Demand-Driven Operational Design for Shared Mobility with Ride-pooling Options

Center for Transportation, Environment, and Community Health
Final Report

By

Dongfang Zhao, Xiaotong Guo, Xiaopeng Li, Samitha Samaranayake

April 30, 2020

DISCLAIMER

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No.	2.Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Demand-Driven Operational Design for Shared Mobility with Ride-pooling Options		5. Report Date Apr, 30, 2020 6. Performing Organization Code
7. Author(s) Dongfang Zhao (ORCID ID # 0000-0002-5424-9915) Xiaotong Guo (ORCID ID # 0000-0003-3993-3716) Xiaopeng Li (ORCID ID # 0000-0002-5264-3775) Samitha Samaranayake (ORCID ID # 0000-0002-5459-3898)		8. Performing Organization Report No.
9. Performing Organization Name and Address Civil and Environmental Engineering 4202 E. Fowler Avenue ENG 207 University of South Florida Tampa, FL 33620-5350		10. Work Unit No. 11. Contract or Grant No. 69A3551747119
12. Sponsoring Agency Name and Address U.S. Department of Transportation 1200 New Jersey Avenue, SE Washington, DC 20590		13. Type of Report and Period Covered Final Report 10/1/2018 – 04/13/2020 14. Sponsoring Agency Code US-DOT
15. Supplementary Notes		

16. Abstract

This project aims to develop a demand-driven approach for shared mobility operations with machine learning and math programming methods. The objective of this approach is to incorporate economic, environment and equity impacts over an entire operational cycle. Both ride-hailing systems (e.g. Lyft) and ride-pooling systems (e.g. UberPool) will be investigated. The developed models are tested with real-world taxi data including detailed trajectories of vehicles and their loading states at all times. We proposed a deep Q learning model to optimize the system performance. In the model, we train a Q network and operate the system with real-time demands. Services include serving travel demands, rebalance, charging in charging stations. Case study in NYC is explored and the results are analyzed in the cast study section. Further, we propose a mathematical programming approach for solving the vehicle routing problem for a high-capacity ride-pooling system. This approach attempts to extend the existing Request-Trip-Vehicle-graph based method that used heuristic methods to generate vehicle routes into one that use customized column generation to generate candidate optimal vehicle routes more efficiently.

17. Key Words	18. Distribution Statement
Q learning, ride sharing, ride pooling, electric vehicle, gravity model, column generation	Public Access; part of the material may be used in the incoming technical paper publications.

19. Security Classif (of this report)	20. Security Classif. (of this page)	21. No of Pages	22. Price
Unclassified	Unclassified		

Form DOT F 1700.7 (8-69)

Demand-Driven Operational Design for Shared Mobility with Ride-pooling Options

1. Introduction and Literature Review

Through being one of the most important components in the modern society, urban transportation systems often bear server congestion and become a major cause to air pollution. In 2014, transportation accounts for 26% of all greenhouse gas emissions in the United States (U.S. DoE, 2016). One of the factors contributing to the significant environmental impacts of the transportation system is its inefficiency. According to the recent national household travel survey data, the average occupancy of personal vehicles is 1.6 (FHWA, 2011), indicating that most cars do not run at their full capacity. Increasing vehicle occupancy rate in private and public vehicles (e.g., taxis) through ride sharing offers great opportunities in improving the transportation system's efficiency. In recent years, with the growth of the sharing economy and the development of electric vehicles (EVs) (Teubner et al., 2014; Krueger et al., 2016; Wadud et al., 2016; Gurumurthy and Kockelman, 2018), ride sharing with EVs has emerged as a potential solution to improve efficiency of transportation systems. Ma et al. (2015) also showed that using ride sharing, 2.2 million kg of carbon dioxide can be saved every year in Beijing. Ride sharing is also able to connect more people with timely and flexible service and helps reduce travel costs and enables more affordable trips. Due to these advantages and benefits, ride sharing using personal vehicles has been tested and implemented in various platforms, such as Uber and Lyft. Uber claims that about 20% of its rides globally are shared rides using UberPool (Fortune, 2016).

Researchers have been seeking solutions to optimize the service design and improve operational performance of ride sharing systems. Earlier studies (Barth and Todd, 1999; Galland et al., 2014) have focused on the traditional ride sharing, for which the ride sharing is pre-arranged and often has the same trip origins and/or destinations. For example, Caulfield (2009) analyzed one day's commute trip data (reported as part of a Census survey) in Dublin, Ireland and found that 4% of the respondents share rides to work. They estimated that this ride sharing reduced 12,674 t of CO_2 emissions annually. Hong et al. (2017) proposed a clustering algorithm on GPS trace data to match trips and select routes for carpooling. Most recently, Dong et al. (2018) analyzed data from China's ride sharing service DiDi and concluded that it is a viable mode of transportation to complement taxis in serving increasing demand.

In recent years, enabled by the development of information technologies, demand-driven ride sharing has received increasing attentions. Demand-driven ride sharing allows shared rides to form among strangers who do not know each other's trip itinerary. The higher flexibility of demand-driven ride sharing offers additional opportunity to maximize sharing benefits and improve system efficiency. In a demand-driven ride sharing system, it is critical to match the appropriate riders to form the shared ride. Therefore, many researchers focus on developing algorithms for ride matching. In particular, Kleiner et al. (2011) proposed an auction mechanism to match demands and tested its performance using the map of Freiburg, Germany with simulated rides randomly sampled from a uniform distribution. Agatz et al. (2011) compared the optimization-based approach with a rule-based greedy matching algorithm using demand data from Atlanta, Georgia and concluded that optimization methods have better system performance in matching rides and reducing total system vehicle-miles-traveled. However, to simplify the analysis, these studies limited the number of demands that can be shared at a time to be two (i.e., maximally, two passengers can share a vehicle). More recent research has proposed flexible models to optimize passenger-vehicle matching and vehicle routing, considering the vehicle capacity and the number of passengers traveling together (Lin et al., 2012; Santos and Xavier, 2015). Levin (2017) designed a model to optimize route choice for autonomous vehicles, considering congestions due to other vehicles in the network, using linear optimization models. Other recent research, Li et al. (2016) has focused on finding an optimal route choice model for last mile parcel delivery using shared autonomous vehicles. Qian et al. (2017) proposed a group-ride system, in which different groups of riders gather at a predefined location and are picked up together. However, different from the door-to-door service provided by traditional taxis, group ride requires the riders to walk to and from the taxi pick-up and drop-off locations, reducing the convenience of taking taxis. Santi et al. (2014) introduced the concept of share-ability networks and proposed a mathematical model to quantify the benefits of ride sharing. They analyzed the taxi trip data in New York City and concluded that ride sharing can reduce cumulative trip length by 40%. However, their model also constrained the sharing to be between two riders, ignoring the potential benefits from a more flexible system. Additionally, they assumed that the tolerance level for trip delay is identical for all riders, ignoring the individual heterogeneous tolerance and needs in the real world.

However, these models are mostly based on simplified system setups, not considering the real-time travel demands. For this matter, this study will incorporate real-time travel demand in a demand-driven ride sharing system and propose a reinforcement learning approach that applies deep Q network (DQN) for fast and effective solution. We consider real-time demand in a ride-sharing system. Further we propose a DQN-based approach for rebalancing ride sharing vehicles that has good performance in terms of both level of service and operational cost. We also explore a mathematical modeling approach with column generation to extend the applicability of the study to ride-pooling scenarios.

The reminder of the report is organized in the following. Section 2 presents the main methodology. Section 3 provides a case study. Section 4 discusses the extension to the ride-pooling system. Section 5 concludes the proposal.

2. Methodology

In order to operate a ride sharing system with EVs and car-pooling options, we build a ride sharing system to operate an EV fleet. For the convenience of readers, the parameters and variables are list in Table 1. In the system, as shown in Figure 1, we consider an EV sharing system with charging stations (labeled as $k \in [1, \dots, K]$) already distributed throughout a city with their position denoted as n_k. All EVs (labeled as $i \in [1, \dots, I]$) in the system are assumed to be identical with finite states of charge (SoC) levels L at the beginning of a discrete time horizon $t \in [0,1, \dots, T]$. If the SoC of an EV i in the fleet $l_{i,t}$ is below the minimum charging level s_t at time point t, the EV will be assigned to charge at the nearest charging station $k' \in [1, \dots, K]$. With known position of the EV i denoted as $p_{i,t}$, the charging state that the EV i will go to charge will be obtained as $k' = argmin_{k \in [1,\dots,K]}(norm(n_k, p_{i,t}))$, where $norm(n_k, p_{i,t})$ is the Euclid distance between the location n_k of the charging station k and the location $p_{i,t}$ of the EV i at time point t. We assume it starts to charge immediately with a constant charging speed q at the charging station when it arrives the charging station when the total number of charging EVs is less or equal to the capacity. Therefore, the change of SoC level of an EV i when it is charging will be $l_{i,t} = q * (t - t_0), l_{i,t} < L$, where t_0 is the time point when it starts to charge. Considering the electricity price e_t and the total number of charging EVs a_t at time point t, the cost of charging the EV fleet will be $e_t * a_t$, where to simplify the equation, we denote the electricity price e_t as the cost that an EV charges for one time interval. Since the SoC level of an EV will increase, the number of EVs in a SoC level will change as well.

Table 1 List of Parameters and Variables

parameters and variables	definition
t	time point, $t \in [0,1, \dots, T]$
k	label of a charging station
i	label of an EV, $i \in [1,2, \dots, I]$

m	travel demand, $m \in M$
m_0	origin of travel demand m, $m \in M$
m_1	destination of demand m, $m \in M$
m_t	request time of demand m, $m \in M$
$p_{i,t}$	location of EV i at time point t, $i \in [1,2,\dots,I], t \in [0,1,\dots,T]$
$l_{i,t}$	SoC of EV i at time point t, $i \in [1,2,\dots,I], t \in [0,1,\dots,T]$
s_t	minimum SoC at time point t, $t \in [0,1,\dots,T]$
n_k	location of charging station k
j	cell of the network
$d_{ij,t}$	distance between idle EV i and center of cell j at time point t, $i \in [1,2,\dots,I], t \in [0,1,\dots,T]$
$u_{j,t}$	number of demands minus number of idle EVs at cell j at time point t, , $t \in [0,1,\dots,T]$
$\overrightarrow{e_{ij,t}}$	unit vector from EV i to cell j at time point t, $i \in [1,2,\dots,I], t \in [0,1,\dots,T]$
$g_{,ij,t}$	gravity between EV i and cell j at time point t, $i \in [1,2,\dots,I], t \in [0,1,\dots,T]$
$g_{i,t}$	rebalance direction of EV i at time point t, $i \in [1,2,\dots,I], t \in [0,1,\dots,T]$
$\overline{g_t}$	maximum rebalance distance at time point t, $t \in [0,1,\dots,T]$
$\bar{v}$	maximum speed
c_t	parameter to control rebalance speed at time point t, $t \in [0,1,\dots,T]$
d_m	revenue of serving demand m by the EV fleet
M^s	set of demands served by the EV fleet
Q_t	total profit from time point 0 to time point t, $t \in [0,1,\dots,T]$
M_t^s	set of demand served by the fleet before time point t, $t \in [0,1,\dots,T]$
x_t	state of the system at time t (including distribution and SoC of EVs and demand), $t \in [0,1,\dots,T]$
a_t	action at time t (combination of s_t, $\mathrm{r_t}$, $\mathrm{c_t}$) , $t \in [0,1,\dots,T]$
λ_t	reward of action a_t at time point t, $t \in [0,1,\dots,T]$
γ	discount parameter
α_t	learning rate at time point t, $t \in [0,1,\dots,T]$

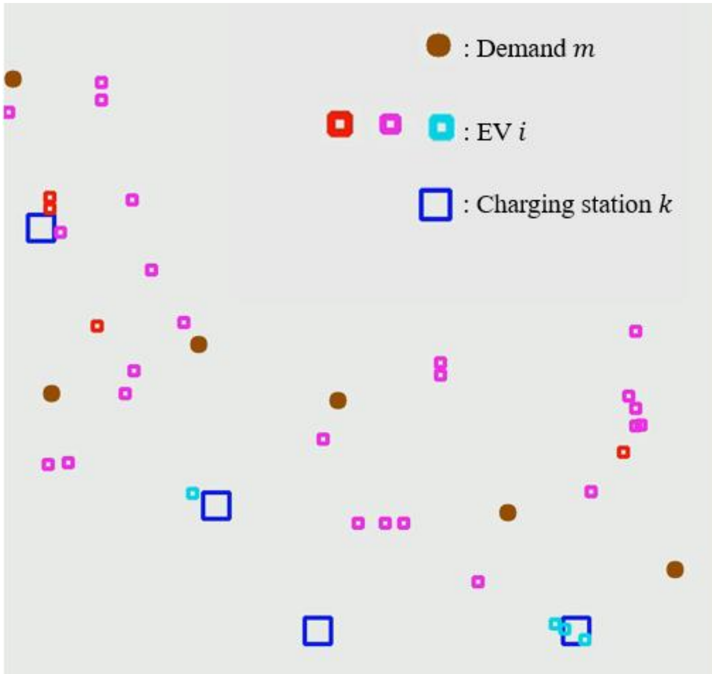

Figure 1 An Example of Ride Sharing System

The system aims to provide mobility service for customers from one location to another. We denote a demand as $m = (m_0, m_1, m_t)$, where m_0 is the origin of the demand and m_1 is the destination of the demand and m_t is the time of the request of the demand. To ensure a requested demand is served as quick as possible, which will maximize the performance of the system, we assume that a requested demand m will be immediately served by the neatest EV when the distance of the demand m and the nearest EV is within the assignment radius r_t at time point t.

In order to make sure that future travel demands (e.g. at time point $t + 1$) are serve with the most efficiency, we propose a gravity model to rebalance the available EVs at time point t. To calculate the gravity between demands and EVs, we divide the horizon into a number of cells denoted as $j \in J$. Considering a distribution of requested travel demands $u_{j,t}$ at time point t and an available EV i, the gravity of travel demands to the EV i is $g_{i,j,t} = u_{j,t}/(d_{i,j,t}^2)$. Therefore, the rebalance direction of the EV i is $g_{i,t} = \sum_{j \in J} g_{i,j,t} \overrightarrow{e_{\iota,J,t}}$, where $\overrightarrow{e_{\iota,J,t}}$ is the unit vector from EV i to cell j at time point t. To optimize the rebalance cost and speed, we add a decision variable c_t as the rebalance speed of EVs at time point t. With the rebalance speed c_t, the rebalance speed of an EV is calculated as $v_{i,t} = \min\left(\frac{|g_{i,t}|}{c_t * \overline{g_t}} * \bar{v}, \bar{v}\right)$, where $\overline{g_t}$ is the maximum rebalance distance at time point t and is calculated as $\overline{g_t} = \max_{i \in [1,2,\ldots,I]} |g_{i,t}|$, and $\bar{v}$ is the maximum speed of EVs.

With the rebalance speed, we are able to rebalance EVs to ensure that the number of served travel demands are maximized.
To obtain the optimal gross profit of the ride sharing system with EVs, we define an objective function of the gross profit by time point t as $Q_t = \max_{s_t, r_t, c_t, t \in [0,1,\ldots,t]} \sum_{m \in M^s} d_m - \sum_{t \in [0,1,\ldots,t]} e_t * a_t$, where d_m is the revenue of serving travel demand $m \in M^s$ and M^s is the set of demands served by the EV fleet. With the definition of the gross profit Q_t by time point t and constraints of operations of the system, we are able to build a non-linear optimization model and find an optimal solution with the known electricity price and future demand. However, to operate a ride sharing system in real time, the non-linear optimization is not efficient because of its computational time and known-travel-demand constraint. Therefore, we propose a deep Q-learning algorithm to train a Q-value network and estimate the Q-value of the system and obtain the optimal policy. To train the Q-value network, we define the transition of Q-value between time point t to time point $t + 1$ as
$Q_{t+1}(x_t, a_t) = Q_t(x_t, a_t) + \alpha_t\left[\lambda_t(x_t, a_t) + \gamma Q_t\left(x_{t+1}, argmax_a Q_t(x_{t+1}, a)\right) - Q_t(x_t, a_t)\right]$, (1)
where x_t is the state of the system at time t (including distribution and SoC of EVs and travel demands), a_t is the action at time t (value combination of $s_t, \mathrm{r_t}, \mathrm{c_t}$), λ_t is the reward of action a_t at time point t, γ is a discount parameter, $\alpha_t$is learning rate. To obtain the Q-value Q_t at time point t with a certain state of the system x_t and an action a_t, we simulate the ride sharing system with a big number of iterations (e.g. 10000 times), and train the Q-value network. In each iteration, we randomly choose an action a_t and operate the ride sharing system. Afterwards, the Q-value will be obtained with state x_t and a_t. When the iteration number is large enough, the estimated value of $Q_t(a_t, x_t)$ will be converged within ε, which we define as a criteria to stop the learning process. For the convenience of readers, the training process is shown in Figure 2.

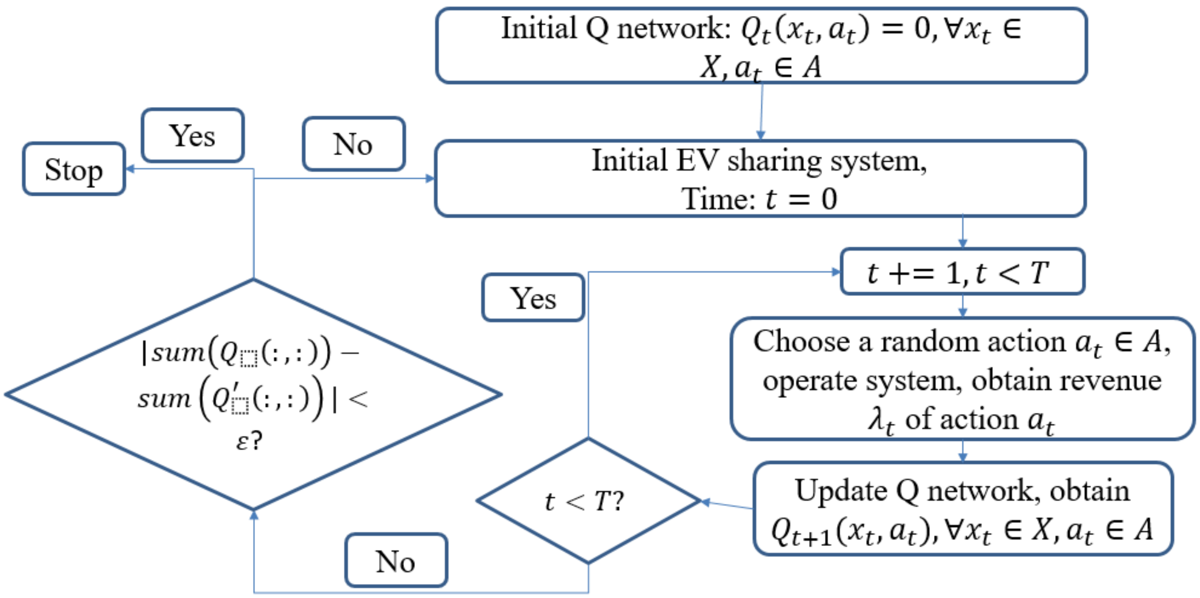

Figure 2 Training Process of the Q-value network

With a trained Q-value network, we are able to estimate the value of Q_t at time point t with a certain value of system state x_t and action a_t. Therefore, the optimal operation policy $\bar{a}_t$ of an EV fleet at time point t can be easily obtained with the equation (2):

$$\bar{a}_t = \max_{a_t} Q_t(x_t, a_t) \quad (2)$$

3. Case Study and Sensitivity Analysis

To test the performance of the proposed Q-learning algorithm in ride sharing systems, we apply the model in a case study Manhattan in New York City considering to serve the travel demands in real time. The distribution of the charging stations in Manhattan is obtained in https://www.nyserda.ny.gov/. We train the Q-value network using taxi data in weekdays in January 2016 and tested the model using the demand data in the first day of Feb, 2016. To model the charging speed and SoC capacity, we take the specifications parameters of Telsa Model S as an example. The fleet size is set to be 300 and the electricity price is obtained on the PJM website (https://www.pjm.com/). We compare the results of using Q learning algorithm and using traditional optimization model.

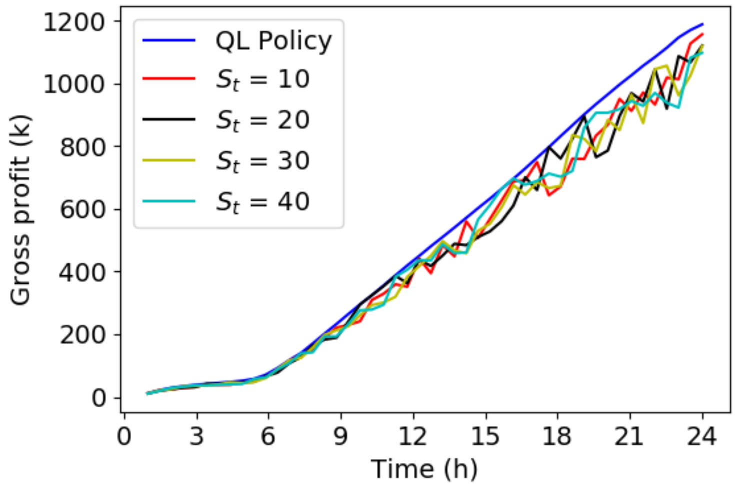

Figure 3 Comparison to Benchmark

In the benchmark, we test different minimum charging SoC of EVs as 10, 20, 30 or 40 kWh. The gross profit of the system is shown in Figure 3. In the figure, we see that with different settings of minimum charging SoC level, the optimal gross profit of the system obtained by applying optimization model will not change much. Since the optimal policies obtained by the optimization method are obtained with history demand and electricity price, the gross profit gained when the system is operated in real time is not optimal because of the stochasticity of travel demands and electricity price. When we apply the proposed Q-learning method, the gross profit is always higher than the gross profit obtained using the optimization methods. The result shows that the strategy is more efficient and better than solutions obtained by optimization models when the ride sharing system is running in real time.

4. Extension with High-capacity Ride-pooling Option

In order to offer high-capacity ride-pooling service, a powerful routing engine is required to find the optimal passenger-passenger and passenger-driver matching. In this section, we propose a column generation approach for solving the vehicle routing problem with high capacity based on the framework introduced by Alonso-Mora et al. (2017). In their framework for high-capacity ride-pooling service, they constructed an RTV-graph, which stands for Request-Trip-Vehicle graph, and formulated an Integer Linear Programming (ILP) to find the optimal routing and assignment between passengers and vehicles. However, they used heuristic methods to generate vehicle routes for picking up multiple passengers when vehicle capacity is high. We will propose an approach which can solve the high-capacity scenario optimally.

Firstly, we reformulate the RTV-graph to a bipartite graph with one column of nodes representing requests and the other column of nodes indicating trips with binding vehicles. Let R be the set of requests and T denote the set of trips with corresponding vehicles. The cost for each trip $\tau \in T$ is C_τ. Let C denote the capacity of vehicles and define $a_{r\tau} = 1$ if request r is in trip τ, 0 otherwise. The decision variable for the assignment ILP is y_τ = {0, 1}, where y_τ = 1 means trip τ will be selected in the optimal assignment. Then the assignment problem is:

$$\min_{y} \sum_{\tau \in T} C_\tau \cdot y_\tau \tag{1}$$

$$s.t. \sum_{\tau \in T} a_{r\tau} y_\tau = 1 \quad \forall r \in R \tag{2}$$

$$y_\tau \in \{0, 1\} \quad \forall \tau \in T \tag{3}$$

The objective of this ILP problem is to find the best trip-vehicle assignment and vehicle routing with the minimum cost. Constraints (2) ensure that each request only appears once in the optimal assignment. The maximum detour and maximum delay constraints are embedded when enumerating set of trips T.

Although problem (1) – (3) has a compact ILP formulation, the feasible trip list T is extremely large and T is exponentially increasing with respect to the number of requests $|R|$. The ILP (1) - (3) has a few constraints with an extremely large number of variables, which corresponding to feasible trips. Therefore, the problem is intractable with off-the-shelf ILP solvers. To address this issue and make the problem solvable for larger instances, we introduce the column generation approach. The idea behind column generation approach is to select a column (variable) at each iteration based on a pricing problem. The problem will be optimized only with variables selected by the pricing problem.

The LP-relaxation for ILP (1) – (3) is:

$$\min_{y} \sum_{\tau \in T} C_\tau \cdot y_\tau \tag{4}$$

$$s.t. \sum_{\tau \in T} a_{r\tau} y_\tau = 1 \quad \forall r \in R \tag{5}$$

$$y_\tau \geq 0 \quad \forall \tau \in T \tag{6}$$

The dual of problem (4) – (6) is:

$$\max_{\pi} \sum_{r \in R} \pi_r \tag{7}$$

$$s.t. \sum_{r \in R} a_{r\tau}\pi_r \leq C_\tau \quad \forall \tau \in T \tag{8}$$

Then the reduced cost for a trip τ is $\overline{C_\tau} = C_\tau - \sum_{r \in R} a_{r\tau}\pi_r$. To formulate the pricing problem, let N be the set of vehicle locations and passengers pickup locations. Let C_{ij}denote the cost for vehicle to travel from i to j and t_{ij} be the corresponding travel time,$i, j \in N$. Introduce x_{ij} to be a binary variable and $x_{ij} = 1$ if arc (i, j) is in trip τ. Let z_i be an integer variable indicating the vehicle seats left at location $i \in N$. Then, the reduced cost for trip τ is $\overline{C_\tau} = \sum_{i \in N} \sum_{j \in N} (C_{ij} - \pi_i) x_{ij}$.

The pricing problem is a resource constrained shortest path problem with limited capacity, maximum delay time, maximum waiting time and fixed origin (vehicle location). In this study, we relax the maximum delay and waiting time constraints to be the maximum travel time t^{max} constraint. And we assume all passengers have the same destination. Let v_0 be the vehicle location and v_d be the common destination location. The ILP formulation for the pricing problem is:

$$\min_{x} \quad \sum_{i \in N} \sum_{j \in N} (C_{ij} - \pi_i) x_{ij} \tag{9}$$

$$s.t. \quad \sum_{j \in R} x_{v_0 j} = \sum_{i \in R} x_{i v_d} = 1 \tag{10}$$

$$\sum_{i \in N} x_{ij} = \sum_{k \in N} x_{jk} \quad \forall j \in R \tag{11}$$

$$\sum_{i \in N \setminus \{v_0\}} \sum_{j \in N \setminus \{v_0\}} t_{ij} x_{ij} \leq t^{max} \tag{12}$$

$$C(1 - x_{ij}) + z_i \geq z_j + 1 \qquad \forall i \in S \cup \{v_0\}, j \in S \cup \{v_d\} \tag{13}$$

$$y_{v_0} = C \tag{14}$$

$$0 \leq z_i \leq C \quad \forall i \in N \tag{15}$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j \in N \tag{16}$$

Where objective (9) minimize the reduce cost for single vehicle trip τ. Constraints (10) and (11) ensure the trajectory and keep flow conservation constraint for the trip τ. Constraint (12) imposes the maximum travel time constraints for passengers in trip τ. Constraints (13) guarantee that if vehicle travel from i to j, the vehicle capacity is enough to pick up request at j. Also, the sub-tour elimination has been considered in these constraints. The range for seats left at each location is defined by Constraints (14) and (15). Constraints (16) enforce x_{ij} is binary variable.

Problem (9) - (16) is intractable for off-the-shelf solvers if the size of N is above 30. Therefore, we introduce augmented graph to approximate problem (9) - (16) and solve the constrained shortest path problem on this augmented graph to get the path with minimum reduced cost.

Before giving a formal definition for augmented graph, we describe how to generalize problem (9) - (16) as a constrained shortest path problem on graph. Let $G = (V, E)$ be a graph with vertices corresponding to locations $V = N$ and edges e_{ij} with edge cost $c_{ij} = C_{ij} - \pi_i$. The pricing problem can be interpreted as finding the shortest path in G from vertex v_0 to vertex $v_d$with both capacity budget and travel time budget. The idea for augmented graph is to embed the capacity and travel time budget information within graph.
Firstly, we will make an assumption to restrict the size of augmented graph.

Assumption: $\forall i, j \in V, t_{ij} \in \mathbb{Z}$, meaning that travel time in minute between any two vertices in graph is integer value.

Then we give the formal definition for augmented graph $\bar{G}$.

Definition (Augmented Graph): Given (G, C, t^{max}), the augmented graph $\bar{G}$ has vertex set $\bar{V} = \{< v, \eta, \delta > | v \in V, \eta = 0,1, \dots, C, \delta = 0,1, \dots, t^{max} \} \cup \{v_{\bar{d}}\}$, the edge set is $\bar{E} =$ $\left\{< v_i, \eta_i, \delta_i >< v_j, \eta_j, \delta_j > \middle| (v_i, v_j) \in E, \eta_j = \eta_i - 1, \delta_j = \delta_i - t_{v_i v_j}\right\} \cup$ $\{< v_d, \eta_i, \delta_i >, v_{\bar{d}} | \eta_i = 0,1, \dots, C, \delta_i = 0,1, \dots, t^{max}\}$. For the distance between vertices in graph, $f\left(< v_i, \eta_i, \delta_i >< v_j, \eta_j, \delta_j >\right) = c_{ij}$ and $f(< v_d, \eta_i, \delta_i >, v_{\bar{d}}) = 0$.

Below is an example for a graph and the corresponding augmented graph:

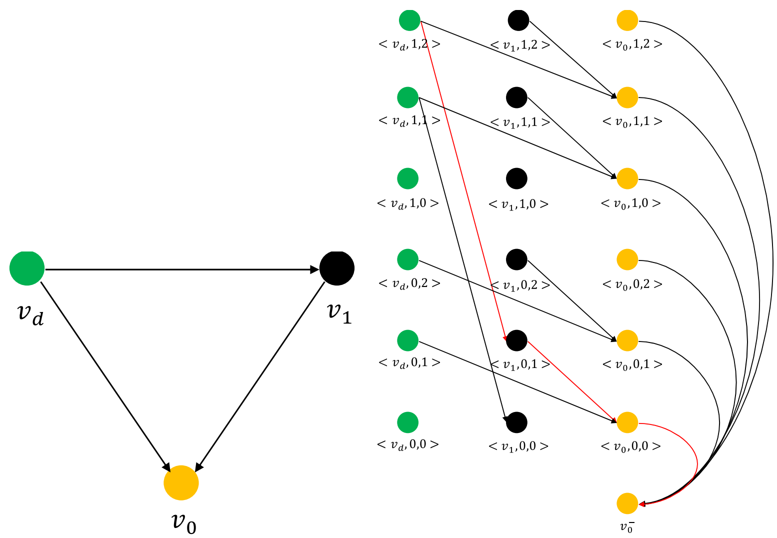

Figure 4. Example for augmented graph

After we construct the augmented graph $\bar{G}$, we are able to get the path with the minimum reduced cost in the pricing problem by running the shortest path algorithm start from $< v_0, C, t^{max} >$ to v_d^-.

This method will give an approximation solution to the LP-relaxation problem (4) – (6). To calculate the optimal solution for problem (1) – (3), we need to further combine our approach with branch-and-bound techniques, which remains as the future work.

5. Conclusion

This study applies a Q-learning network to obtain efficient operation policies for a fleet of EVs in a real-time ride sharing system. It incorporates real-time travel demand in a demand-driven ride sharing system and propose a reinforcement learning approach that for fast and effective solution. In the model, we train a Q network with history demand data and electricity price data and operate the system with real-time data with the trained network. Actions at each time point include serving travel demands, rebalance, charging in charging stations. A case study in NYC is conducted. We compared the performance of the solutions obtained by the proposed Q learning method and traditional optimization models. The results indicate that the proposed DQN-based approach for rebalancing ride sharing vehicles has better performance than the traditional models in terms of both level of service and operational cost when operating the ride sharing system in real time. In the end, we explore to ways of extending this problem to ride-pooling with high-capacity vehicles and we formulated a column generation model. In the future, we will continue to complete the extended ride-polling model and test its solution efficiency.

6. References

Shaheen, S., Cohen, A. and Mark, J., 2013. Innovative Mobility Carsharing Outlook: Carsharing Market Overview, Analysis, and Trends; Transportation Sustainability Research Center. University of California, Berkeley.

Greenblatt, J.B. and Shaheen, S., 2015. Automated vehicles, on-demand mobility, and environmental impacts. Current sustainable/renewable energy reports, 2(3), pp.74-81.

Morency, C., Verreault, H. and Demers, M., 2015. Identification of the minimum size of the shared-car fleet required to satisfy car-driving trips in Montreal. Transportation, 42(3), pp.435-447.

Machado, C.A.S., De Salles Hue, N.P.M., Berssaneti, F.T. and Quintanilha, J.A., 2018. An overview of shared mobility. Sustainability, 10(12), p.4342.

Agatz, N., Erera, A.L., Savelsbergh, M.W., Wang, X., 2011. Dynamic ride-sharing: A simulation study in metro atlanta. Procedia-Soc. Behav. Sci. 17, 532–550.

Barth, M., Todd, M., 1999. Simulation model performance analysis of a multiple station shared vehicle system. Transp. Res. Part C: Emerg. Technol. 7 (4), 237–259.

Caulfield, B., 2009. Estimating the environmental benefits of ride-sharing: A case study of dublin. Transp. Res. Part D: Transp. Environ. 14 (7), 527–531.

Dong, Y., Wang, S., Li, L., Zhang, Z., 2018. An empirical study on travel patterns of internet based ride-sharing. Transp. Res. Part C: Emerg. Technol. 86, 1–22. URL. <http:// www.sciencedirect.com/science/article/pii/S0968090X17302954>.

FHWA, 2011. Summary of Travel Trends: 2009 National Household Travel Survey, 82. URL: <http://nhts.ornl.gov>.

Fortune, 2016. Uber Now Has 40 Million Monthly Riders Worldwide — Fortune.com. <http://fortune.com/2016/10/20/uber-app-riders/> (accessed: 8/29/2017).

Galland, S., Knapen, L., Yasar, A.-U.-H., Gaud, N., Janssens, D., Lamotte, O., Koukam, A., Wets, G., 2014. Multi-agent simulation of individual mobility behavior in carpooling. Transp. Res. Part C: Emerg. Technol. 45, 83–98.

Gurumurthy, K.M., Kockelman, K.M., 2018. Analyzing the dynamic ride-sharing potential for shared autonomous vehicle fleets using cell phone data from orlando, florida. In: Transportation Research Board 97th Annual Meeting. TRB.

Hong, Z., Chen, Y., Mahmassani, H.S., Xu, S., 2017. Commuter ride-sharing using topology-based vehicle trajectory clustering: Methodology, application and impact evaluation. Transp. Res. Part C: Emerg. Technol. 85, 573–590.
Kleiner, A., Nebel, B., Ziparo, V.A., 2011. A mechanism for dynamic ride sharing based on parallel auctions. In: IJCAI. Vol. 11. pp. 266–272.
Krueger, R., Rashidi, T.H., Rose, J.M., 2016. Preferences for shared autonomous vehicles. Transp. Res. Part C: Emerg. Technol. 69, 343–355.
Levin, M.W., 2017. Congestion-aware system optimal route choice for shared autonomous vehicles. Transp. Res. Part C: Emerg. Technol. 82, 229–247.
Li, B., Krushinsky, D., Woensel, T.V., Reijers, H.A., 2016. The share-a-ride problem with stochastic travel times and stochastic delivery locations. Transp. Res. Part C: Emerg. Technol. 67, 95–108.
Lin, Y., Li, W., Qiu, F., Xu, H., 2012. Research on optimization of vehicle routing problem for ride-sharing taxi. Procedia-Soc. Behav. Sci. 43, 494–502.
Ma, S., Zheng, Y., Wolfson, O., 2015. Real-time city-scale taxi ridesharing. IEEE Trans. Knowledge Data Eng. 27 (7), 1782–1795.
Qian, X., Zhang, W., Ukkusuri, S.V., Yang, C., 2017. Optimal assignment and incentive design in the taxi group ride problem. Transp. Res. Part B: Methodol.
Santi, P., Resta, G., Szell, M., Sobolevsky, S., Strogatz, S.H., Ratti, C., 2014. Quantifying the benefits of vehicle pooling with shareability networks. Proc. Nat. Acad. Sci. 111 (37), 13290–13294.
Santos, D.O., Xavier, E.C., 2015. Taxi and ride sharing: A dynamic dial-a-ride problem with money as an incentive. Expert Syst. Appl. 42 (19), 6728–6737.
Teubner, T., Adam, M.T., Camacho, S., Hassanein, K., 2014. Understanding resource sharing in c2c platforms: the role of picture humanization. ACIS.
U.S. DoE, 2016. Inventory of U.S. Greenhouse Gas Emissions and Sinks: 1990 - 2014. Tech. Rep., U.S. Department of Energy.
Wadud, Z., MacKenzie, D., Leiby, P., 2016. Help or hindrance? the travel, energy and carbon impacts of highly automated vehicles. Transp. Res. Part A: Policy Practice 86, 1–18.
Alonso-Mora, Javier, Samitha Samaranayake, Alex Wallar, Emilio Frazzoli, and Daniela Rus. 2017. “On-Demand High-Capacity Ride-Sharing via Dynamic Trip-Vehicle Assignment.” Proceedings of the National Academy of Sciences 114 (3): 462. https://doi.org/10.1073/pnas.1611675114.