are listed in Table 4.3.

AMoD Replacement Rate	5%	10%	15%	20%	30%	40%	50%	75%
Fleet Size	300	600	900	1200	1800	2400	3000	4500
Number of Seeds Used	5	5	5	3	3	3	3	3

Table 4.3: Simulation points and setups for mixed traffic study

Note that more simulation points are chosen at the lower AMoD replacement cases and this is because of two reasons. First, AMoD is now in the emerging phase and it is likely that we will start from a small AMoD fleet and thus simulation results with a smaller AMoD replacement rate will be of more value to serve as a reference. Second, due to the higher computational workload to simulate autonomous vehicles in MATSim as well as the complexity of the vehicle operation algorithm itself, the time it takes to complete one iteration grows with the number of autonomous vehicles in the simulation. A smaller AMoD replacement rate will lead to less autonomous vehicles in the simulation and the time to run through each simulation will be significantly shorter than the cases where AMoD replacement rate is higher.

The results of the mixed case simulations are shown in the figures (4.13-4.16) below. The term SLTTE refers to the Smoothed Link Travel Time Estimation, which is introduced in section 3.3.1. As SLTTE relies on the knowledge of vehicles' future move and thus it is only effective when AMoD vehicles take up the great majority of traffic on the road. In the full AMoD case, applying this technique will improve the result to some extent, as suggested by the difference between the dotted line and the solid line in figure 4.13, 4.15 and 4.16.

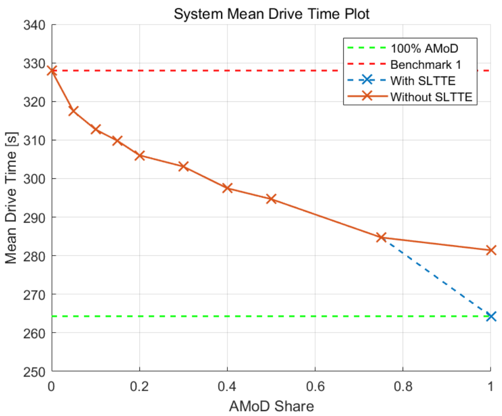

Figure 4.13: Mean drive time of whole network

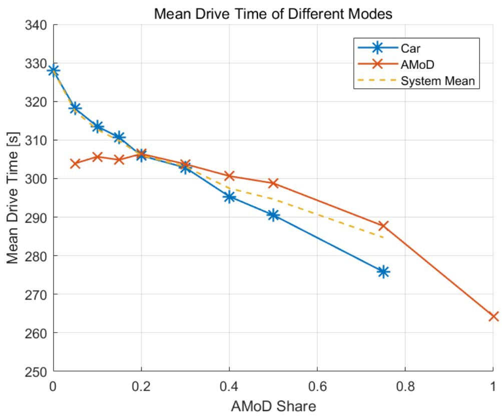

Figure 4.14: Mean drive time of different modes

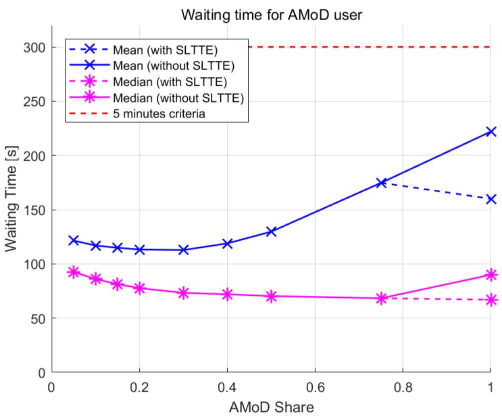

Figure 4.15: AMoD users waiting time

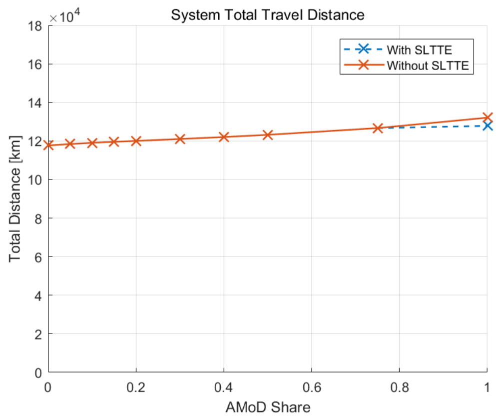

Figure 4.16: Total travel distance of all agents

From the results above, it can be observed that replacing private cars with shared autonomous vehicles will improve the overall traffic condition under any replacement rate and the overall traffic condition gets better as AMoD replacement rate increases. In figure 4.14, we split the mean drive time (on-board time) of private car drivers and AMoD users. The plot shows that when AMoD replacement rate is lower than 20%, the mean drive time (on-board time) of autonomous vehicle riders will be lower than that of private car users and when AMoD replacement rate is higher than 20%, private car users will spend less time traveling on the road than AMoD users. Figure 4.15

shows that the mean and median waiting time for AMoD users is well below 5 minutes criteria under all replacement rate, which is an indication for good service quality in terms of waiting time and majority of the people will not be discouraged to ride an autonomous car because of waiting time. Lastly, from figure 4.16 we can see that total travel distance for the whole traffic system increases as the replacement rate of AMoD increases. This is expected because autonomous vehicles have to cover the additional distance between two consecutive travel demands and with more AMoD ride, there will be more additional distance.

The plot of average drive time (on-board time) for different travel modes in Figure 4.14 is very interesting and it provide us an insight on the interaction on the route choice between autonomous vehicles and human-driven cars. At low AMoD replacement rate, there will be few autonomous vehicles on the road. Autonomous vehicles will follow a real time optimal path by constantly updating the travel time information and thus the time to travel on the road will be reduced. Since autonomous vehicles will actively take detour if there is congestion, private car users are also benefited to some extent thanks to the reduced number of vehicles present on the congested links. Due to the small AMoD replacement rate, however, this effect is limited. Therefore, AMoD users will have a shorter on-board time than the private car users. When AMoD replacement rate is higher, the positive effect of coordinated operation of autonomous vehicles will have a greater influence on the network and thus there will be less heavily congested links. While autonomous vehicles try to maintain a high social benefit, private car users are simply optimizing their personal interests and behave selfishly, and as a result, private car users benefit more from this condition.

This situation is similar to the prisoner's dilemma in Game Theory and the autonomous vehicles are cooperative players while private car users are selfish players. If everyone behaves selfishly (i.e., no autonomous vehicles), the system will reach the Wardrop's equilibrium and the total social benefits are low. If everyone chooses to cooperate (i.e., 100% AMoD case), the social benefits can be improved significantly. The only difference to a standard prisoner's dilemma lies in the mixed case, where some players behave cooperatively and the others only care for their own interest. Thanks to the positive effect of the active routing algorithm used by autonomous vehicles, the lose a cooperative player bears can be offset to some extent and at lower AMoD replacement rates, choosing to use AMoD will even lead to a higher benefit for the player. For higher AMoD replacement rates, however, the offset is not large enough to cover the loss and we have the same situation as in the standard prisoner's dilemma where selfish players are better off. AMoD operators and policy makers may be especially interested in this phenomenon and we will discuss this in Chapter 5.

Chapter 5

Discussion

5.1 Key Learnings

Is replacing private car trips with AMoD rides good for traffic?

Based on the results in this study, our answer to this question is yes. With a Congestion-Aware AMoD Operating Strategy, replacing all the private car trips in a city with AMoD rides will relieve traffic congestion and commuters can spend less time on the road. We have also extended our simulations to the mixed traffic case, where only some of the private cars are replaced by autonomous vehicles. Results show that traffic condition can still be improved in the mixed traffic case. Furthermore, the more private car trips are replaced by the autonomous vehicle rides, the better the overall traffic condition will be. This suggests that in the near future, when autonomous driving technology is ready for large scale application, introducing congestion-aware AMoD system into our cities may be one of the solutions to the traffic congestion problem.

One thing, however, needs to be pointed out is that we do not consider the potential induced trips by this new mode of transport. AMoD can provide very convenient taxi-like service but with only a fraction of the conventional taxi fare. It is possible that introducing AMoD into city may induce some new trips from other modes of transport than driving private cars (e.g., walking, cycling and public transport). With additional travel demands, the benefits of AMoD on the traffic network may be reduced. This can be a very important point to be considered by the policy makers and city planners. Some regulatory terms on the AMoD pricing and adaptation of public transport network may be necessary.

Who will benefit from AMoD?

Apart from the overall traffic condition, we should also look at who will be benefited the most by introducing AMoD service to our cities. Speaking of benefits, one may think of the business opportunities brought by this emerging technology. In fact, we don't need to look into the future to realize the potential of the mobility market by seeing the success stories of several online ride-hailing operators around the globe. Here, however, we will focus on the benefits of the people traveling in the network, or in other word the public interests.

In general, the time spent on the road, especially in congestion, is considered as undesirable and therefore we can improve the benefits of a group of people (i.e., commuters) by reducing the time they have to spend on the road. From the result of mixed case study, we can see that when only a very small size of AMoD fleet is introduced, AMoD user will be benefited more than the private car users and when AMoD replacement rate is higher (i.e., greater than 20% in Sioux Falls Scenario), private car users will save more time in the traffic. If we assume the cost of driving private cars and taking AMoD ride to be the same, then the usage rate of shared autonomous vehicles will converge to around 20% in Sioux Falls Scenario under current setting. If we want to increase the usage rate of the autonomous vehicles to further improve the traffic condition, incentives from price and policy are needed. Price incentive can be divided into two parts: reducing perceived cost of the time spending on traveling and reducing the fare of the ride. As an advantage of AMoD, sitting in an autonomous vehicle and relax can be more comfortable and enjoyable than maneuvering a car in the congested traffic and therefore the cost perceived by AMoD users on the time spent on the road can be lower compared to private car drivers. In this way, people may still choose to use AMoD to commute even though a little bit longer time need to be spent on the road compared to driving private cars. On the other hand, a lower fare on the AMoD ride can be another way to attract more users. The reduced fare can be the results of different factors at different level, such as more efficient vehicles and better autonomous driving algorithm or a policy from the authority, which tax private car users to compensate the AMoD riders who “behave” more cooperatively in the traffic.

Besides, the behavior of the autonomous vehicles is adjustable and the level of cooperativeness (or selfishness) can be tuned by setting different parameters. This will also have a big influence on the social benefits of the whole traffic system. By behaving more aggressively, the average time AMoD riders spend on the road may be reduced at the cost of total social benefits. For a private AMoD operator, it is desired as shorter ride time will attract more users to use the service. For city’s transportation authority, on the other hand, this should be avoided since the benefits of all travelers across the city is more important.

Bridging the gap among previous studies

In this study, we have not only come up with techniques to tackle the congestion issue, such as introducing new online routing algorithm and limiting departure rate, but also bridged the gap among different previous results. While there are some rigorous proved theoretical results on the AMoD operation, they are limited to the highly simplified network and some very strict assumption, which sometimes make those results difficult to be directly applied in the real world. On the other hand, many simulation-based studies on the topic of AMoD rely on the very basic operational strategies and more focus is given to the other important aspects, such as behaviors of human beings, pricing and policies. Therefore, as mentioned in Literature Review section, we have seen some contradictory opinions on whether it is good to introduce AMoD into cities at a large scale.

To bridge this gap, we have adapted some of the well formalized theoretical algorithm to the realistic simulation environment. The results have shown that the well-designed vehicle operation strategy, along with congestion-aware elements, can indeed improve the performance of AMoD significantly. In the scenarios we have used in this study, the improvement is so large that it can turn the tide to the favor of the AMoD system. This means a good operation strategy is the key to realize the potential benefits of the AMoD system and it deserve more attention. Meanwhile, a good strategy should be not only based on the theoretical foundation, but also applicable and adaptable to real world scenarios. Similar to many of the transportation problems, AMoD operation in large cities is a very complex problem, an approach with a good combination between theory and practical application will be necessary to solve the problem.

Fleet size versus total Vehicle Miles Traveled (VMT)

For AMoD operators, fleet size and total VMT are two major sources of cost. It is desirable to keep them both low for a lower operational cost but reducing fleet size is usually accompanied by the increase in total VMT. From the comparison between the more advanced strategy, CAOS3, and the simpler strategy, CAOS1, it can be observed that while CAOS3 enables the travel demands to be served by a smaller fleet without sacrificing service quality, it also increases VMT. Although the additional driving distance does not lead to additional congestion thanks to congestion-aware operating strategy, the cost associate to the additional VMT, such as fuel cost and maintenance cost, cannot be saved. Moreover, this additional VMT is due to the imbalanced distribution of the travel demands, which means empty VMT will inevitably go up as we are trying to reduce the fleet size. Therefore, AMoD operators need to trade-off between fleet size and VMT and find the most suitable combination and operation strategy based on their portfolio.

5.2 Possible Future Extensions to This Study

Include Cost Analysis

In the study on the mixed traffic case (i.e., both AMoD vehicles and private cars are present in the network), we have fixed the mode choice of all the agents throughout the iteration process. We carry out sequential simulations with a series of hypothetical replacement rates of private car trips by AMoD rides. This can provide us a general idea on what potential influence introducing different scales of AMoD service will have on the traffic network. In the future work, we can include cost analysis and allow agents to freely choose between driving private cars and taking AMoD rides.

With cost analysis, we can model the preference structure of the agents in the simulation loop. The cost analysis not only includes the explicit monetary costs, such as fuel cost or fare for AMoD rides, but also takes implicit costs, such as travel time, opportunity costs and comfort, into consideration. For example, maneuvering a car through congested traffic network can be much more demanding than sitting back on an autonomous vehicle and enjoying the ride, and therefore the traveling time for the former mode will have a higher cost than the later one. On the other hand, with a limited AMoD fleet size, if too many people choose to use AMoD, then waiting time will increase and some AMoD users may not be able to arrive at their destination on time (e.g., late for work). The excess waiting time and late arrival are obviously not desirable, and thus a high cost will be associated to the journey. During the iteration process, agent will explore both travel modes (i.e., driving private cars and using AMoD) and choose the one that yield less cost. With this, the simulation will converge to an equilibrium point that has a higher resemblance to reality.

Study on Induced Travel Demands

As mentioned in the Literature Review, AMoD may induce some new travel demands. Since AMoD can provide very convenient door to door service with a relatively low price, some people may switch from walking, cycling and taking public transport to using AMoD. Serving those additional travel demands by AMoD is likely to increase the burden to the traffic network. In the future work, we can include those induced trips into our analysis.

This can be done by extending the cost analysis to all modes of transport. By associating cost to different modes of transport, agents in the simulation loops can choose the transport mode minimize their cost. For example, to attract an agent to switch from taking public transport to using AMoD, the time and effort saved by riding an autonomous vehicle need to be greater than the increase in the fare. As can be seen from this example, the induced travel demands directly relies on the cost structure. In addition, the conversion between implicit cost and explicit cost requires a multidisciplinary study and it can vary significantly among different cities. Therefore, a suitable cost structure is the key for the study on the induced demands.

Enable Ride Sharing

AMoD is an idea based on sharing economy where several users share a vehicle instead of owning one. We can move one step further to allow several commuters to not only share a vehicle but also to share the ride. In fact, ride sharing is considered as a method to relieve the congestion. With ride sharing, several passengers are on board one vehicle and there will be less vehicles running on the road. This can be especially effective against the congestion on critical links, such as the major roads that connects residential areas and working places. With less vehicles running on the road, congestion can be reduced. There are many studies on ride sharing algorithms that can optimize the matching between vehicles and customers. But most of the available studies do not take congestion effect into consideration as the matching problem for ride sharing is already a very complex problem. In future works, relieving congestion can also be included as an objective in the optimization process so that ride sharing algorithms can also become congestion-aware.

It should also be pointed out that enabling ride sharing in AMoD system can be a challenging task. To begin with, although ride sharing has been a highly regarded method to improve the mobility for the whole society and tackle the congestion problem, it may not be as effective as suggested by the results in many literatures, many of which relies on highly simplified traffic model. According to a recent study on the performance of different ride sharing algorithms [42], where a more realistic simulation platform is used to test different algorithms, the performance of all the sharing algorithms fall a distance behind the claimed results in original studies and the potential fare reduction is very limited. Besides, while ride sharing will reduce number of vehicles on the major links, where the routes of many travel demand overlap, on smaller links, where there is much less overlapping of route among travel demands, multi-capacity AMoD vehicles need to travel back and forth to serve all the passengers on board. For example, during the rush hours in a region with many workplaces (e.g., commercial center, industrial area) or a residential region, many vehicles need to travel back and forth to pick up or drop off different passengers and this can be a very heavy burden for those smaller roads inside these regions. Last but not least, there are also some practical issues with ride sharing on autonomous vehicles, such as passengers may not find themselves feeling comfortable sharing a very small space in a car with several strangers. With a compromised comfort and only limited reduced fare, whether a passenger will choose to share a ride remains a question.

Study on The Interaction Between AMoD and Public Transport

A very interesting future extension to current work is to integrate the AMoD into the public transportation network. In addition to the possible alternative choice for driving private vehicles, AMoD can also be operated in a way such that it become a complement to the public transport.

Today, even without autonomous driving technology, some cities are trying to replace the conventional public transport with ride hailing or ride-sharing service operated by companies like Uber. For example, in a Canadian city Innisfil, Ontario, instead of investing in the public transportation, the city government subsidize ride within the city to make the fare comparable to the public transport ticket, and the service is very popular among the residents in that city [43] [44]. When autonomous driving is introduced, such replacements of public transport with autonomous vehicles ride can become even more popular because of the lower fare. In this way, AMoD may be viewed as a competitor to the public transport. And as a study on the reaction of the public transport network to the rising of autonomous vehicles indicates that it is likely for the public transport network to shrink in the future [45].

On the other side, conventional public transport like bus, tram and train do possess some distinct characteristic that is not fully replaceable by the autonomous vehicles and efficiency is on the top of that list. As suggested by [1], with a reduced price to ride autonomous vehicles, more AMoD trips are likely to be induced and some of those induced trips were originally public transport rides. This will reduce the overall efficiency of the transport system and may lead to more congestion on the road. One countermeasure to this problem is to consider AMoD as a part of the public transportation network and by actively operate the fleet and setting the price, more people can be encouraged to use public transport. For example, after a person being picked up by an autonomous vehicles from home, if the traffic condition along the way is poor, then instead of delivering him/her to the destination directly, the autonomous car will send the passenger to a major public transport stop (e.g., train station, bus stop). By taking public transport, the congestion can be bypassed easily thanks to the higher priority of the public transport (e.g., dedicated track for train and tram, bus lane, traffic light control). After arriving at the public transport stop near the destination, the rider may be picked up by another autonomous vehicle or simply walk to the final destination. With a good interaction between AMoD and public transport, the transfer process can become seamless as the autonomous vehicles will choose the most suitable stop for passengers to catch up a public transport service based on the timetable. In fact, some public transport company has started operating the integrated mobility service and car sharing program, such as Green Class from SBB [46] and Mobility Car Sharing [47]. With the rising of autonomous driving technology, AMoD may also be integrated to the system in the near future, to provide better door to door mobility service in cooperation with public transport.

Chapter 6

Conclusion

In this work, we have explored the congestion effect of operating Autonomous Mobility-on-Demand (AMoD) systems in cities and developed several congestion-aware operation strategies for the AMoD systems. There are many existing operating strategies in literature, but many of them do not take congestion effects into consideration and the result cannot be directly applied to the real world operation. Among existing studies that include congestion effect, no consensus has been reached on the potential influence of introducing AMoD into cities. While some studies suggest that introducing autonomous ride sharing service will aggravate the poor traffic condition suffered by many major cities around the globe, other studies conclude that AMoD will be the solution to congestion problems we are facing nowadays.

To mitigate the disagreement among different studies, we have developed several Congestion-Aware Operating Strategies and tested them on a realistic traffic simulation platform (MATSim). The newly constructed strategies not only contain some innovative techniques in relieving the congestion, but also include elements from some of the well formalized theoretical results. We bridge the gap between the theoretical algorithms and realistic simulation environment by making adaptations in our implementations. Results from our simulations indicate that replacing all private car trips with AMoD rides will not aggravate the traffic congestion, as proposed in some previous studies, and can even improve the overall traffic condition, if our Congestion-Aware Operating Strategies are used. This demonstrates the importance of a good operating strategy and implies that AMoD can be a potential solution to the traffic congestion problem. Furthermore, we have managed to show the general applicability of our operating strategies by carrying out simulations in different scenarios. Finally, we have also extended our simulations to the mixed traffic case, where both autonomous vehicles and human-driven car are present in the network. Results indicate that replacing a fraction of the private car trips with AMoD rides can still relieve traffic congestion. Moreover, the overall traffic condition improves monotonically as the number of private car trips replaced by AMoD rides increases.

As transportation and traffic problems are in general very complex, results of this study can serve as a starting point for many interesting further researches and several potential extensions to this study have been proposed.

Acknowledgement: This work is carried out under the joint supervision and support from Claudio Ruch and Prof. Dr. Emilio Frazzoli from the Institute of Dynamic Systems and Control (IDSC) and Sebastian Horl from Institut für Verkehrsplanung und Transportsysteme (IVT). In addition to the supervisors, special thanks are also given to Prof. Dr. Kay Axhausen (IVT) and Jan Hakenberg (IDSC) for valuable insights, feedback and support on this work, Dr. Marcel Rieser for the license of Via visulization software and Dr. Michal Maciejewski for the discussion on MATSim DVRP module.

Chapter 7

Appendix

Visualization video

The link to the full video of visualization process is available upon request: luc@student.ethz.ch

Generating Artificial Travel Demands in Zürich Scenario

First, we divide the full day into 15-minutes intervals. Within each time interval, the travel demands can be considered as relatively constant. Then we collect the travel demand distribution among different regions. In specific, we collect the departure rate of travel demands from region i to region j during the time interval t and we denote it as $\alpha_{ij}(t)$. By summing up $\alpha_{ij}(t)$ over j (as shown in Equation 7.1), we get departure rate α_i of region i.

$$\alpha_i(t) = \sum_j \alpha_{ij}(t) \tag{7.1}$$

To generate a virtual travel request, we need to have departure location, arrival location and departure time. We will generate the artificial request based on the distribution of the real travel demand. When choosing the departure location of a request, we first choose the region and then pick a random link in that region as the departure location of the request. A region will be chosen as the departure region of the request with probability of $p(i)$ where $p(i, t)$ is calculated by normalizing the departure rate of that region to the total departure rate as shown in Equation 7.2. After choosing the departure region, we assign the request's departure location to a random link in that region.

$$p(i, t) = \frac{\alpha_i(t)}{\sum_i \alpha_i(t)} \tag{7.2}$$

With the chosen departure region, we will choose the arrival location of the request in a similar way. To choose the arrival region, we calculate $p_i(j, t)$ for each region j by Equation 7.3, which is the probability of a request arrives in region j given that it departs in region i. Then we will choose the arrival region of the request based on the probability distribution. Region j will be chosen as the arrival region of the request with probability $p_i(j, t)$. After choosing the region, a random link in that region will be set as the arrival location of the virtual request.

$$p_i(j, t) = \frac{\alpha_{ij}(t)}{\alpha_i(t)} \tag{7.3}$$

The departure time for the virtual request will be randomly chosen from the 15-minutes time interval. In the MATSim simulation environment, the resolution of the time is second. 15 minutes correspond to 900 seconds. We will uniformly choose one second from the 900 seconds as the departure time of the virtual request.

In Zürich scenario, the number of virtual requests we generated equals to 100% of the real requests in the data. This means the total travel demands is twice as many as in the original data. With the artificially generated travel demands, the scenario is no longer representative to the real city of Zürich, but it still serve as a good general applicability test for the newly developed Congestion-Aware Operating Strategies.

Problems with Benchmark 1 in Zürich Scenario

In Zürich Scenario, due to the existence of artificially created travel demands, the MATSim iteration process does not converge. Even with very small probability (0.05) for agents to explore new routes during the re-planning process, there are still obvious fluctuations in the scoring plot and distance plot (Figure 7.1 and 7.2).

In addition, When we generate the artificial travel demands, we only include the origin and destination locations and departure times. Other information, such as activities and attributes, are excluded as it requires more advanced techniques and data to generate which is beyond the scope of this study. For that reason, Benchmark 1 (i.e., private car benchmark) is no longer accurate. For Zürich Scenario, we will only compare the performance of the new CAOSs to the existing operating strategy derived from [22].

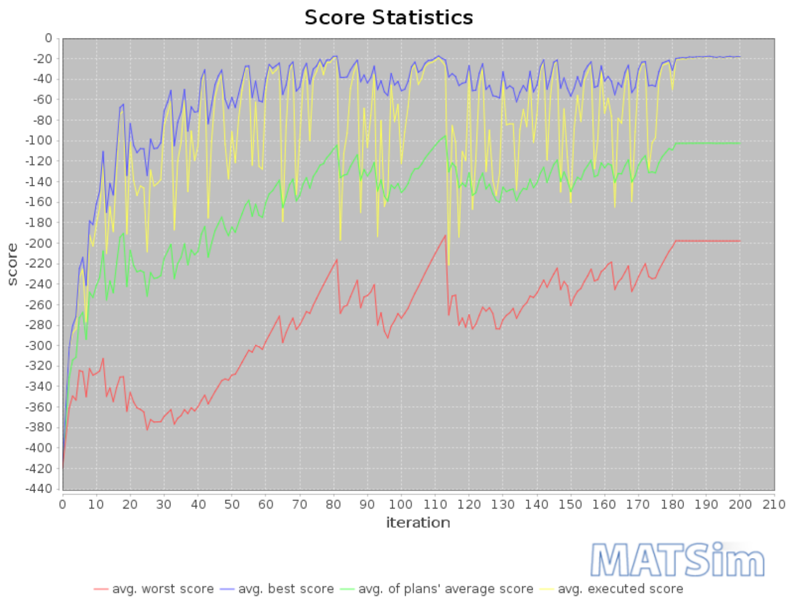

Figure 7.1: Zürich Benchmark 1: not converged scoring plot

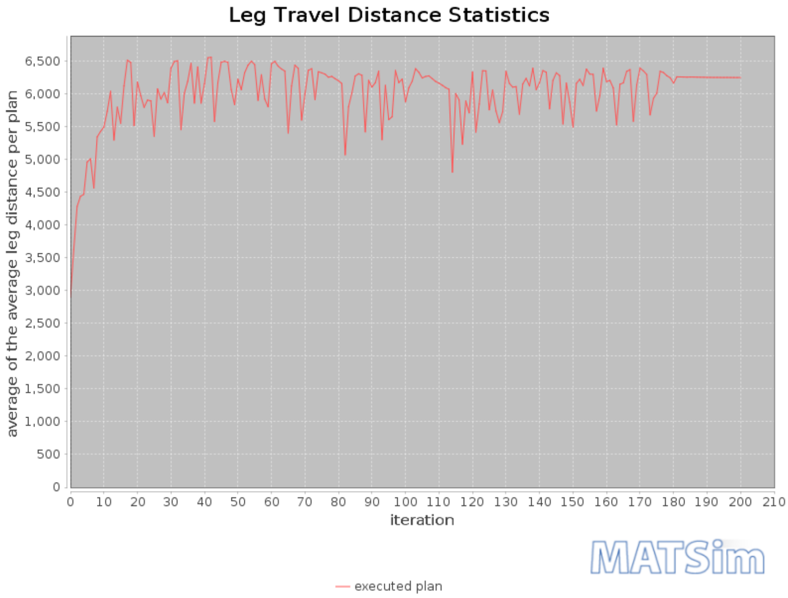

Figure 7.2: Zürich Benchmark 1: not converged VMT plot

Bibliography

[1] S. Hörl, F. Becker, T. Dubernet, and K. W. Axhausen, *Induzierter Verkehr durch autonome Fahrzeuge: Eine Abschätzung*, Bundesamt für Strassen, Eidgenössisches Departement für Umwelt, Verkehr, Energie und Kommunikation UVEK and Institut für Verkehrsplanung und Transportsysteme, ETH Zürich, 2019.

[2] A. Noto, *NYC economy may be losing $20 billion a year due to traffic congestion*, New York Business Journal.

[3] *How air pollution is destroying our health*, World Health Organization, https://www.who.int/air-pollution/news-and-events/how-air-pollution-is-destroying-our-health.

[4] J. G. Wardrop, *Some Theoretical Aspects of Road Traffic Research*, ICE Proceedings of the Institution of Civil Engineers. 1 (3): 325–362. doi:10.1680/ipeds.1952.11259, 1952.

[5] *Federal Highway Administration (FHWA), United States Department of Transportation*, https://www.fhwa.dot.gov/.

[6] K. Treleaven, M. Pavone, and E. Frazzoli, *Asymptotically Optimal Algorithms for Pickup and Delivery Problems with Application to Large-Scale Transportation Systems*, IEEE Transactions on Automatic Control, 2013.

[7] L. Ruschendorf, *The wasserstein distance and approximation theorems*, Probability Theory and Related Fields, vol. 70, pp. 117-129, 1985.

[8] R. Zhang, K. Spieser, E. Frazzoli, and M. Pavone, *Models, Algorithms, and Evaluation for Autonomous Mobility-On-Demand Systems*, American Control Conference, 2015.

[9] M. Pavone, S. L. Smith, E. Frazzoli, and D. Rus, *Load Balancing for Mobility-on-Demand Systems*, Robotics: Science and Systems, 2011.

[10] K. Spieser, S. Samaranayake, and E. Frazzoli, *Vehicle Routing for Shared-Mobility Systems with Time-Varying Demand*, American Control Conference, 2016.

[11] F. R. Rick Zhang and M. Pavone, *Analysis, Control, and Evaluation of Mobiliby-on-Demand Systems: a Queueing-Theoretical Approach*, Transactions on Control of Network Systems.

[12] R. Iglesias, F. Rossi, K. Wang, D. Hallac, J. Leskovec, and M. Pavone, *Data-Driven Model Predictive Control of Autonomous Mobility-on-Demand Systems*, arXiv:1709.07032, 2017.

[13] C. Ruch, J. Gachter, J. Hakenberg, and E. Frazzoli, *The +1 Method Model-Free Adaptive Repositioning Policies for Robotic Multi-Agent Systems*, ETH Zurich, 2019.

[14] M. W. Levin, K. M. Kockelman, S. D. Boyles, and T. Li, *A general framework for modeling shared autonomous vehicles with dynamic network-loading and dynamic ride-sharing application*, Computers, Environment and Urban Systems, 2017.

[15] L. Jiangtao, P. Mirchandani, and Z. Xuesong, *Problem decomposition and approximation for shared mobility applications with endogenous congestion: integrated vehicles assignment and routing in capacitated transportation networks*, Transportation Research Part C.

[16] M. W. Levin, *Congestion-aware system optimal route choice for shared autonomous vehicles*, Transportation Research Part C, 2017.

[17] D. Bertsimas, P. Jaillet, and S. Martin, *Onlin Vehilce Routing: The Edge of Optimization in Large-Scale Applications*, Operations Research.

[18] Z. Li, Y. Hong, and Z. Zhang, *An empirical analysis of on-demand ride-sharing and traffic congestion*, Proceddings of the 50th Hawaii International Conference on System Sciences 2017.

[19] G. D. Erhardt, S. Roy, D. Cooper, B. Sana, M. Chen, and J. Castiglione, *Do transportation network companies decrease or increase congestion?*, SCIENCE ADVANCES.

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

[21] F. Rossi, R. Zhang, and M. Pavone, *Congestion-Aware Randomized Routing in Autonomous Mobility-on-Demand Systems*.

[22] M. Maciejewski and J. Bischoff, *Congestion Effects of Autonomoous Taxi Fleets*, Transport, Special Issue on Multi-Stakeholder Collaboration in Urban Transport, 2017.

[23] S. Horl, M. Balac, and K. W. Axhausen, *A first look at bridging discrete choice modeling and agent-based microsimulation in MATSim*, The 7th International Workshop on Agent-based Mobility Traffic and Transportation Models, Methodologies and Applications 2018.

[24] A. Horni, K. Nagel, and K. W. Axhausen, *A Closer Look at Scoring. In: Horni, A, Nagel, K and Axhausen, K W. (eds.) The Multi-Agent Transport Simulation MATSim, Pp. 23–34.*, London: Ubiquity Press. DOI: http://dx.doi.org/10.5334/baw.3. License: CC-BY 4.0, 2016.

[25] M. Balac, H. Becker, F. Ciari, and K. W. Axhausen, *Modeling competing free-floating carsharing operators- A case study for Zurich, Switzerland*, Transportation Research Part C, 2018.

[26] M. W.Levin and S. D.Boyles, *A cell transmission model for dynamic lane reversal with autonomous vehicles*, Transportation Research Part C: Emerging Technologies, 2016.

[27] K. Dresner and P. Stone, *Multiagent Traffic Management: A Reservation-Based Intersection Control Mechanism*, Proceeding AAMAS '04 Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems - Volume 2, Pages 530-537, 2004.

[28] S. Horl, M. Balac, and K. W. Axhausen, *Dynamic demand estimation for an AMoD system in Paris*.

[29] A. Horni, K. Nagel, and K. W. Axhausen, *The Multi-Agent Transport Simulation MATSim*, London: Ubiquity Press DOI: http://dx.doi.org/10.5334/baw. License: CC-BY 4.0, 2016.

[30] C. Ruch, S. Hörl, , and E. Frazzoli, *Amodeus, a simulation-based testbed for autonomous mobility-on-demand systems*, In: Proc. 21th IEEE Conf. Intelligent Transportation Systems, 2018.

[31] A. Horni, K. Nagel, and K. W. Axhausen, *Within-Day Replanning. In: Horni, A, Nagel, K and Axhausen, K W. (eds.) The Multi-Agent Transport Simulation MATSim, Pp. 187-200.*, London: Ubiquity Press. DOI: http://dx.doi.org/10.5334/baw.3. License: CC-BY 4.0, 2016.

[32] J. de Dios Ortuzar and L. G. Willumsen, *Modelling Transport*, Chapter 10 and Chapter 11, pp.321-394, John Wiley and Sons LTD, 2001.

[33] H. W. Kuhn, *The Hungarian Method for the assignment problem*, Naval Research Logistics Quarterly, 2: 83–97, 1955.

[34] M. Pavone, K. Treleaven, and E. Frazzoli, *Fundamental Performance Limits and Efficient Policies for Transportation-On-Demand Systems*, 49th IEEE Conference on Decision and Control, 2010.

[35] A. Chakirov, *Sioux Falls. In: Horni A, Nagel K and Axhausen K W. The Multi-Agent Transport Simulation MATSim, Pp. 385-388*, London: Ubiquity Press. DOI: http://dx.doi.org/10.5334/baw.59. License: CC-BY 4.0, 2016.

[36] *Open Street Map*, https://www.openstreetmap.org.

[37] K. Spieser, K. Treleaven, R. Zhang, E. Frazzoli, D. Frazzoli, and M. Pavone, *Toward a systematic approach to the design and evaluation ofautomated mobility-on-demand systems: A case study in Singapore*, In: Road vehicle automation. Springer, pp.229-245, 2014.

[38] D. Fagnant and K. Kockelman, *Dynamic ride-sharing and fleet sizing for a system of shared autonomous vehicles in Austin, Texas*, Transportation 45(1), 2016.

[39] J. Bischoffa and M. Maciejewskiba, *Simulation of City-wide Replacement of Private Cars with Autonomous Taxis in Berlin*, Procedia Computer Science, Volume 83, 2016, Pages 237-244, 2016.

[40] M. Rieser, *Via*, https://simunto.com/via/.

[41] D. Ziemke, I. Kaddoura, and K. Nagel, *The MATSim Open Berlin Scenario: An openly available agent-based transport simulation scenario based on synthetic demand modeling and Open Data*, Transport Systems Planning and Transport Telematics, Technische Universit ̈at Berlin, 2019.

[42] C. Ruch, L. Sieber, C. Lu, and E. Frazzoli, *Ride-Sharing: Is It a Good Idea?*, under review, 2019.

[43] *The Innisfil experiment: the town that replaced public transit with Uber*, The Guardian, https://www.theguardian.com/cities/2019/jul/16/the-innisfil-experiment-the-town-that-replaced-public-transit-with-uber.

[44] L. Bliss, *‘Uber Was Supposed To Be Our Public Transit’*, CITYLAB, https://www.citylab.com/transportation/2019/04/innisfil-transit-ride-hailing-bus-public-transportation-uber.

[45] P. Manser, *Public Transport Network Design in a World of Autonomous Vehicles*, Master Thesis, Institute for Transport Planning and Systems (IVT), ETH Zurich.

[46] *SBB Green Class*, https://www.sbb.ch/de/abos-billette/abonnemente/greenclass.html.

[47] *Mobility – more than car sharing*, https://www.mobility.ch/en/.

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute for Dynamic Systems and Control

Prof. Dr. R. D'Andrea, Prof. Dr. E. Frazzoli, Prof. Dr. C. Onder, Prof. Dr. M. Zeilinger

Title of work:

Congestion-Aware Operation of Coordinated Autonomous Mobility-on-Demand Systems

Thesis type and date:

Master's Thesis, August 2019

Supervision:

Claudio Ruch
Sebastian Hörl
Prof. Dr. Emilio Frazzoli
Prof. Dr. Kay Axhausen

Student:

Name:	陆承骐 Chengqi Lu
E-mail:	luc@student.ethz.ch
Legi-Nr.:	17-940-735
Semester:	FS 2019

Statement regarding plagiarism:

By signing this statement, I affirm that I have read and signed the Declaration of Originality, independently produced this paper, and adhered to the general practice of source citation in this subject-area.

Declaration of Originality:

`https://www.ethz.ch/content/dam/ethz/main/education/rechtliches-abschluesse/leistungskontrollen/declaration-originality.pdf`

Zurich, 16.8.2019: 陆承骐 Lu Chengqi