2017 IEEE 56th Annual Conference on Decision and Control (CDC)
December 12-15, 2017, Melbourne, Australia

Hierarchical Data-Driven Vehicle Dispatch and Ride-sharing

Ximing Chen, Fei Miao, George J. Pappas and Victor Preciado

***Abstract*—Modern transportation system suffers from increasing passenger demand, limited vehicle supply and inefficient mobility service. Towards building an intelligent transportation system to address these issues, we propose a hierarchical framework to implement strategies that is capable of allocating vehicles to serve passengers in different locations based on limited supply. In the higher hierarchy, we optimize idle mileage induced by rebalancing vehicles across regions using receding horizon control towards current and predicted future requests. In addition, we design a dispatch strategy that is robust against passenger demand and vehicle mobility pattern uncertainties. In the lower hierarchy, within each region, pick-up and drop-off schedules for real-time requests are obtained for each vehicle by solving mixed-integer linear programs (MILP). The objective of the MILP is to minimize total mileage delay due to ride-sharing while serving as many requests as possible. We illustrate the validity of our framework via numerical simulations on taxi trip data from New York City.**

I. INTRODUCTION

Modern transportation systems face enormous number of challenges such as air pollution caused by vehicles [1] and traffic congestions [2] due to the steady growth of urban population. As a consequence, there is an urgent need to build intelligent transportations systems. Towards this goal, researchers have proposed two major strategies recently: (i) *vehicle dispatch*, which is a framework to allocate vehicles to different areas in the city, and (ii) *ride-sharing*, which is a framework to assign a single vehicle to serve multiple requests.

In order to design vehicle dispatch strategies, it is crucial to construct mobility patterns of passengers [3, 4] based on large-scale data recording passenger locations information. Subsequently, vehicles allocation strategies are designed based on request-models [5–11]. In [12], Wong et al. provided heuristics based on spatial-temporal distribution of requests from passengers to improve the utilization rate of urban fleets. In [6], Pavone et al. proposed a mobility-on-demand system and obtained an optimal strategy to minimize the number of rebalanced vehicles. In [9], Miao et al. designed a taxi-dispatch strategy based on Receding Horizon Control (RHC) [13] framework using real-time requests as well as predicted future demand. Later, the authors in [10] further proposed a dispatch strategy that is robust against demand uncertainties.

In addition to vehicle dispatching, ride-sharing is also a viable solution to traffic congestion problems [14]. The primal objective of ride-sharing problem is to minimizes customer's waiting time [15], or to reduce travelling mileage [16]. Nonetheless, this problem is closely related to vehicle-routing problems [17, 18], whose solution is computationally demanding; thus, it is unsatisfactory to be implemented in real-time. To tailor this issue, in [19], the authors proposed a greedy approach to match the vehicles and requests, whereas in [20], the authors innovated a framework allowing ride-share across multiple requests. Nonetheless, these proposals are based on heuristics, which do not have quality guarantees.

The advantages of applying vehicle dispatching and ride-sharing strategies to transportation systems are that the average waiting time of passengers can be reduced and the utilization rate of vehicles are increased. However, on one hand, existing system-level vehicle dispatch strategies [6, 9] ignore detailed implementation on the pairwise assignment between requests and vehicles, which may cause inefficiency when the planning period is long. On the other hand, ride-sharing strategies has fundamental limitations on scalability as its underlying problem can be reformulated as the travelling salesman problem, which has *NP-complete* complexity.

In this work, we aim to establish a framework taking advantages of vehicle dispatch and ride-sharing formulations simultaneously, while countering the mentioned disadvantages. To achieve these goals, we notice that ride-sharing occurs frequently when the origins and/or destinations of two groups of passengers are close in physical distances or their trips are similar, i.e., serving an additional request will not deviate from the route of serving only one of them too much. This observation motivates us to consider a local and regional formulation on the ride-sharing problem. More specifically, we propose a hierarchical framework based on partitioning the entire city into regions. On one hand, in the higher hierarchy, we dispatch vehicles every hour or 30 minutes. On the other hand, in the lower hierarchy, within each region, we design vehicle pick-up and drop-off schedules to serve real-time requests for a short time horizon (e.g., every 5 minutes).

The contributions of this work are three-fold. Firstly, we establish a hierarchical framework, where in the system-level we dispatch vehicles to regions of the city and minimize the ideal travelling mileage of relocating vehicles, and within each region we assign vehicles to serve an ordered sequence of ride-sharing requests, by solving a mixed-integer linear program (MILP). Secondly, we consider uncertainties in passengers' demand and vehicles' supply, and prove a computationally tractable convex reformulation that attains the optimal solution under the worst case scenario induced by uncertainties in the higher hierarchy. Finally, we explore

Ximing Chen, George J. Pappas and Victor Preciado are with the Department of Electrical and Systems Engineering at the University of Pennsylvania, Philadelphia PA 19104. Email:{xmingch, pappasg, preciado}@seas.upenn.edu. Fei Miao is with the University of Connecticut, Storrs CT 06268. Email: fei.miao@uconn.edu.

978-1-5090-2873-3/17/$31.00 ©2017 IEEE

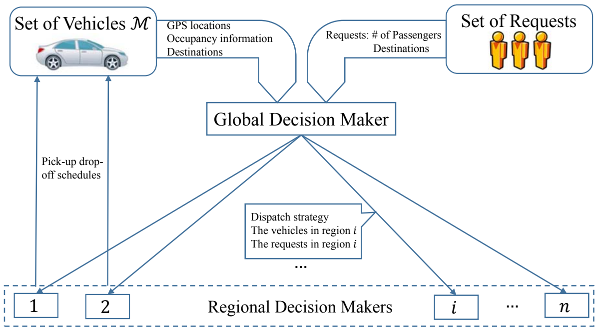


Figure 1: The infrastructure of the hierarchical vehicle-dispatch and ride-share mechanism.

our proposed solutions to a real data-set containing taxi-trips information from New York City and demonstrate that the vehicle occupancy rate is increased while total idle-travelling mileage is reduced with the help of our hierarchical model.

The rest of the paper is organized as follows. In Section II, we formulate our global vehicle dispatch model. In Section III, the regional vehicle ride-sharing assignment problem is introduced. In Section IV, we illustrate our results using the NYC taxi-trip data-set. Finally, conclusions and discussion of future research are presented in Section V.

II. Centralized Vehicle Dispatch Model

In our framework, we maintain two types of decision makers. In the higher hierarchy, a global and centralized decision maker periodically (e.g., every hour or 30 minutes) collects and stores real-time information from all regions of the city, including the GPS locations, occupancy status of all vehicles (i.e., the number of available seats in each of them) and all appeared requests from passengers. We utilize these information to estimate the future customer requests. Then, the global decision maker implements a dispatch strategy and assigns certain amount of vacant vehicles to different regions of the city according to service requirements. In the lower hierarchy, each region has a unique localized decision maker that receives real-time vehicle information and customer requests inside the region, and the vehicle dispatch result from the global decision maker. Based on these information, in a shorter time period (e.g., every 5 minutes), the regional decision maker assigns vehicles within the region to serve ride-share requests. The infrastructure of our hierarchical ride-share mechanism is depicted in Fig. 1. In the next two sections, we introduce the models of the global and local decision makers in details, respectively.

Hereafter, we introduce the decision process of the global decision maker. As a first step, we partition the city into n disjoint regions, indexed as[1] $\mathcal{N} = [1, N]$. Moreover, we label all vehicles in the entire city using a set of integers $\mathcal{M} = [1, m]$, where m denotes the total number of vehicles. The maximum number of available seats in a vehicle $v \in \mathcal{M}$ is referred to as the *capacity* of v, denoted by $C_{\max} > 0$. Then, we discretize time into time steps. For instance, with 30 minutes as a discretization period, we associate the current time and 30 minutes from now with instances t and $t+1$, respectively. At the beginning of time t, the global decision maker collect real-time information on the available vehicles and predict the number of requests that may arise in all regions in the future time instances $[t+1, t+T-1]$, where T is the maximum prediction horizon. The global decision maker implements the decision at time t obtained from solving the vehicle dispatch problem. When the time rolls forward to $t+1$, the global decision maker collects new information on vehicle locations and requests, and a new dispatch solution is obtained by solving the same optimization, with possibly different predicted requests.

In what follows, we introduce the decision variables, objective and constraints of the vehicle dispatch problem. Let $X^k_{ij,c}$ be the number of vehicles with c available seats heading towards region j from region i at $k \in [t, t+T-1]$. In the vehicle dispatch problem, the global decision maker decides the amount of vacant vehicles that should be dispatched from source region i to destination region j at each time instance t, denoted by $X_{ij,C_{\max}}$. We require $X^k_{ij,C_{\max}} \geq 0$ for all $i, j \in \mathcal{N}$, and $k \in [t, t+T-1]$. After defining the decision variables, we begin to formulate the objective and constraints in the vehicle dispatch problem.

Idle travelling mileage: We define the *idle-driving mileage* as the total distance travelled by a vehicle without any passenger on it. When relocating a vacant vehicle from one region to another, the driver receives no revenue and pay a certain amount of fee (e.g., gasoline fee). Thus, it is of interest to minimize their idle driving mileage. Specifically, we approximate the idle travelling mileage from region i to j by a known distance $W_{ij} > 0$. The values W_{ij} depend on how we partition the entire city. For example, if we consider partitioning a city into regions grouped by 4 blocks, then W_{ij} can be approximated by the city-block distance between the centers of region i and j. Subsequently, the total idle travelling mileage of all dispatched vehicles is characterized by:

$$\sum_{k=t}^{t+T-1} \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}} X^k_{ij,C_{\max}} W_{ij}. \tag{1}$$

Limited travelling mileage: Given a fixed discretization period, if a vehicle is dispatched from region i to region j, we must guarantee that it is able to arrive at region j before the time horizon rolls forward to $t+1$. Therefore, we do allow dispatch decisions to distant regions due to speed limitations or congestion conditions. This constraint is captured by:

$$X^k_{ij,C_{\max}} W_{ij} \leq D_{\max} X^k_{ij,C_{\max}}, \forall i, j, k, \tag{2}$$

where $D_{\max}$ is the maximum distance allowed for dispatching and it is known a priori to the global decision maker.

Service fairness: We define the total number of requests in region i at time instance $k \in [t, t+T-1]$ as r^k_i, and $\mathbf{r}^k \in \mathbb{R}^n$ is constructed by stacking r^k_i. Using prior knowledge

[1]We use $[A, B]$ to denote the set of integers $\{A, A+1, \dots, B\}$ where $A, B \in \mathbb{N}$.

and past data on passenger requests, we characterize $\mathbf{r}^k$ by an uncertain vector supported on a compact and convex set. More specifically, let Δ^k be the set of uncertain request vectors, and $\mathbf{r}^k \in \Delta_k$. In this paper, we consider the set of uncertain requests characterized by polytopes $\Delta_k = \{\mathbf{r}^k : A^k\mathbf{r}^k \leq \mathbf{b}^k\}$ – see [10].

Let $L^k_{i,c} \in \mathbb{R}_+$ be the number of vehicles with c available seats at region $i \in \mathcal{N}$ at time instance k before executing dispatch decision. In particular, $\mathbf{L}^k_c \in \mathbb{R}^N_+$ is obtained by stacking $L^k_{i,c}$, and such information is provided by the global decision maker at time instance k. At each time instance, the number of vacant vehicles with $C_{\max}$ available seats relocated to other regions from i is $\sum_{j=1}^n X_{ij,C_{\max}}$. Hence, the total number of vehicles with c available seats is:

$$\gamma^k_{i,c} \equiv \begin{cases} \sum_{j=1}^n X^k_{ji,c} - \sum_{j=1}^n X_{ij,c} + L^k_{i,c}, \text{ if } c = C_{\max}, \\ L^k_{i,c}, \text{ otherwise.} \end{cases} \tag{3}$$

Consequently, the total number of available seats in all vehicles in region i at time instance k equals to:

$$s^k_i \equiv \sum_{c=1}^{C_{\max}} c\gamma^k_{i,c} \geq 0 \tag{4}$$

Combing above, we define the demand supply ratio by: $\frac{r^k_i}{s^k_i}$. We aim to design a dispatching strategy such that the total number of requests (i.e., demand) and total number of available seats in vehicles to serve these requests (i.e., supply) are balanced. Nonetheless, balanced dispatching strategies are often unachievable. For example, after a football game, the number of requests may be excessive while the number of available vehicles may be limited. Therefore, instead of enforcing perfect balance between supply and demand, we impose constraints on the ratio between supply and demand and allow the ratio to vary within certain range. More specifically, we restrict the demand supply ratios in range of variation specified by $[\underline{\theta}, \overline{\theta}]$, where $\underline{\theta}$ and $\overline{\theta}$ capture the scenarios when there are extra and shortage in supply, respectively. In particular, the values of these two parameters should satisfy $0 < \underline{\theta} < 1 < \overline{\theta}$. Hence, the constraint on service fairness is described by:

$$\underline{\theta} s^k_i \leq r^k_i \leq \overline{\theta} s^k_i, \forall i, k. \tag{5}$$

Dynamics of vacant vehicles: Now, we characterize the mobility model of vehicles. We notice that the state of a vehicle may encounter either a location transition, i.e., travelling from region i to region j, or a capacity transition i.e., the number of its available seats changes from c_k to c_{k+1}, from t to $t+1$. In other words, $L^{k+1}_{i,c}$ is affected by the number of vehicles that arrives at region i from other regions, and vehicles that have c available seats, after possibly several drop-off or pick-up actions. However, these two types of transitions are determined by the mobility pattern of passengers, which is difficult to obtain explicitly and deterministically. Instead, we denote $P_{ij,ls}$ as the probability of a vehicle travels from region i to region j while its seat availability changes from l to s, which can be estimated by utilizing past vehicle trip data [9]. Therefore, the dynamics of $L^k_{i,c}$ can be characterized by:

$$L^{k+1}_{p,s} = \sum_{l=0}^{C_{\max}} \sum_{i \in \mathcal{N}} \gamma^k_{i,l} P_{ip,ls}. \tag{6}$$

Nonetheless, estimations on $P_{ij,ls}$ may induce uncertainties. For example, a confidence interval of $P_{ij,ls}$ can be captured by $P_{ip,ls} \in \left[\underline{P}_{ip,ls}, \overline{P}_{ip,ls}\right]$. We model the uncertainties of the values $P_{ij,ls}$ as a polytope. In other words, we let the uncertainty set of transition probabilites be $\Delta_P = \left\{\mathcal{P} \in \mathbb{R}^{n(C_{\max}+1)\times n(C_{\max}+1)} : A_{\mathcal{P}} \operatorname{vec}(\mathcal{P}) \leq \mathbf{b}_{\mathcal{P}}\right\}$, where $\operatorname{vec}(\mathcal{P})$ is a vectorization of $P_{ip,ls}$.

Vehicle dispatch problem formulation: The global decision maker aims to obtain a vacant vehicles dispatch decision while minimizing their idle travelling mileage. Additionally, the strategy is capable of planning ahead of time by taking future demand into account. However, due to the uncertainty in prediction process, dispatch strategy can perform poorly. Therefore, it is necessary to consider a dispatch decision that performs reasonably well even in the worst case scenario of predicted demand. Consequently, we formulate our vehicle dispatch problem from a robust optimization point of view.

$$\begin{aligned} \underset{X^k_{ij,l}, \mathbf{L}^k_c}{\text{min.}} \quad & \underset{\mathbf{r}^k \in \Delta_k, \mathbf{P} \in \Delta_P}{\text{max.}} \quad (1) \\ \text{s.t.} \quad & (2), (3), (4), (5), (6), \\ & X^k_{ij,l} \geq 0, \quad i, j \in \mathcal{N}. \end{aligned} \tag{7}$$

Theorem 1. *Given* $\Delta_k = \{\mathbf{r}^k : A^k\mathbf{r}^k \leq \mathbf{b}^k\}$ *and* $\Delta_P = \left\{\mathcal{P} \in \mathbb{R}^{n(C_{\max}+1)\times n(C_{\max}+1)} : A_{\mathcal{P}} \operatorname{vec}(\mathcal{P}) \leq \mathbf{b}_{\mathcal{P}}\right\}$, *Problem* (7) *is equivalent to a linear program.*

Due to space limitations, we provide a sketch of prove idea instead. Notice that in (7) the objectives constraints are linear while the uncertain sets are described by polytopes Δ_k and Δ_P; thus, one can rewrite the maximization problem in (7) into a minimization problem using Lagrangian duality. Strong duality holds due to convexity and non-empty interior of the feasible regions [21].

III. Ride-sharing Model

The global decision maker dispatches vacant vehicles to different regions across the city. A regional decision maker is responsible for scheduling detailed pick-up and drop-off procedures for all vehicles with available seats within the region. In order to utilize the available seats in vehicles, we design optimal pick-up and drop-off schedules for vehicles that are within the same region for ride-sharing. We refer to this problem as *regional ride-sharing problem*.

In the regional level, we adopt a finer discretization of time. As illustrated in Figure 2, We divide every time period considered in the global level into Ψ equal pieces such that t' and $t'+\Psi$ corresponds to t and $t+1$ in the higher hierarchy. At each time instance t', the local decision maker solves an optimization program to obtain ride-sharing strategies.

In addition, within each time period $[t', t'+1]$, we define $\kappa \in [1, \mathcal{T}]$ as an index representing a pick-up (resp. drop-off) order of requests (resp. passengers). For instance, if requests w_1 and w_2 are scheduled to be served at κ and $\kappa+1$ by the same vehicle, then it means that the vehicle will pick up w_1 prior to w_2. At time instance t', the regional decision makers updates regional the vehicle locations, destination information and requests within i. Then, based on these information, the regional decision maker provides an pick-up and drop-off schedule to all the vehicles within this region by solving the regional ride-sharing problem. The vehicles will serve requests and drop-off passengers during $[\tau, \tau+\mathcal{T}-1]$ according to the schedule. Next, we describe the information available to regional decision maker in more details.

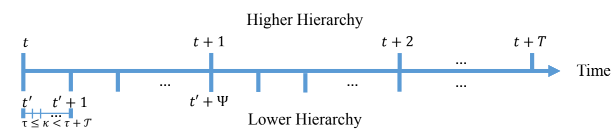


Figure 2: Illustration of the time instances difference between higher and lower hierarchy.

Region-specific information: Let $\mathcal{V}_i \subseteq \mathcal{M}$ be the set of vehicles in region i. We assume that the vehicles in any region will remain in the same region during $[\tau, \tau+\mathcal{T}-1]$. The destinations of vehicles are provided to the regional decision maker. Specifically, let $\mathcal{S}_v^\kappa$ be the set of pairs $\mathbf{p} \equiv (q, d)$ containing the service information in vehicle v at time κ, where q is the number of passengers and d is their corresponding destinations. In what follows, we use $\mathbf{p}(q)$ and $\mathbf{p}(d)$ to refer these two quantities, respectively. Subsequently, we extract the destination information from the pairs in $\mathcal{S}_v^\kappa$ to form a set of destinations $\mathcal{D}_v^\kappa$. The set $\mathcal{D}_v$ may contain multiple elements if the passengers are travelling to different locations. Furthermore, if a vacant vehicle is assigned with a relocating decision from the global decision maker, then we also include its destination into $\mathcal{D}_v$. Let $\mathcal{W}_i$ be the set of requests in region i. Each request w in $\mathcal{W}_i$ contains following elements: (i) the number of passengers q_w, and (ii) the two-dimensional GPS location of their destination $d_w \equiv [d_{w,x}, d_{w,y}]^\top$.

Based on $\mathcal{W}_i$, the regional decision makers assign vehicles in $\mathcal{V}_i$ to serve a sequence of requests. We let y_{vw}^κ be an indicator variable, where $y_{vw}^\kappa = 1$ if vehicle v is assigned to pick up request w at time instance κ, and 0 otherwise. We let $z_{v\mathbf{p}}^\kappa$ be another indicator variable representing whether vehicle v should drop-off passenger $\mathbf{p} \in \mathcal{S}_v^\kappa$ at time instance κ. In particular, passenger $\mathbf{p}$ will be dropped off provided that $\mathbf{p}(d)$ is contained in the current region of v, i.e., $\mathbf{p}(d) \in i$. Next, we introduce the objective and constraints in the ride-sharing scheduling problem.

Regional mileage delay: In general, ride-sharing scheme may cause delay. More On one hand, if a non-vacant vehicle v is assigned to serve requests, it may cause extra travel time for passengers on the vehicle v. Consequently, it is crucial to design optimal pick-up and drop-off order to minimize the total pick-up and drop-off delay caused by ride-sharing. Nonetheless, finding such an optimal order is a version of the Travelling Salesman Problem, which has *NP-complete* [22] complexity. Thus, we seek an alternative representation of mileage delay. We assume that the distance from the pick-up location of request w (resp., drop-off location of $\mathbf{p}$) to the center of region i, denoted as l_w (resp., $l_\mathbf{p}$) is available to regional decision maker initially. Therefore, the total regional mileage delay can be approximated as:

$$\sum_{\kappa=1}^{\mathcal{T}} \sum_{v \in \mathcal{V}_i} \left[\sum_{w \in \mathcal{W}_i} l_w y_{vw}^\kappa + \sum_{\mathbf{p} \in S_v^\kappa, \mathbf{p}(d) \in i} l_\mathbf{p} z_{v\mathbf{p}}^\kappa \right]. \tag{8}$$

Note that the proxy we proposed for modelling the mileage delay is an upper bound on the true regional mileage delay, hence, as we reduce this objective, the true regional idle travelling mileage will also be reduced. Within $[\tau, \tau+\mathcal{T}-1]$, the total number of requests served by $v \in \mathcal{V}_i$ can be represented by:

$$\sum_{\kappa=1}^{\mathcal{T}} \sum_{w \in \mathcal{W}_i} \sum_{v \in \mathcal{V}_i} y_{vw}^\kappa. \tag{9}$$

There is a trade-off between the mileage delay and the number of served passengers, i.e., the more served requests imply larger mileage delay. Consequently, we seek to find a strategy that schedules vehicle to serve as many request as possible without causing extensive mileage delay. To achieve this goal, we propose to consider a weighted-linear combination of Eq.(8) and Eq. (9), with weight $\lambda > 0$:

$$\begin{aligned} &\sum_{\kappa=1}^{\mathcal{T}} \sum_{v \in \mathcal{V}_i} \left[\sum_{w \in \mathcal{W}_i} l_w y_{vw}^\kappa + \sum_{\mathbf{p} \in S_v^\kappa, \mathbf{p}(d) \in i} l_\mathbf{p} z_{v\mathbf{p}}^\kappa \right] \\ &- \lambda \left(\sum_{\kappa=1}^{\mathcal{T}} \sum_{w \in \mathcal{W}_i} \sum_{v \in \mathcal{V}_i} y_{vw}^\kappa \right). \end{aligned} \tag{10}$$

Constraint on pick-up and drop-off procedure: According to our definition on the variable y_{vw}^κ, we require that every vehicle v is allowed to serve at most one request at any time instance $\kappa \in [1, \mathcal{T}]$, i.e.,

$$\sum_{w} y_{vw}^\kappa \leq 1, \text{ for all } v, \kappa. \tag{11}$$

Moreover, every request can only be served by at most one vehicle. Furthermore, if a vehicle is assigned to serve a particular request, there is a unique associative time instance at which the request will be served. Thus, we require:

$$\sum_{\kappa=1}^{\mathcal{T}} \sum_{v} y_{vw}^\kappa \leq 1, \text{ for all } w. \tag{12}$$

Similarly, we impose the following constraint on dropped-off procedure:

$$\sum_{\kappa=1}^{\mathcal{T}} \sum_{v} z_{v\mathbf{p}}^\kappa = 1, \text{ for all } \mathbf{p}(d) = i. \tag{13}$$

Constraint on vehicle seat availabilities: We require that the available seats on every vehicle is non-negative during pick-up and drop-off assignments, and the total number of passengers in any vehicle should not exceed the maximum capacity $C_{\max}$. In particular, when a vehicle v only has $c_v < C_{\max}$ available seats at time τ, it should not be assigned to serve requests w with passenger number $q_w > c_v$. To model this scenario, we define c_v^κ as the number of empty seats of vehicle v at time instance κ, where c_v^τ is provided initially. At instances $\kappa > \tau$, the number of empty seats are determined by the previous pick-up and drop-off behaviours, which results in the following constraint:

$$c_v^{\kappa+1} = c_v^\kappa + \sum_w q_w y_{vw}^\kappa - \sum_{\mathbf{p}\in\mathcal{S}_v^\kappa, \mathbf{p}(d)=i} \mathbf{p}(q) z_{v\mathbf{p}}^\kappa \geq 0, \quad (14)$$
$$c_v^\kappa \leq C_{\max}, \text{ for all } v \in \mathcal{V}_i, \kappa \in [1, \mathcal{T}].$$

Trip similarities: Finally, when assigning non-vacant vehicles to serve new requests, we aim to serve new requests that have similar destinations as that of passengers on the corresponding vehicles. Because it is time-consuming and undesired to serve requests whose destinations are completely opposite of those of passengers on the vehicles. To avoid this undesired scenario, we measure trip similarities when making local decisions. Therefore, we set $y_{vw}^\kappa = 0$, if the maximum distance between destinations in $\mathcal{D}_v^\kappa$ and request is larger than a certain threshold D_{loc}, i.e., if $\max_{d\in\mathcal{D}_v^\kappa}(|d_x - d_{w,x}| + |d_y - d_{w,y}|) > D_{loc}$. This constraint is equivalent to:

$$y_{vw}^\kappa \max_{d\in\mathcal{D}_v^\kappa}(|d_x - d_{w,x}| + |d_y - d_{w,y}|) \leq D_{loc} y_{vw}^\kappa. \quad (15)$$

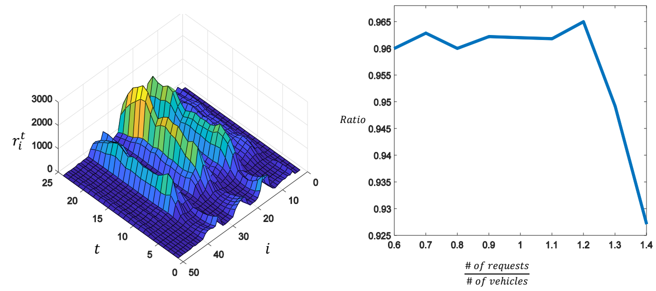
Formulation of optimal ride-sharing problem: For each region, our target is to schedule a pick-up assignment and drop-off process for every vehicle in the region by solving a mixed integer linear program (MILP). In particular, the objective of the program is to minimize the total mileage delay caused by ride-sharing scheme and the number of left requests. The constraints characterize pick-up and drop-off schemes, limitation of seats in vehicles, and trip similarities. The optimal ride-sharing problem is defined as follows:

$$\min_{y_{vw}^\kappa, z_{v\mathbf{p}}^\kappa} \ (10) \quad \text{s.t.} \ (11), (12), (13), (14), (15). \quad (16)$$

After the assignment decisions are made and executed, each regional decision maker collects information on the number of available seats within each vehicle and update it to the global decision maker before the time instance $t' + \Psi$. Based on these new information, the global decision maker can re-categorize the vehicles according to their seat availabilities and perform another dispatch decision.

IV. Simulations

In this section, we demonstrate the performance of our vehicle dispatch and ride-share framework using taxi trip data in New York City [23]. Our data set consists of following information: (i) the pick-up and drop-off locations of each request specified in GPS coordinators, (ii) the pick-up/drop-off time and date of each request.



(a) (b)

Figure 3: In figure (a), i, t and r_i^t represent the index for region, time and the average total number of requests within a hours in each region i. In addition, t ranges from 0 AM to 0 AM (the next day) and it is indexed as 1 to 24. In figure (b), we plot the ratio between the number of served requests and total number of requests versus the demand supply ratio.

As a first step, we partition Manhattan into $n = 50$ regions. In total, there are 15000 - 17000 vehicles travelling in the city area. We set the discretization period as 30 minutes and let $T = 3$. In other words, we rebalance vehicles inside Manhattan area every 30 minutes. Moreover, we predict the requests that may emerge over the window of next 90 minutes in the each region. After reaching the dispatched regions, drivers pick up the passengers according to the ride-sharing mechanism. Finally, we let the capacity of vehicles to be 4, i.e., $C_{\max} = 4$.

The vehicle dispatch problem is implemented using CVX [24] in MATLAB. By solving the problem, we observe that rebalancing happens with increase of requests. More specifically, the demand for taxis keeps increasing from 5 PM onwards and peaks at around 8 PM on weekdays. As illustrated by Fig. 3-(a), the majority of requests are concentrated on a few regions corresponding to down-town Manhattan. To compensate the potential lack of vehicles, the global decision maker will start rebalancing vacant vehicles from up-town area to down-town Manhattan in advance.

In our lower hierarchy implementation, we use the `intlinprog` solver in Matlab to solve our ride-sharing problem. The parameters of the ride-sharing problem are set as follows: (i) we set $\Psi = 6$, i.e., ride-sharing decisions are updated every 6 minutes, and (ii) set $\mathcal{T} = 4$ since a vehicle can pick-up at most 4 passengers, as limited by $C_{\max}$. At time t, we also obtain the total number of vehicles in region i, denoted as $|\mathcal{V}_i|$. Finally, the request information are directly retrieved from the data set.

To assess the benefit brought by ride-sharing, we propose to compute the utility rate of vehicles, which is defined as the requests served per vehicle. In addition, we focus on designing ride-sharing strategies in region $i = 40$ at 8 PM. Nonetheless, the number of passengers information is not contained in the provided data set. To address this issue, we randomly generate the number passengers from a range of 1 to 4 for each request. From our experiment, we observe that our ride-sharing scheme allows vehicles to serve 2.18 extra passengers per vehicle on average. Thus, to serve the same amount of requests, ride-sharing requires less amount of vehicles comparing with ordinary pick-up algorithms. Furthermore, we explore the fraction of requests served by vehicles through ride-sharing at different time of a day in region i. As illustrated in Fig. 3-(b), the ride-sharing scheme allows at least 90% of requests to be served at any time. In particular, when we encounters supply shortage, i.e., ratio between demand and supply equals to 1.4, ride-sharing ensures 92% of the requests to be served. When each vehicle serves at most one request, only 71.43% of the requests can be served.

Finally, there are more than 10,000 vehicles traversing tin the city area, it is computationally challenging to solve ride-sharing problem in this scale. However, as we have decomposed the problem into two levels, the number of vehicles are limited to around 100, i.e., the problem scale is reduced drastically. In fact, it takes around 15s to solve the MILP with $|\mathcal{V}_i| = 100$, and $|\mathcal{W}_i| = 500$.[2]

V. CONCLUSIONS

In this paper, we considered the problem of mitigating traffic congestion by building an intelligent transportation system. To tackle the issues of scalability and lack of optimality guarantees in previous work, we proposed a hierarchical framework to implement vehicle dispatch and ride-share strategies simultaneously based on region partitioning. In the higher hierarchy, we predicted future requests on vehicles, and obtained an optimal dispatch strategy by solving an optimization problem using receding horizon control approach. In addition, the obtained strategy is robust against uncertainties on both requests and vehicle mobility patterns. In the lower hierarchy, region-specific pick-up and drop-off schedules were obtained by solving a mixed-integer linear programs (MILP). We illustrated the performance of our framework via numerical simulations on a data-set containing taxi trip data from New York City.

REFERENCES

[1] P. Pant and R. M. Harrison, "Estimation of the contribution of road traffic emissions to particulate matter concentrations from field measurements: a review," *Atmospheric Environment*, vol. 77, pp. 78–97, 2013.

[2] D. A. Hennessy and D. L. Wiesenthal, "Traffic congestion, driver stress, and driver aggression," *Aggressive behavior*, vol. 25, no. 6, pp. 409–423, 1999.

[3] L. Moreira-Matias, J. Gama, M. Ferreira, J. Mendes-Moreira, and L. Damas, "Predicting taxi–passenger demand using streaming data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1393–1402, 2013.

[4] X. Li, G. Pan, Z. Wu, G. Qi, S. Li, D. Zhang, W. Zhang, and Z. Wang, "Prediction of urban human mobility using large-scale taxi traces and its applications," *Frontiers of Computer Science*, vol. 6, no. 1, pp. 111–121, 2012.

[5] A. Pantoja and N. Quijano, "A population dynamics approach for the dispatch of distributed generators," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 10, pp. 4559–4567, 2011.

[6] M. Pavone, S. L. Smith, E. Frazzoli, and D. Rus, "Robotic load balancing for mobility-on-demand systems," *The International Journal of Robotics Research*, vol. 31, no. 7, pp. 839–854, 2012.

[7] R. Zhang and M. Pavone, "Control of robotic mobility-on-demand systems: a queueing-theoretical perspective," *The International Journal of Robotics Research*, vol. 35, no. 1-3, pp. 186–203, 2016.

[8] R. Zhang, F. Rossi, and M. Pavone, "Model predictive control of autonomous mobility-on-demand systems," in *Proceedings of IEEE ICRA*, 2016.

[9] F. Miao, S. Han, S. Lin, J. A. Stankovic, D. Zhang, S. Munir, H. Huang, T. He, and G. J. Pappas, "Taxi dispatch with real-time sensing data in metropolitan areas: A receding horizon control approach," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 463–478, 2016.

[10] F. Miao, S. Han, S. Lin, Q. Wang, J. A. Stankovic, A. Hendawi, D. Zhang, T. He, and G. J. Pappas, "Data-driven robust taxi dispatch under demand uncertainties," *arXiv preprint arXiv:1603.06263*, 2016.

[11] K. Spieser, K. Treleaven, R. Zhang, E. Frazzoli, D. Morton, and M. Pavone, "Toward a systematic approach to the design and evaluation of automated mobility-on-demand systems: A case study in singapore," in *Road Vehicle Automation*. Springer, 2014, pp. 229–245.

[12] K. Wong and M. G. H. Bell, "The optimal dispatching of taxis under congestion: a rolling horizon approach," *Journal of advanced transportation*, vol. 40, no. 2, pp. 203–220, 2006.

[13] C. E. Garcia, D. M. Prett, and M. Morari, "Model predictive control: theory and practice – a survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.

[14] N. Agatz, A. Erera, M. Savelsbergh, and X. Wang, "Optimization for dynamic ride-sharing: A review," *European Journal of Operational Research*, vol. 223, no. 2, pp. 295–303, 2012.

[15] J. Alonso-Mora, S. Samaranayake, A. Wallar, E. Frazzoli, and D. Rus, "On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment," *Proceedings of the National Academy of Sciences*, 2017.

[16] V. Armant and K. N. Brown, "Minimizing the driving distance in ride sharing systems," in *Proceedings of International Conference on Tools with Artificial Intelligence*, 2014.

[17] B. L. Golden, S. Raghavan, and E. A. Wasil, *The vehicle routing problem: latest advances and new challenges*. Springer Science & Business Media, 2008, vol. 43.

[18] G. Berbeglia, J.-F. J. Cordeau, and G. Laporte, "Dynamic pickup and delivery problems," *European journal of operational research*, vol. 202, no. 1, pp. 8–15, 2010.

[19] N. Agatz, A. L. Erera, M. W. Savelsbergh, and X. Wang, "Dynamic ride-sharing: A simulation study in metro atlanta," *Procedia-Social and Behavioral Sciences*, vol. 17, pp. 532–550, 2011.

[20] S. Ma, Y. Zheng, and O. Wolfson, "T-share: A large-scale dynamic taxi ridesharing service," in *Proceedings of IEEE International Conference on Data Engineering*, 2013.

[21] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[22] J. K. Lenstra and A. H. G. R. Kan, "Complexity of vehicle routing and scheduling problems," *Networks*, vol. 11, no. 2, pp. 221–227, 1981.

[23] B. Donovan and D. B. Work, "Using coarse gps data to quantify city-scale transportation system resilience to extreme events," *arXiv preprint arXiv:1507.06011*, 2015.

[24] M. Grant, S. Boyd, and Y. Ye, "Cvx: Matlab software for disciplined convex programming," 2008.

[2] The computations are carried out on a Windows laptop with Intel(R) i7-3520M 2.9GHz CPU.