Two-Stage Request Scheduling for Autonomous Vehicle Logistic System

James J. Q. Yu, *Member, IEEE*

***Abstract*— Autonomous vehicles are expected to play an important role in handling the last mile logistics in intelligent transportation systems thanks to their unmanned nature and full-fledged controllability. Recently, an autonomous vehicle logistic system (AVLS) was proposed, which employs autonomous vehicles to serve logistic requests and utilize the excessive renewable energy generated by distributed generations. In this paper, we propose an optimization problem for AVLS to develop schedules for request allocation, vehicle routing, and battery charging. By considering the unique characteristics of AVLS, the proposed scheduling problem can exploit its advantages in goods delivery and renewable energy utilization over existing logistic request allocation algorithms. We formulate the problem as a mixed integer non-linear program. To improve its scalability, we also devise a two-stage scheduling methodology to approach the optimal solutions of the original problem. We conduct comprehensive simulations to assess the performance of the proposed request scheduling problem and two-stage scheduling methodology. The results indicate that the proposed problem can improve the efficacy of AVLS in terms of total travel distance and utilized renewable energy, and the two-stage methodology can develop near-optimal solutions with notably reduced computation time.**

***Index Terms*— Autonomous vehicles, logistic system, vehicle routing, request allocation, intelligent transportation.**

I. Introduction

INTELLIGENT transportation system (ITS) is an important component of the future sustainable smart cities [1]. It is envisioned that ITS can accommodate the increasing volume of transportation demand of human society while imposing limited and controllable damage to the environment [2]. Autonomous vehicles (AVs), a kind of vehicles capable of driving without human intervention, are among the key constituents of the system [3]. They can adapt and respond to various transportation situations and events thanks to their sensing capability and full-fledged controllability. Furthermore, since most current AVs are electric vehicles (EVs), e.g., Tesla auto-cars [4], they can significantly reduce the carbon footprint of the transportation sector on the environment. Recent industrial developments and investigations suggest that AVs and EVs will revolutionize the automobile industry and become prevalent in the next decade [5].

One important function of transportation systems is to handle the last mile logistics, which refers to the last leg of goods delivery from storage hubs to final destinations [6]. It is reported that this process is among the most expensive components in the supply chain due to the significant labor and fuel costs [7]. Meanwhile, the demand of last mile logistic service in mature markets, such as U.S. and Germany, is witnessing growth rates ranging between seven to ten percent annually, mainly driven by e-commerce [7]. The increasing demand is advocating ITS to adopt more cost-efficient solutions for last mile logistics. In fact, AVs can act as a good candidate to handle these demands and form a new logistic system with high efficiency and flexibility [7], [8].

Recently, an autonomous vehicle logistic system (AVLS) [8] was proposed, in which AVs serve as logistic carriers. AVLS integrates AVs/EVs, logistics, and renewable energy management in one system. Different from conventional logistic systems, AVLS greatly benefits from the "electric" characteristics of vehicles and the excessive energy generated by renewable energy sources. AVLS manages a fleet of AVs/EVs, which are driven by on-board batteries, to serve logistic requests in the transportation system thanks to their full-fledged controllability and energy storage capability. Specifically, AVLS considers all vehicles in the system as EVs without human drivers. When considering the routing and charging of these vehicles, AVs are identical to EVs if EV drivers can strictly implement instructions from the system. We focus on the EV nature of the fleet in this work. In addition, renewable fluctuations can be partially mitigated [9], resulting in less power system stability issues and improved renewable penetrations in future smart cities [10]. And the adoption of AV can lead to a reduced labor cost, making the system more economic efficient [8].

However, the previous effort to serve logistic requests proposed in [8] is incomplete. While the routing and charging strategy for each vehicle has been investigated in the literature, it remains unclear on how to allocate logistic requests to individual vehicles considering both components' characteristics. The quality of request allocation strategy has a significant impact on the system performance, as bad request-vehicle allocations may lead to unnecessary vehicle routing and/or battery charging behaviors, rendering deteriorated system performance. In the literature, it was assumed that the requests are assigned to vehicles using existing request allocation

Manuscript received February 1, 2018; revised April 29, 2018 and June 10, 2018; accepted June 16, 2018. This work was supported by the National Natural Science Foundation of China under Grant 51707170. The Associate Editor for this paper was D. Sun.

The author is with the HKU Shenzhen Institute of Research and Innovation (SIRI), Shenzhen 518057, China, and also with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong (e-mail: jqyu@eee.hku.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2018.2849091

algorithms, e.g., [11]–[13]. Nonetheless, these allocation algorithms overlook the unique characteristics of AVLS, such as renewable energy utilization. Therefore, they cannot fully enjoy the advantages of AVLS, specifically EVs, over existing last mile logistic systems. A tailor-made request allocation strategy is required to exploit the goods delivery and renewable energy utilization capability of AVLS.

In this paper, we investigate the request allocation process in AVLS and propose a request scheduling methodology to handle this task. We formulate an optimization problem to develop schedules for delivering logistic requests, in which request allocation, vehicle routing, and battery charging are jointly considered. In addition, since solving the proposed problem is computationally expensive, we also develop a two-stage scheduling methodology to pursue the global optimum. The proposed methodology separates the request allocation and battery charging schedules from the route generation process, which can be later calculated on-line. The contribution of this paper is summarized as follows:

- We formulate a request scheduling problem to address the request allocation issue in AVLS. The proposed problem jointly optimizes request allocation, vehicle routing and battery charging to develop globally optimal AVLS schedules.
- We propose a two-stage scheduling methodology to efficiently solve the formulated problem, which is computationally expensive. The proposed methodology separates the route calculations from the others, and can effectively reduce the computation time while maintaining the solution quality.
- We validate the performance of the proposed scheduling problem and two-stage methodology with extensive simulations considering real-world transportation system data.

The rest of this paper is organized as follows. We provide and discuss the background of this work in Section II. Then we introduce the model of AVLS considering the request allocation process in Section III. Section IV presents the formulation of the proposed request scheduling problem with analyses, and the proposed two-stage scheduling methodology is elaborated on in Section V. In Section VI, the performance of the proposed scheduling problem and two-stage methodology is evaluated with comprehensive simulations. Finally, this paper is concluded in Section VII.

II. Background

AVLS was firstly proposed in [8] to employ AVs/EVs for last mile logistics. In the system, a control center manages a fleet of vehicles to serve logistic requests and get charged at Distributed Generations (DGs). The complete delivery process is divided into two steps: scheduling and implementation. In [8], all logistic requests are pre-assigned to different vehicles based on existing request allocation methods. The control center first gathers required information on the transportation network, vehicles, logistic requests, and participating DGs in the system. Then it develops the optimal routing and charging plan for each AV considering all system information. The plan is later transmitted to the respective vehicle. Contributed by the recent development of vehicular communication technologies, data collection and plan instruction processes can be facilitated [14].

DG units refer to small-size power generating plants directly connected to the distribution network or on the power customer site. Renewable DGs can provide environmentally friendly, economic, and reliable power sources to energy consumer [15], and their urban penetration has significantly increased in the past decade contributed by societies' increasing concern on greenhouse gas emission [16]. Additionally, typical DGs are available in modular forms, making these generations easily adopted in urban area where large power plant sites are scarce [16]. In the meantime, renewable energy fluctuations and intermittencies can introduce stability issues to the power grid [17]–[19]. It may contribute to the system stability if the excessive renewable energy can be utilized locally at the DG sites, and AVLS is among the feasible energy consumption solutions.

While there exists little research on adopting AVs/EVs into logistic system considering vehicle charging or renewable energy utilization, previous studies have advocated the active participation of AVs and EVs in the logistic service. It is expected that AVs will deliver close to 100 percent of consumer logistic requests and more than 80 percent of all parcels in the next decade according to [7], due to the diminishing labor and fuel cost. Research has shown the economic efficiency of AVs in modern intelligent transportation systems [20]. Furthermore, battery-driven vehicles have already demonstrated advantages over traditional diesel-driven counterparts in real-world logistics scenarios in China [21], Portugal and Spain [22]. The results indicated good "environmental performance" and "technical performance", and participating companies were satisfied with EVs and seeking for opportunities to enlarge their fleets [23]. These real-world implementations of adopting EVs in logistics also work as examples and guidelines for potential AV-driven logistic systems. On the other hand, logistic request allocation has been extensively investigated in many previous studies, some of which are labeled as "cargo allocation" [24], "freight allocation" [25], etc. Interested readers can refer to [26], [27] for surveys on recent development of strategies for logistic systems, including but not limited to request allocation strategies. However, a careful investigation on the literature reveals that no such strategy considers battery charging during the traverse, which is an essential component in AVLS. Hence, AVLS requires a tailor-made request allocation strategy to fully exploit its merits, and this paper aims to bridge the research gap.

III. Autonomous Vehicle Logistic System Model

In this work, the AVLS model is formulated based on the previous work [8]. Significant changes and enhancements are introduced to the original system to integrate request allocation, vehicle routing, and battery charging processes into the system.

In AVLS, there are generally five basic components that influence the request scheduling process, i.e., transportation

TABLE I

DEFINED SYMBOLS FOR AVLS SYSTEM COMPONENTS

Symbol	Definition
$\mathcal{V}$	Set of nodes in the transportation system (road intersection, request service and charging facility locations).
$\mathcal{E}$	Set of road segments connecting nodes.
D_{ij},T_{ij}	Travel distance (km) and time (h) of $(i,j) \in \mathcal{E}$.
$\mathcal{K}$	Set of vehicles in the system.
L_k^0,L_k^*	Initial and stopping locations of vehicle k.
$\overline{B}_k$,B_k^0	Battery capacity and initial energy of k in kWh.
R_k^+,η_k	Maximum charging rate (kW) and efficiency (%) of k.
$\mathcal{E}_k$	Set of road segments on which k can drive.
$R_{k,ij}$	Energy consumption rate for k on $(i,j) \in \mathcal{E}_k$ in kWh/km.
$\overline{C}_k$	Logistic capacity of k in per units.
$\mathcal{Q}_k$	Set of logistic requests that k is currently serving.
$\mathcal{Q}$	Set of logistic requests to be served by any vehicle.
P_q^0,P_q^*	Pickup and delivery location of any request q in the system.
D_q^0,D_q^*	Pickup and delivery service time of q in hours.
$\underline{T}_q^0$,$\overline{T}_q^0$	Earliest and latest time to pick up q.
$\underline{T}_q^*$,$\overline{T}_q^*$	Earliest and latest time to deliver q.
C_q	Logistic capacity required on vehicle to serve q in per units.
$\mathcal{G}$,$\mathcal{D}$	Sets of renewable DGs and depots.
P_g,P_d	Locations of renewable DG g and depot d.
$\mathcal{V}^{\mathrm{c}}$	Set of locations of all charging facilities in the system.
$\Omega_g(t)$	Maximum charging power of g at time t in kWh.

network, vehicles, logistic requests, renewable DGs, and depots [8]. Table I presents the symbols employed in their definitions, and the mathematical models are explained as follows.

The transportation network of AVLS is defined as a directed graph $\mathrm{G}(\mathcal{V}, \mathcal{E})$. A node $i \in \mathcal{V}$ in the graph can represent a road intersection, the location of a charging facility, or a logistic request service point (pickup or delivery location). Each edge $(i, j) \in \mathcal{E}$ represents a segment of vehicular road in the transportation system, which is associated with a length D_{ij} and a travel time T_{ij}. Note that in order to account for possible asymmetries in the transportation network, D_{ij} and T_{ij} may not be identical to D_{ji} and T_{ji}, respectively.

We denote the set of vehicles in the system by $\mathcal{K}$. Vehicles in the system can have heterogeneous vehicular configurations, i.e., different properties and states such as logistic capacity and initial state-of-charge (SoC). At the planning time, AV $k \in \mathcal{K}$ starts from its initial location $L_k^0 \in \mathcal{V}$ and tries to pick up and deliver logistic requests before finally stopping at $L_k^* \in \mathcal{V}$. We use $\overline{B}_k$, B_k^0, R_k^+, and η_k to denote the capacity, initial energy, maximum charging rate, and charging efficiency of the battery in k, respectively. Set $\mathcal{E}_k$ denotes the set of feasible road segments $(i, j) \in \mathcal{E}$ on which k can traverse, and obviously $\mathcal{E}_k \subseteq \mathcal{E}$. For each $(i, j) \in \mathcal{E}_k$, the energy consumption rate for k to drive on the road segment is denoted by $R_{k,ij}$. Finally, k has a logistic capacity defined by $\overline{C}_k$, and it initially has a set of on-board logistic requests $\mathcal{Q}_k$ to be delivered.

The logistic requests in AVLS can be characterized by the following attributes. We use $\mathcal{Q}$ to denote the collection of logistic requests that has not been served by any vehicles. For any request $q \in \bigcup_{k\in\mathcal{K}} \mathcal{Q}_k \cup \mathcal{Q}$ in the system (on-board or to be served), $P_q^0 \in \mathcal{V}$ and $P_q^* \in \mathcal{V}$ denote its pickup and delivery locations, respectively. Both request pickup and delivery incur service times, which are defined by D_q^0 and D_q^*, respectively. There are also time windows for picking up q, denoted by $[\underline{T}_q^0, \overline{T}_q^0]$, and delivering q, denoted by $[\underline{T}_q^*, \overline{T}_q^*]$. In order to serve request q, a logistic capacity C_q must be reserved on its service vehicle.

There are two types of charging facilities in AVLS, namely, renewable DGs and depots. In this work, we follow the configurations in [28] [29] and consider a set of renewable DGs[1] $g \in \mathcal{G}$. For g located at $P_g \in \mathcal{V}$, it can provide at most $\Omega_g(t) \geq 0$ power to charge vehicles with its excessive energy at time t. In addition, there is a collection of depots $d \in \mathcal{D}$ located at $\{P_d \in \mathcal{V} | d \in \mathcal{D}\}$, which can also charge vehicles in the system. It is assumed that these depots directly draw power from the main grid, thus the aggregated charging power is not limited. We use $\mathcal{V}^{\mathrm{C}}$ to represent the locations of all charging facilities in the system, i.e., $\mathcal{V}^{\mathrm{C}} = \{P_g | g \in \mathcal{G}\} \cup \{P_d | d \in \mathcal{D}\}$. While renewable DGs are scattered in the city, returning the few depots for charging can be inefficient. Meanwhile, the restricted charging power of these renewable DGs renders charging at depots sometimes necessary. How to optimally allocate and route vehicles to serve logistic requests while maintaining sufficient SoC at renewable DGs or depots is an important problem.

IV. AVLS REQUEST SCHEDULING PROBLEM

As discussed in Sections II and III, it is of great importance to schedule vehicles in AVLS to serve requests with efficacy. In particular, AVLS is an application of ITS, which aims at accommodating a massive volume of vehicles and transportation demands while advocating green transportation [1]. In this section, we formulate a request scheduling problem for AVLS considering these ITS objectives and AVLS constraints.

A. Decision Variables and Objective Functions

In this problem, we first introduce three sets of decision variables which correspond to the three aspects of request scheduling problem, i.e., request allocation, vehicle routing, and battery charging. We define a binary variable $x_q^k \in \mathbb{B}$ to indicate whether request $q \in \mathcal{Q}$ will be served by vehicle $k \in \mathcal{K}$. In addition, another binary variable $y_{ij}^k \in \mathbb{B}$ is used to represent whether k will drive on the road segment $(i, j) \in \mathcal{E}_k$ in its routing plan.[2] Finally, we use $p_i^k(t) \geq 0$ to stand for the charging profile of k at node $i \in \mathcal{V}$ at time t.

Besides the above variables, we also adopt the following ancillary decision variables to describe the status of k at an arbitrary node i in the network. These variables greatly facilitate the formulation of the optimization problem to be introduced later:

- $t_i^k \geq 0$: the time when k arrives at i,
- $d_i^k \geq 0$: the duration of stay of k at i,
- $l_i^k \geq 0$: the reserved logistic capacity of k at i, and
- $c_i^k \geq 0$: SoC of k when it arrives at i.

[1] This model can be extended to include conventional DGs which utilize dispatchable energy sources, or on/off-grid self-sustainable DGs.

[2] Here, for the ease of formulation, we assume that a vehicle can drive along each road segment and visit each node in the transportation network at most once. In practice, duplicates of road segments and nodes can be added to the network if a vehicle needs to drive along or visit them for multiple times.

TABLE II
DEFINED DECISION VARIABLES FOR REQUEST SCHEDULING

Symbol	Definition
$x_q^k \in \mathbb{B}$	Request $q \in \mathcal{Q}$ will be served by vehicle k or not.
$y_{ij}^k \in \mathbb{B}$	Vehicle k will drive on the road segment (i, j) in its routing plan or not.
$p_i^k(t) \geq 0$	Charging profile of k at node i at time t.
$t_i^k \geq 0$	The time when vehicle k arrives at node i.
$d_i^k \geq 0$	The duration of stay of vehicle k at node i.
$l_i^k \geq 0$	The reserved logistic capacity of vehicle k at node i.
$c_i^k \geq 0$	SoC of vehicle k when it arrives at node i.

If k does not visit i in its route, the values of corresponding decision variables are meaningless and not considered in the calculations. The decision variables are summarized in Table II.

Recall that AVLS serves logistic requests with efficacy and favors green transportation. These can be achieved by minimizing the total driving distances for all vehicles, and maximizing the utilization of renewable resources. On the one hand, driving distance has always been a key concern in vehicle scheduling problems [6], and longer travel distance implies more energy consumption and heavier traffic. Hence, shorter routes are more preferred:

$$\text{minimize } D = \sum_{k \in \mathcal{K}} \sum_{(i,j) \in \mathcal{E}_k} D_{ij} \, y_{ij}^k.$$

On the other hand, renewable penetrations can be further enhanced in smart cities if AVLS can effectively utilize excessive renewable energy [30]. Furthermore, more DG excessive energy consumed locally by charging vehicles also means that less "unplanned" power will be injected to the main grid, which reduces the impact of renewable generations on power system stability [19]. This objective can be implemented by

$$\text{maximize } E = \sum_{k \in \mathcal{K}} \sum_{i \in \{P_g | g \in \mathcal{G}\}} \int_{t_i^k}^{t_i^k + d_i^k} p_i^k(t) dt.$$

B. Constraints

To develop feasible schedules for vehicles in AVLS, system constraints must be considered. Specifically, the constraints can be classified into three groups, i.e., request allocation, vehicle route, and battery charging constraints.

1) Request Allocation Constraints: In AVLS, each logistic request must be served by exactly one vehicle. This can be ensured by

$$\sum_{k \in \mathcal{K}} x_q^k = 1, \quad \forall q \in \mathcal{Q}. \tag{1}$$

In order to serve an arbitrary request q, AV k must visit its pickup and delivery locations:

$$\sum_{(j,i) \in \mathcal{E}_k} y_{ji}^k \geq x_q^k, \quad \forall k \in \mathcal{K}, \ q \in \mathcal{Q}, \ i \in \{P_q^0, P_q^*\}, \tag{2}$$

where $\sum_{(j,i) \in \mathcal{E}_k} y_{ji}^k$ denotes the number of incoming flows to node i. In addition, the time of visit should fall into the given time windows, and the duration of stay should be long enough to facilitate the (un)loading process:

$$\underline{T}_q^0 \leq t_i^k \leq \overline{T}_q^0, d_i^k \geq D_q^0, \quad i = P_q^0; \tag{3a}$$

$$\underline{T}_q^* \leq t_i^k \leq \overline{T}_q^*, d_i^k \geq D_q^*, \quad i = P_q^*; \quad \forall k \in \mathcal{K}, \quad q \in \mathcal{Q}, \ x_q^k = 1. \tag{3b}$$

Finally, on-board requests $q \in \mathcal{Q}_k$ of k also impose similar constraints on their service vehicles:

$$\sum_{(j,i) \in \mathcal{E}_k} y_{ji}^k \geq 1, \quad \forall k \in \mathcal{K}, \ q \in \mathcal{Q}_k, \ i = P_q^*; \tag{4a}$$

$$\underline{T}_q^* \leq t_i^k \leq \overline{T}_q^*, d_i^k \geq D_q^*, \quad \forall k \in \mathcal{K}, \ q \in \mathcal{Q}_k, \ i = P_q^*. \tag{4b}$$

The capacity of AV k is capped by $\overline{C}_k$:

$$l_i^k \leq \overline{C}_k, \quad \forall k \in \mathcal{K}, \tag{5}$$

and the reserved capacity at a node can be derived from that of the previous node along the route of k:

$$l_i^k + \sum_{q \in \mathcal{Q}} C_{q,i}^+ - \sum_{q \in \mathcal{Q}} C_{q,i}^- = l_j^k, \quad \forall k \in \mathcal{K}, \ (i,j) \in \mathcal{E}_k, \ y_{ij}^k = 1, \tag{6}$$

where

$$C_{q,i}^+ = \begin{cases} C_q & \text{if } i = P_q^0 \\ 0 & \text{otherwise} \end{cases}, \quad C_{q,i}^- = \begin{cases} C_q & \text{if } i = P_q^* \\ 0 & \text{otherwise} \end{cases}$$

are two ancillary constants which represent the capacity changes at an arbitrary node by a specific logistic request. Finally, the initial logistic capacity can be calculated by

$$l_i^k = \sum_{q \in \mathcal{Q}_k} C_q, \quad \forall k \in \mathcal{K}, \ i = L_k^0. \tag{7}$$

2) Vehicle Route and Battery Charging Constraints: The previous work [8] formulated a preliminary routing and charging problem. In this work, we summarize the constraints in the literature with brief introductions, and adapt them to the presented system model in Section III.

We formulate the continuous route of each AV k in the system with the network flow model:

$$\sum_{(j,i) \in \mathcal{E}_k} y_{ji}^k - \sum_{(i,j) \in \mathcal{E}_k} y_{ij}^k = \begin{cases} 0 & \forall i \in \mathcal{V} \setminus \{L_k^0, L_k^*\} \\ -1 & i = L_k^0 \\ 1 & i = L_k^*, \end{cases} \quad \forall k \in \mathcal{K}. \tag{8}$$

In this equality constraint, $\sum_{(j,i) \in \mathcal{E}_k} y_{ji}^k$ and $\sum_{(i,j) \in \mathcal{E}_k} y_{ij}^k$ are the numbers of incoming and outgoing flows of node i. For the starting and stopping location nodes, there should be an extra outgoing and incoming flow to initiate and terminate a continuous flow. Otherwise, these two values should be the same.

While (8) guarantees that the route between the starting and stopping locations is continuous, there may exist islanded subtours which are round tours but not connected to the starting location, i.e., un-reachable by the vehicle. Such subtours can be eliminated if the travel time constraint is considered [12]:

$$t_i^k + d_i^k + T_{ij} \le t_j^k, \quad \forall k \in \mathcal{K},\ (i,j) \in \mathcal{E}_k,\ y_{ij}^k = 1. \tag{9}$$

This constraint suggests that if k drives along (i, j), then the arrival time at j can be deduced using the arrival time at i, the duration of stay at i, and the travel time on (i, j).

Finally, vehicles require energy to drive, and in AVLS this characteristic is reflected in the form of battery energy. The following constraints are constructed to prevent batteries from either energy deficit or being overcharged:

$$\overline{B}_k c_i^k + \int_{t_i^k}^{t_i^k+d_i^k} p_i^k(t)dt - D_{ij} R_{k,ij} = \overline{B}_k c_j^k, \quad \forall k \in \mathcal{K},\ (i,j) \in \mathcal{E}_k,\ y_{ij}^k = 1; \tag{10}$$

$$\overline{B}_k c_i^k = B_k^0, \quad \forall k \in \mathcal{K}, i = L_k^0; \tag{11}$$

$$p_i^k(t) \le R_k^+, \quad \forall k \in \mathcal{K},\ i \in \mathcal{V}^{\mathrm{C}},\ t \ge 0; \tag{12}$$

$$\int_0^{t_i^k} p_i^k(t)dt + \int_{t_i^k+d_i^k}^{+\infty} p_i^k(t)dt = 0, \quad \forall k \in \mathcal{K},\ i \in \mathcal{V}^{\mathrm{C}}; \tag{13}$$

$$p_i^k(t) = 0, \quad \forall k \in \mathcal{K},\ i \in \mathcal{V} \setminus \mathcal{V}^{\mathrm{C}},\ t \ge 0; \tag{14}$$

$$\sum_{k\in\mathcal{K}} p_i^k(t) \le \eta_k \Omega_g(t), \quad \forall i \in \{S_g | g \in \mathcal{G}\},\ t \ge 0. \tag{15}$$

Constraint (10) establishes the energy charging and consumption relationship between nodes. Specifically, if k drives along (i, j), the energy at j can be derived based on that at i, the energy charged at i, and the energy consumed to drive on (i, j). Constraint (11) gives the initial SoC of k. Then the remaining constraints limit the charging rates of vehicles in the system. Constraint (12) confines the maximum charging rate to be less than the maximum allowed rate by vehicles. Constraints (13) and (14) ensure that vehicles cannot be charged if they are not staying at a charging facility. Finally, (15) guarantees that the aggregated charging power of all parking vehicles does not exceed the maximum available charing power provided by the renewable DG.

C. Linear Transformation

The constructed optimization problem is non-linear since constraint (3) is conditioned on $x_q^k = 1$, and constraints (6), (9), (10) are conditioned on $y_{ij}^k = 1$. Moreover, the decision "function" $p_i^k(t)$ cannot be effectively handled by most optimization solvers, e.g., [31], [32]. In this work, we adopt classical linear programming transformation [33] and discretization techniques [34] to address these problems.

1) Conditioned Constraints: A straight-forward method to handle conditioned constraints with $x = 1$ is to multiply x on both sides of the (in)equality. However, this transformation introduces a quadratic constraint to the problem, which is more computationally costly than a linear one to tackle. An alternative way is to adopt the "big-M method" [33]. By having a sufficiently large constant M, (3) can be transformed as

$$t_i^k \ge \underline{T}_q^0 x_q^k, \quad t_i^k - (1 - x_q^k)M \le \overline{T}_q^0,$$
$$d_i^k \ge D_q^0 x_q^k, \quad \forall k \in \mathcal{K},\ q \in \mathcal{Q},\ i = P_q^0; \tag{16a}$$

$$t_i^k \ge \underline{T}_q^* x_q^k, \quad t_i^k - (1 - x_q^k)M \le \overline{T}_q^*,$$
$$d_i^k \ge D_q^* x_q^k, \quad \forall k \in \mathcal{K},\ q \in \mathcal{Q},\ i = P_q^*. \tag{16b}$$

Similarly, (6), (9) can be recast as [8]

$$l_i^k + \sum_{q\in\mathcal{Q}} C_{q,i}^+ - \sum_{q\in\mathcal{Q}} C_{q,i}^- - (1 - y_{ij}^k)M \le l_j^k, \tag{17}$$

$$l_i^k + \sum_{q\in\mathcal{Q}} C_{q,i}^+ - \sum_{q\in\mathcal{Q}} C_{q,i}^- + (1 - y_{ij}^k)M \ge l_j^k, \tag{18}$$

$$t_i^k + d_i^k + T_{ij} - (1 - y_{ij}^k)M \le t_j^k, \quad \forall k \in \mathcal{K},\ (i,j) \in \mathcal{E}_k; \tag{19}$$

2) Decision Function $p_i^k(t)$: While it is computationally inefficient to optimize $p_i^k(t)$ in its current form, we can always approximate the optimal $p_i^k(t)$ function using multiple decision variables. We first discretized the scheduling time horizon $\mathcal{T}$ into multiple time slots, which are indexed by τ. Let $\rho_{i,\tau}^k \ge 0$ be the decision variable representing the charging power of k at node i from time τ to $\tau + \Delta\tau$, where $\Delta\tau$ is the length of each time slot. We adopt a new set of ancillary decision variables $h_{i,\tau}^k$ to denote whether k is charging in time slot τ. Evidently, $h_{i,\tau}^k$ is correlated with t_i^k and d_i^k:

$$h_{i,\tau}^k = \begin{cases} 1 & t_i^k \le \tau\Delta\tau \le t_i^k + d_i^k - \Delta\tau \\ 0 & \text{otherwise,} \end{cases} \quad \forall i \in \mathcal{V},\ k \in \mathcal{K},\ \tau \in \mathcal{T}. \tag{20}$$

which can be re-written as [33]

$$h_{i,\tau}^k \le 1 - (t_i^k - \tau\Delta\tau)/M, \tag{21a}$$

$$h_{i,\tau}^k \le 1 + (t_i^k + d_i^k - \Delta\tau - \tau\Delta\tau)/M, \quad \forall i \in \mathcal{V},\ k \in \mathcal{K},\ \tau \in \mathcal{T}. \tag{21b}$$

Consequently, the objective E and constraints (10), (12)–(15) can be respectively transformed into

$$E = \sum_{k\in\mathcal{K}} \sum_{i\in\{P_g|g\in\mathcal{G}\}} \sum_{\tau\in\mathcal{T}} \rho_{i,\tau}^k$$

$$\overline{B}_k c_i^k + \sum_{\tau\in\mathcal{T}} \rho_{i,\tau}^k - D_{ij} R_{k,ij} = \overline{B}_k c_j^k, \quad \forall k \in \mathcal{K},\ (i,j) \in \mathcal{E}_k,\ y_{ij}^k = 1; \tag{22}$$

$$\rho_{i,\tau}^k \le R_k^+, \quad \forall k \in \mathcal{K},\ \tau \in \mathcal{T},\ i \in \mathcal{V}^{\mathrm{C}}; \tag{23}$$

$$\rho_{i,\tau}^k = 0, \quad \forall k \in \mathcal{K},\ \tau \in \mathcal{T},\ i \in \mathcal{V}^{\mathrm{C}},\ h_{i,\tau}^k = 0; \tag{24}$$

$$\rho_{i,\tau}^k = 0, \quad \forall k \in \mathcal{K},\ \tau \in \mathcal{T},\ i \in \mathcal{V} \setminus \mathcal{V}^{\mathrm{C}}; \tag{25}$$

$$\sum_{k\in\mathcal{K}} \rho_{i,\tau}^k \Delta\tau/\eta_k \le \overline{\mathbf{\Omega}}_{g,\tau}\Delta\tau, \quad \forall \tau \in \mathcal{T},\ i \in \{S_g|g \in \mathcal{G}\}, \tag{26}$$

where $\overline{\mathbf{\Omega}}_{g,\tau}$ is the excessive power supply by g at time slot τ, which is a constant value and can be calculated using $\Omega_g(t)$ [8]. Contributed by recent research on renewable energy forecasting techniques, e.g., [35], [36], $\overline{\mathbf{\Omega}}_{g,\tau}$ values in the few immediately future hours can be predicted with a reasonable accuracy. Furthermore, DGs are typically equipped with battery energy storage system, which can effectively mitigate the random fluctuations of renewable energy sources [37], [38].

Hence, we assume that $\overline{\Omega}_{g,\tau}$ is available, which is also a common assumption in previous literature, see [39], [40] for examples.

Finally, (22) and (24) are conditioned on $y_{ij}^k = 1$ and $h_{i,\tau}^k = 0$, respectively. These conditions can be removed using the same big-M method as in Section IV-C1, which yields the following transformed constraints:

$$\overline{B}_k c_i^k + \sum_{\tau\in\mathcal{T}} \rho_{i,\tau}^k - D_{ij} R_{k,ij} - (1 - y_{ij}^k)M \leq \overline{B}_k c_j^k, \tag{27}$$

$$\overline{B}_k c_i^k + \sum_{\tau\in\mathcal{T}} \rho_{i,\tau}^k - D_{ij} R_{k,ij} + (1 - y_{ij}^k)M \geq \overline{B}_k c_j^k, \quad \forall k \in \mathcal{K},\ (i,j) \in \mathcal{E}_k; \tag{28}$$

$$\rho_{i,\tau}^k \leq h_{i,\tau}^k M, \quad \forall k \in \mathcal{K},\ \tau \in \mathcal{T},\ i \in \mathcal{V}^{\mathrm{C}}. \tag{29}$$

With the above transformation and approximation, the request scheduling problem for AVLS is formulated as follows:

Problem 1 (Request Scheduling Problem):

minimize D or $-E$

subject to (1), (2), (4), (5), (7), (8), (11), (16)–(19), (21), (23), and (25)–(29).

D. Discussion

The proposed problem inherits the vehicle re-fueling idea from Green Vehicle Routing Problem (GVRP) [41], [42], which aims to help "vehicle fleets in overcoming difficulties that exist as a result of limited vehicle driving range in conjunction with limited refueling infrastructure" [41]. In the meantime, GVRP typically simplifies the vehicle re-fueling process as "pass-and-fully-refuel", which neglects the SoC, (dis)charging rate and waiting time characteristics of EVs. The depots in GVRP can also refuel vehicles without fuel/energy limits. In this work, we provide a fine-grained EV charging and DG capacity limit model with (11), (23), and (25)–(29) to make the problem more realistic. Furthermore, the proposed problem extends GVRP into logistic systems, which have unique characteristics and considerations.

By observing Problem 1, one may also note that the constraints share similarities with the Pickup and Delivery Problem with Time Windows (PDPTW) [43], [44]. Specifically, constraints (1), (2), (4), (5), (7), (8), (11), and (16)–(19) have their counterparts in PDPTW. Nonetheless, the major difference between the proposed problem and PDPTW lies in the remaining constraints, which describe the energy management process of AVLS. In addition, constraints (10) and (13) adopt a continuous decision "function" in the problem, which cannot be properly handled by canonical problem solving techniques to PDPTW. Constraint (15) ((26) after transformation) makes the problem non-separable with respect to vehicle k, which further increased the computational complexity to solve Problem 1. Consequently, we propose a new two-stage scheduling methodology to address this problem, which will be elaborated on in the following section.

V. Two-Stage Scheduling Methodology

In Section IV, the request scheduling problem is formulated as a mixed-integer program. While Problem 1 can be solved using commercial optimizer such as [31] and [32], its integer part and quadratic constraint makes the optimization inefficient. As will be demonstrated in Section VI, the computation time increases drastically with the instance size. In [8], a distributed optimization technique is employed to decompose the problem. However, this technique cannot relax the quadratic constraint, thus still results in a high computation time for large instances. In this section, we propose a novel two-stage scheduling methodology to develop near-optimal schedules for AVLS with efficacy.

We start by observing Problem 1. It is evident that the number of decision variables y_{ij}^k increases much faster than the others when the instance size (road network size, number of vehicles, etc.) increases. If the routing plan, i.e., y_{ij}^k, can be decoupled from all other decision variables when searching for the optimal schedules, it is possible to achieve a much faster computing speed. This is the fundamental idea of the proposed two-stage scheduling methodology. In the first stage, the control center tries to determine 1) how requests $q \in \mathcal{Q}$ should be served by vehicles $k \in \mathcal{K}$, 2) in what sequence k should visit the pickup and delivery locations of its serving requests, 3) which DGs/depots k should visit *en route*, and 4) how much energy k should draw from them. Then in the second stage, each AV k optimizes the detailed route in a distributed manner to serve requests and charge itself, according to the plan developed in the first stage.

A. First Stage: Allocation and Charging

In the first stage, a request allocation and vehicle charging optimization is formulated based on Problem 1. This optimization is formulated based on a new approximated graph of the original transportation network G with a smaller density. We first construct a set of point of interests (POIs) $\mathcal{V}^*$, including the initial and final locations of all vehicles, pickup and delivery locations of all requests, and the locations of all charging facilities:

$$\mathcal{V}^* = \{L_k^0, L_k^* | k \in \mathcal{K}\} \cup \{P_q^0, P_q^* | q \in \bigcup_{k\in\mathcal{K}} \mathcal{Q}_k \cup \mathcal{Q}\} \cup \mathcal{V}^{\mathrm{C}}.$$

In general, vehicles in the system are only interested in stopping at these locations. Based on $\mathcal{V}^*$, a new graph $\mathrm{G}^*(\mathcal{V}^*, \mathcal{E}^*)$ is created with edges constructed as follows:

- For each node $\upsilon \in \{L_k^0 | k \in \mathcal{K}\}$, edges from υ to its nearest γ nodes in $\{L_k^*\} \cup \{P_q^0, P_q^* | q \in \mathcal{Q}_k \cup \mathcal{Q}\} \cup \mathcal{V}^{\mathrm{C}}$ are included in $\mathcal{E}^*$.
- For each node $\upsilon \in \{L_k^* | k \in \mathcal{K}\}$, edges from its nearest γ nodes in $\{P_q^0, P_q^* | q \in \mathcal{Q}_k \cup \mathcal{Q}\} \cup \mathcal{V}^{\mathrm{C}}$ to υ are included in $\mathcal{E}^*$.
- For each node $\upsilon \in \{P_q^0, P_q^* | q \in \bigcup_{k\in\mathcal{K}} \mathcal{Q}_k \cup \mathcal{Q}\} \cup \mathcal{V}^{\mathrm{C}}$, edges from υ to its nearest γ nodes in the same set are included in $\mathcal{E}^*$.
- If the included edges form multiple sub-graphs, the nearest nodes from two different sub-graphs are connected to ensure connectivity.

In the edge construction process, parameter γ is employed to tune the density of G*. Consequently, the newly developed G* is generally significantly sparser than G. After creating G*, the shortest route from the source of each edge $(\upsilon, \omega) \in \mathcal{E}^*$ to its destination is developed using shortest path algorithms, e.g., A* search [45] or Dijkstra's algorithm [46], based on the underlying transportation network G. We use $\mathcal{A}_{\upsilon\omega}$ to denote the set of road segments which construct the shortest route from υ to ω.

Let $z^k_{\upsilon\omega} \in \mathbb{B}$ be the binary decision variable indicating whether k will drive from POI υ to ω in its route. Evidently, the total number of $z^k_{\upsilon\omega}$ is substantially smaller than that of y^k_{ij}, since the former is only related to the number of vehicles, requests, and charging facilities in the system instead of the number of road segments. With $\mathcal{A}_{\upsilon\omega}$ and $z^k_{\upsilon\omega}$, we can re-formulate the allocation and charging problem based on Problem 1 as follows:

Problem 2 (First Stage Allocation and Charging):

$$\text{minimize} \sum_{k\in\mathcal{K}} \sum_{(\upsilon,\omega)\in\mathcal{E}^*} (z^k_{\upsilon\omega} \sum_{(i,j)\in\mathcal{A}_{\upsilon\omega}} D_{ij}) \text{ or } -E$$

$$\text{subject to} \sum_{(\omega,\upsilon)\in\mathcal{E}^*} z^k_{\omega\upsilon} \geq x^k_q, \quad \forall k\in\mathcal{K},\ q\in\mathcal{Q},\ \upsilon\in\{P^0_q, P^*_q\}; \tag{30}$$

$$\sum_{(\omega,\upsilon)\in\mathcal{E}^*} z^k_{\omega\upsilon} \geq 1, \quad \forall k \in \mathcal{K},\ q \in \mathcal{Q}_k,\ \upsilon = P^*_q;$$

$$\sum_{(\omega,\upsilon)\in\mathcal{E}^*} z^k_{\omega\upsilon} - \sum_{(\upsilon,\omega)\in\mathcal{E}^*} z^k_{\upsilon\omega} = \begin{cases} 0 & \forall \upsilon \in \mathcal{V}^* \setminus \{L^0_k, L^*_k\} \\ -1 & \upsilon = L^0_k \\ 1 & \upsilon = L^*_k, \end{cases} \quad \forall k \in \mathcal{K}; \tag{31}$$

$$l^k_\upsilon + \sum_{q\in\mathcal{Q}} C^+_{q,\upsilon} - \sum_{q\in\mathcal{Q}} C^-_{q,\upsilon} - (1 - z^k_{\upsilon\omega})M \leq l^k_\omega, \tag{32}$$

$$l^k_\upsilon + \sum_{q\in\mathcal{Q}} C^+_{q,\upsilon} - \sum_{q\in\mathcal{Q}} C^-_{q,\upsilon} + (1 - z^k_{\upsilon\omega})M \geq l^k_\omega, \tag{33}$$

$$t^k_\upsilon + d^k_\upsilon + \sum_{(i,j)\in\mathcal{A}_{\upsilon\omega}} T_{ij} - (1 - z^k_{\upsilon\omega})M \leq t^k_\omega, \tag{34}$$

$$\overline{B}_k c^k_\upsilon + \sum_{\tau\in\mathcal{T}} \rho^k_{\upsilon,\tau} - E_{\upsilon\omega} - (1 - z^k_{\upsilon\omega})M \leq \overline{B}_k c^k_\omega, \tag{35}$$

$$\overline{B}_k c^k_\upsilon + \sum_{\tau\in\mathcal{T}} \rho^k_{\upsilon,\tau} - E_{\upsilon\omega} + (1 - z^k_{\upsilon\omega})M \geq \overline{B}_k c^k_\omega,$$

$$\forall k\in\mathcal{K},\ (\upsilon,\omega) \in \mathcal{E}^*,\ E_{\upsilon\omega} = \sum_{(i,j)\in\mathcal{A}_{\upsilon\omega}} D_{ij} R_{k,ij};$$

$$\text{(1), (4b), (5), (7), (11), (16), (21), (23), (25), (26), and (29).} \tag{36}$$

In Problem 2, constraints (30)–(36) are transformed from (2), (4a), (8), (17)–(19), (27), and (28) considering G*, respectively. However, since $\mathcal{E}^*$ is generally significantly smaller than $\mathcal{E}$, solving this optimization problem incurs less computation time than Problem 1.

B. Second Stage: Routing

The objective of the second stage is to develop the detailed route for each AV in the system to visit the POIs. The route can be different from the previously calculated shortest route, since the new route considers traffic conditions, e.g., average traffic speed. An intuitive formulation for this stage is to construct an optimization problem for each vehicle to optimize the complete route to visit all POIs determined in the first stage. However, since the sequence to visit each POI has already been decided in Problem 2, the complete route can be developed in multiple steps, each of which only calculates the optimal route to the next POI. For instance, suppose AV k is instructed to visit $i_1, i_2, \cdots, L^*_k$ from L^0_k, the vehicle can develop the route from L^0_k to i_1 for now. When the vehicle reaches i_1, it will then generate the optimal route to i_2. In this way, the problem size and computation time can be significantly reduced. Moreover, the computation can be conducted during the service/charging time at each POI.

Let L'_k be the next POI for k to visit according to the optimal solution to Problem 2, and SoC at L'_k is c'_k in the solution. If L'_k is the pickup or delivery location of a request q, we use $[\underline{T}'_k, \overline{T}'_k]$ to represent its pickup/delivery time window. Consequently, the optimal route from L^0_k to L'_k can be developed by solving the following problem:

Problem 3 (Second Stage Next POI Routing for k):

$$\text{minimize} \sum_{(i,j)\in\mathcal{E}_k} D_{ij}\, y^k_{ij}$$

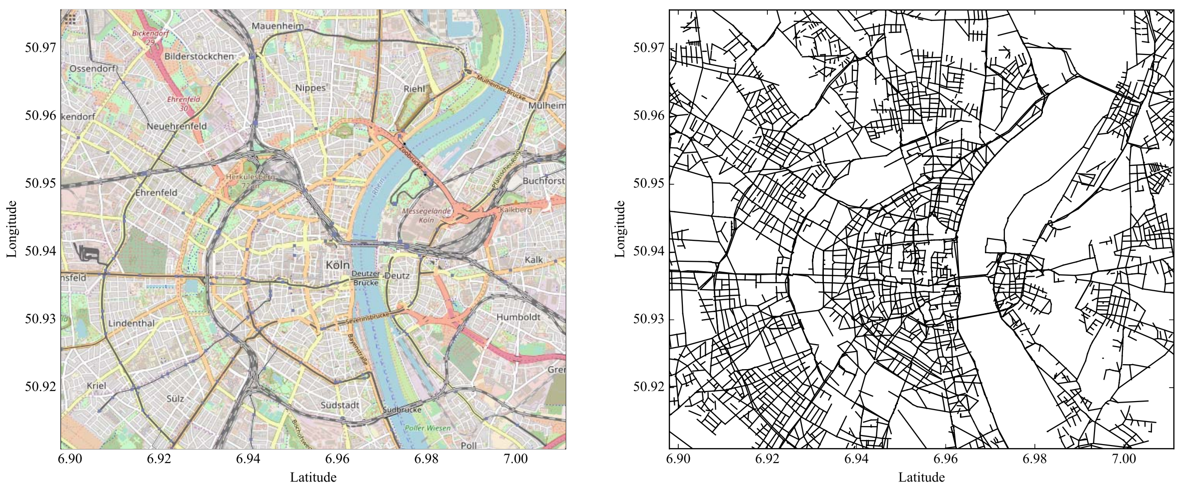
$$\text{subject to } \underline{T}'_k \leq t^k_i \leq \overline{T}'_k, \quad i = L'_k; \tag{37}$$

$$\sum_{(j,i)\in\mathcal{E}_k} y^k_{ji} - \sum_{(i,j)\in\mathcal{E}_k} y^k_{ij} = \begin{cases} 0 & \forall i \in \mathcal{V} \setminus \{L^0_k, L'_k\} \\ -1 & i = L^0_k \\ 1 & i = L'_k; \end{cases}$$

$$c^k_i \geq c'_k; \tag{38}$$

$$\text{(11), (19), (27), and (28).} \tag{39}$$

In Problem 3, constraint (37) is only valid when the next POI L'_k is a pickup/delivery location. This constraint ensures that the time window requirement is met. Constraint (38) is transformed from the original constraint (8). Together with (19), these constraints guide the optimization problem to develop a continuous route from L^0_k to L'_k. Constraint (38) guarantees that when arriving L'_k, the remaining energy is sufficient for subsequent trips. Finally, (11), (27), and (28) calculate the SoC on the way to L'_k, and (19) calculates the arrival time at L'_k. One may note that in this second stage, each AV optimizes its own route only with respect to the minimized travel distance $\sum_{(i,j)\in\mathcal{E}_k} D_{ij}\, y^k_{ij}$. This is due to the fact that vehicle charging plans have been generated in the first stage by maximizing E. The objective of the second stage is to develop the optimal route to the next POI, in which vehicle charging is not involved in. Therefore, it is extraneous to consider E in this stage.



(a) (b)

Fig. 1. The city map and road network of Cologne. Image on the left is obtained from OpenStreetMap [47]. (a) Cologne city map. (b) Cologne city road network.

This two-stage scheduling methodology separates the request allocation and battery charging schedules from the route generation process. Despite the fact that original joint optimization Problem 1 is splitted into Problem 2 and 3, the developed solution can still achieve near-optimal quality. This is because the first stage, although does not optimize the route, implies that the geographically shortest path is optimal to Problem 1. In practice, these paths are typically good enough to provide estimated driving time and related information required by the first stage, even considering the dynamic traffic conditions. Hence, the first stage solution is likely to be near-optimal to the real optimal one. Subsequently, the second stage corrects the shortest path with an optimization to provide the optimal route. The solution quality of the two-stage scheduling methodology is comprehensively studied in the following section.

VI. Case Studies

We conduct a series of case studies to evaluate the proposed two-stage request scheduling methodology. In particular, we first test the quality of solutions developed by the two-stage methodology with respect to the optimal schedules developed by Problem 1, which are considered as global optimum of the problem. Then we assess the system performance of the proposed methodology on different test cases with various network sizes and number of vehicles/requests. Next, we investigate the influence of changing requests on the proposed methodology. Finally, we perform sensitivity tests on the control parameters $\Delta\tau$ and γ.

A. Test Settings

In this work, we employ the real world transportation network and traffic data of Cologne, Germany in the simulation. We first adopt the map data of the city from OpenStreetMap [47] and construct its transportation network. Specifically, all primary, secondary, tertiary, and residential roads are exported to form a graph as depicted in Fig. 1b. Then the average traffic speed of each road in the network is obtained from [48], [49], in which the vehicular mobility traces from 7am to 8am[3] are extracted. Consequently, the travel distance and time of each road can be determined.

We consider vehicles with different configurations in the system. Each vehicle is equipped with a battery, whose charging efficiency is a random value between 80% and 90% to emulate the effect of battery aging [8]. The battery capacity and maximum charging rate are randomly selected from 24kWh/6.6kW, 30kWh/6.6kW (Nissan LEAF [50]), 75kWh/22kW, 85kWh/22kW (Tesla Model S [51]), and 90kWh/22kW (Tesla Model X [52]). The initial SoC of each vehicle is randomly selected between 20% and 90%. In addition, we follow [8] and [53] to configure the energy consumption for driving. Identical to [8], the energy consumption for driving on (i, j) is set to a random value from $0.3D_{ij}$ to $1.0D_{ij}$, which roughly resembles the trend given in [53, eq. (1)]. When constructing $\mathcal{E}_k$ for each vehicle, random 5% to 20% roads are removed from $\mathcal{E}$ to create the sub-graph,[4] and the initial and final locations of the vehicle are randomly placed on $\mathcal{E}_k$. Finally, the logistic capacity for each vehicle is set to a random value between 50 and 100 units, and one to three random requests are generated and assigned to each vehicle as its current on-board requests $\mathcal{Q}_k$.

In the simulations, all logistic requests are randomly generated. The pickup and delivery locations of each request are randomly selected from $\mathcal{V}$, and it is guaranteed that there is at least one vehicle that can reach both locations. The service times for pickup and delivery are set between 5 and 30 minutes, which are typical values in vehicle routing benchmark tests, e.g., [54]. For earliest and latest pickup and delivery time, we first randomly generate four values between zero and three hours. These values are sorted from the smallest to the largest, and are assigned to the earliest

[3]7am to 8am is selected because it is among the periods with the most vehicular traces in [48], which can reflect a more practical traffic condition.

[4]It is guaranteed that the roads in $\mathcal{E}_k$ are connected.

pickup, latest pickup, earliest delivery, and latest delivery times, respectively. Finally, the required logistic capacity is a random value between 5 and 50 unit.

In addition, we assume that there are $\lceil |\mathcal{V}|/25 \rceil$ renewable DGs and $\lceil |\mathcal{V}|/100 \rceil$ depots randomly located in the network [8]. For renewable DGs, we adopt the Eastern and Western Wind Integration Data Set provided by National Renewable Energy Laboratory [55] and utilize the wind power generation data near Los Angeles city. The generation profile is scaled down such that the maximum power output is less than 50kW [56].

In the proposed request scheduling problem, two contradicting objective functions are proposed. Nonetheless, the objectives are independently applied to construct the optimization problem in the case studies. On the one hand, the size of problem instances have different impact on these two objectives as will be analyzed in Section VI-C, and the single-objective formulation can better illustrate the difference. On the other hand, the proposed two-stage methodology is based on single-objective optimization problems, thus cannot handle multi-objective effectively. Multi-objective formulation and solution techniques for AVLS request scheduling are beyond the scope of this paper, and will be investigated in future work.

All simulations are performed on a computer with two Xeon E5 CPU and 128 GB RAM. The testing code is developed in Python 3 and all numerical optimizations are solved using Gurobi optimization solver [31]. In all simulations but the parameter sensitivity test, $\Delta\tau$ and γ are set to one minute and five, respectively.

B. Solution Quality of the Two-Stage Scheduling Methodology

In this work, a request scheduling problem and a two-stage scheduling methodology are presented in Sections IV and V, respectively. We are interested in the performance of the request allocation and vehicle routing plans, especially how well does the two-stage methodology perform compared with the global optimum. In addition, a vehicle routing and charging mechanism was proposed in [8], which relies on existing request allocation plans. We also compare the performance of the proposed scheduling problem and methodology to the previous results.

Due to the relatively high computation complexity of the scheduling problem, we first study the downtown area of Cologne whose road network is shown in Fig. 2. In this network, we consider $|\mathcal{K}| \in \{5, 20\}$ vehicles and $|\mathcal{Q}| \in \{20, 50\}$ new logistic requests to be served. For each combination of $|\mathcal{K}|$ and $|\mathcal{Q}|$, ten independent and random cases are generated. We employ 1) requests scheduling problem, 2) two-stage scheduling methodology, and 3) pre-defined allocation and routing strategy proposed in [8] to handle all random cases, and the objective function results are averaged for statistical significance. For the last strategy proposed in the previous work, the request allocation plans are pre-defined, in which each vehicles will serve its nearest requests until all requests are handled. Since there is no logistic capacity defined for each vehicle in [8], we assume that vehicles have unlimited capacities for the strategy proposed in [8]. Note that we still consider logistic capacities in the proposed problem and two-stage methodology.

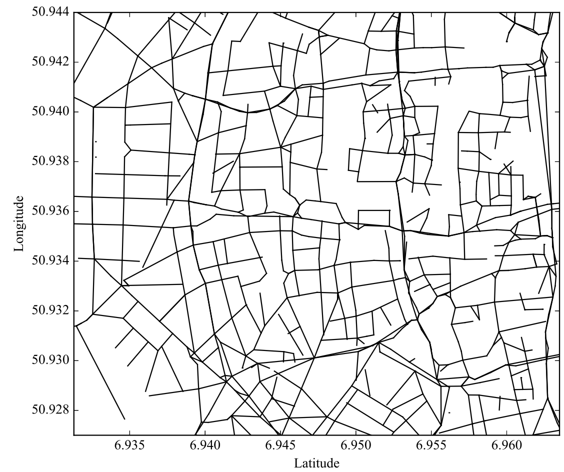


Fig. 2. The city map and road network of Cologne downtown.

The simulation results are presented in Table III. In this table, the minimized travel distance and maximized renewable energy utilization of the request scheduling problem are considered as benchmark values, and the results for the other strategies are presented in the form of solution quality:

$$\text{Solution quality of } D = D^{\text{optim}}/D' \times 100\%,$$
$$\text{Solution quality of } E = E'/E^{\text{optim}} \times 100\%,$$

where D^{optim} and E^{optim} are the optimal values developed by the request scheduling problem, D' and E' are the compared values. In addition, the simulation times for all strategies on minimizing the total travel distance are demonstrated, among which the times for the two stages in the two-stage methodology are listed separately.

In the comparison, it is obvious that the two-stage methodology can achieve an outstanding solution quality in both D and E with a substantially reduced computation time. This is because that in the optimal solution, vehicles are likely to take the shortest geographical path from one location to another. Hence, the shortest paths developed in the first stage by A* search are usually the final paths calculated in the second stage. In addition, the developed problem and methodology can outperform those by the strategy in [8]. While [8] can generate optimal routing and charging plan for each vehicle, the request allocation plan is inferior. Note that the computation time for this strategy is smaller than that of the two-stage methodology. We will show that the two-stage methodology has a better scalability in the following subsection.

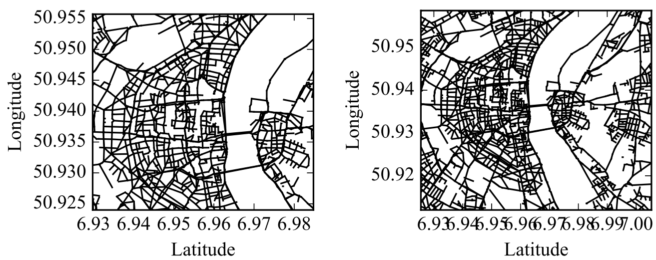
C. Impact of Instance Size

In this subsection we investigate the impact of instance size on the request scheduling performance. In AVLS, there are three major problem configurations that influence the instance size, i.e., network and AV-fleet sizes, and the number of

TABLE III

RESULT COMPARISON ON COLOGNE DOWNTOWN NETWORK

		Scheduling Problem			Two-stage Methodology			Pre-defined Allocation [8]		
$\lvert\mathcal{K}\rvert$	$\lvert\mathcal{Q}\rvert$	D	E	Time	D(%)	E(%)	Time(Stage 1 + 2)	D(%)	E(%)	Time
5	20	16.03km	14.62kWh	78.3s	99.9+%	100%	33.8s+3.2s	96.0%	98.8%	27.2s
5	50	21.96km	18.52kWh	182.2s	99.9+%	99.9+%	76.3s+8.0s	95.3%	98.3%	59.6s
20	20	15.20km	22.48kWh	247.8s	99.9+%	100%	125.9s+1.0s	92.7%	98.6%	115.6s
20	50	19.47km	30.82kWh	503.4s	99.7%	99.9+%	263.8s+2.4s	91.5%	98.5%	222.7s



(a) (b)

Fig. 3. Medium and large-size road networks of Cologne. (a) Medium-size. (b) Large-size.

logistic requests to be served. We first create two more sub-networks from Fig. 1b as depicted in Fig. 3, and label them by "Medium" and "Large", respectively. The networks in Figs. 1b and 2 are labeled by "Full" and "Small", respectively. For all these four networks, $\lvert\mathcal{K}\rvert \in \{5, 20\}$ new vehicles and $\lvert\mathcal{Q}\rvert \in \{20, 50\}$ new requests are randomly generated to construct request scheduling instances of difference sizes. Note that these vehicles and requests can be located at any positions in the network. Hence, larger area will likely lead to "sparser" vehicles and requests, and vice versa. Ten random cases are generated for each instance size, and the simulation results are presented in Table IV.

From the table it can be observed that generally speaking, the minimized total distance and maximized utilized DG energy by the proposed two-stage scheduling methodology increase with the instance size. Specifically, the total driving distance is more related to the number of requests, and increasing the number of vehicles can slightly reduce the distance. On the other hand, utilized DG energy has a positive correlation with both the number of vehicles and requests. This conclusion accords with the intuition that larger transportation networks can lead to increased delivery distance, which in turn advocates vehicles to charge more energy from DGs. In addition, the computation time for optimizing D also increases with the instance size. While the time for first stage increases moderately, the second stage demonstrates significantly higher computation cost in larger networks. This is because that increasing network size alone will not introduce extra control variables in Problem 2, and only the A* search requires more computational effort. On the other hand, the second stage optimizes the path between POIs by Problem 3, of which the number of control variables notably increases with the network size. This renders a longer second stage computation. In the meantime, since the second stage calculations are conducted during service or charging time, the system does not need to wait for it to complete before sending out vehicles. Hence the two-stage methodology can be more efficient than strategies which optimize request allocation and vehicle routing simultaneously before sending any instructions to vehicles.

Comparing with the two-stage methodology, the previous allocation and routing strategy proposed in [8] can also develop sub-optimal solutions. However, the computation time increases drastically with the instance size. This indicates that the proposed two-stage methodology has better scalability than the previous strategy. In city-size transportation networks, the proposed methodology can develop near-optimal plans with significantly shorter time.

In this test, the results to the request scheduling problem are not included. This is because the problem has a high computational complexity as demonstrated in Section VI-B. Optimization solvers cannot calculate optimal solutions in reasonable computation time for "Large" and "Full" networks. Meanwhile, the simulation in Section VI-B indicate that the proposed two-stage scheduling methodology can also develop AVLS schedules with near optimal performance. Therefore, only this methodology is adopted in the test.

D. Dynamic Logistic Requests

In the previous test, all logistic requests are assumed to be known at the time of scheduling. In practical scenarios, however, it is likely that new requests are submitted and/or existing requests are canceled. Therefore, how the proposed request scheduling methodology performs considering such events is worth investigating.

In this section, we adopt the full-sized network as depicted in Fig. 1b with 20 random vehicles and 50 random initial logistic requests to construct a "Dynamic" AVLS scenario. In this scenario, it is assumed that the unserved logistic requests are subjected to changes at the first and second hour. At each of the time, five random unserved requests are removed. Five new requests are randomly generated, whose pickup/delivery locations/time windows and required capacities are random. Their pickup and delivery time windows are also postponed by one or two hours accordingly. We employ the proposed two-stage request scheduling methodology to develop request schedules. Upon changing the logistic requests, the algorithm is re-executed and vehicles will traverse the city following the new POI sequence. The simulation result is compared with another "Static" scenario, in which the same requests are known at the beginning of scheduling and no changes are made.

The comparison is presented in Table V. From the result it can be observed that the performance is not significant influenced by the changing requests. This can be credited

TABLE IV
RESULT COMPARISON ON REQUEST SCHEDULING PROBLEMS WITH DIFFERENT SIZES

			Two-stage Methodology			Pre-defined Allocation [8]		
Network	$\lvert\mathcal{K}\rvert$	$\lvert\mathcal{Q}\rvert$	D	E	Time(Stage 1 + 2)	D (Quality)	E (Quality)	Time
Small			See Table III.					
Medium	5	20	22.95km	21.63kWh	41.4s+6.1s	23.96km (95.8%)	20.70kWh (95.7%)	58.4s
	5	50	31.19km	28.51kWh	87.7s+15.7s	33.51km (93.1%)	27.32kWh (95.8%)	122.8s
	20	20	22.00km	31.84kWh	149.2s+1.9s	23.86km (92.2%)	31.09kWh (97.6%)	232.2s
	20	50	28.94km	44.53kWh	300.5s+4.7s	32.15km (90.0%)	43.97kWh (98.7%)	474.1s
Large	5	20	39.90km	40.49kWh	57.5s+13.3s	40.75km (97.9%)	38.99kWh (96.3%)	206.7s
	5	50	53.89km	49.90kWh	123.9s+32.7s	56.51km (95.4%)	48.20kWh (96.6%)	433.0s
	20	20	40.42km	57.82kWh	209.0s+4.0s	41.36km (97.7%)	56.54kWh (97.8%)	823.3s
	20	50	52.78km	78.19kWh	411.7s+9.9s	55.54km (95.1%)	76.23kWh (97.5%)	1670.2s
Full	5	20	52.41km	55.15kWh	76.2s+56.4s	53.94km (97.1%)	53.07kWh (96.2%)	1145.4s
	5	50	71.99km	67.66kWh	164.9s+139.4s	73.68km (97.7%)	65.77kWh (97.2%)	2426.7s
	20	20	54.34km	79.96kWh	279.0s+17.2s	55.77km (97.4%)	77.26kWh (96.6%)	4651.1s
	20	50	70.82km	106.41kWh	549.8s+43.4s	73.29km (96.6%)	103.72kWh (97.5%)	9492.6s

TABLE V
RESULT COMPARISON ON DYNAMIC REQUEST SCHEDULING SCENARIOS

Scenario	D	E
Dynamic	72.65km	102.52kWh
Static	71.09km	106.80kWh

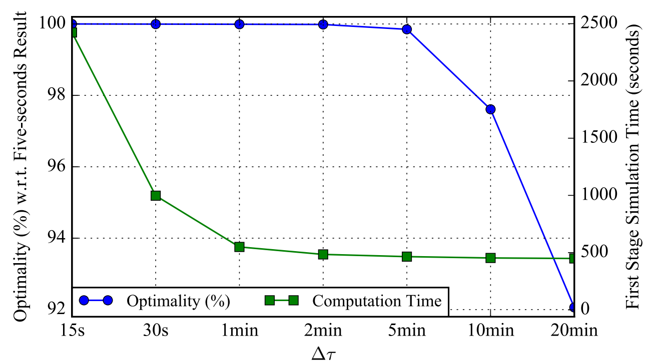
to the standalone route optimization process, i.e., the second stage of the proposed methodology. Since vehicles only need to develop the optimal route to the next POI, no extra computation is wasted when the POI visiting sequence is changed due to logistic request changes. Hence, vehicles can quickly (tens of seconds according to Table IV) adapt to new instructions and serve the requests.

One more thing to note is the computational efficiency of the proposed methodology. In the simulation we observe that the system requires 330.4 and 149.2 seconds to finish the first stage computation after the two request changing time, respectively. On the other hand, the previous strategy in [8] requires more than an hour to finish each optimization according to Table IV, rendering the previous strategy impractical to handle dynamic request changes.

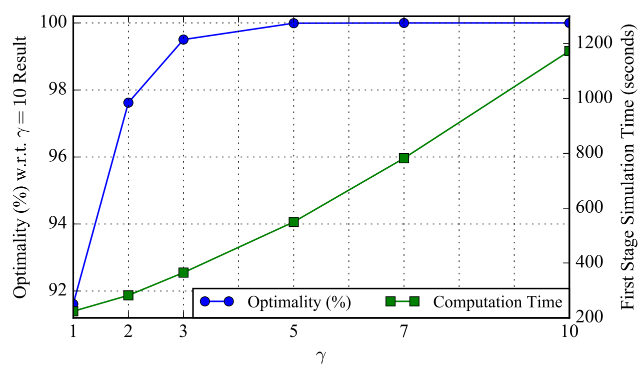
E. Parameter Sensitivity Test

In the proposed request allocation problem, the parameter $\Delta\tau$ is introduced to discretize the continuous decision variable $p_i^k(t)$ into $\rho_{i,\tau}^k$. In addition, the proposed two-stage methodology adopts another parameter γ to control density of graph G*, which in return impacts the complexity of Problem 2. While finer granularities make better approximation, they can also result in higher computational cost. In this section, we adopt the ten random case with 20 vehicles and 50 requests in the "Full" network from Section VI-C to conduct a parameter sweep test. The test aims to investigate the sensitivity of these parameters on the request scheduling performance and required computation time.

We first test the sensitivity of $\Delta\tau$. Similar to [8], we set $\Delta\tau$ to be 15 seconds, 30 seconds, 1 minute, 2 minutes, 5 minutes, 10 minutes, and 20 minutes. In all these cases, $\gamma = 5$, and the maximized DG energy is presented since $\Delta\tau$ mainly concerns the granularity of discretizing $p_i^k(t)$. The simulation results are presented in Fig. 4a. From the figure it is clear that the solution quality deteriorates with $\Delta\tau$, and the performance deteriorates



(a)



(b)

Fig. 4. Results of parameter sensitivity tests with respect to solution quality and computation time. (a) Parameter sensitivity of $\Delta\tau$. (b) Parameter sensitivity of γ.

notably with coarse-grained time slots, e.g., $\Delta\tau \geq 10$ minutes. In the meantime, fine-grained time slots can achieve better renewable energy utilization, but the computation time also drastically increases for $\Delta\tau \leq 30$ seconds. This observation accords with our previous conclusion in [8] that there is a trade-off between solution quality and computation time when selecting $\Delta\tau$ value, and $\Delta\tau = 1$ minute is generally suggested.

We also test the sensitivity of γ with a similar method, i.e., γ is assigned from $\{1, 2, 3, 5, 7, 10\}$ and $\Delta\tau = 1$ minute. The simulation results are depicted in Fig. 4b, where the solution quality on the total travel distance is presented since γ mainly influences the transportation network complexity. It can be observed that increasing γ can significantly improve the solution quality when $\gamma \leq 5$, and the computation time has a roughly linear relationship with γ. Therefore, $\gamma = 5$ is a recommended value for the proposed two-stage request scheduling methodology considering the performance and computation time trade-off.

VII. Conclusion

With the introduction of AVs and EVs to transportation systems, they can play an important role in fulfilling the ever-increasing last mile logistics demand. Recently, an AV-driven logistic system, i.e., AVLS, was proposed, which manages a fleet of AVs to serve logistic requests and charges the vehicles with the excess energy from renewable DGs. However, the previous work employs existing request allocation algorithms to assign logistic requests to vehicles, which potentially undermines the system performance since the algorithms do not consider the unique characteristics of AVLS. In this paper, we propose a request scheduling problem to jointly optimize the request allocation, vehicle routing, and battery charging plans of AVLS. We formulate the problem as a mixed integer non-linear program and employ linear transformation techniques to alleviate the problem computational complexity. Nonetheless, the integer part of the optimization still leads to scalability issues. To develop a practical request scheduling algorithm for AVLS, we further propose a two-stage methodology to the formulated optimization. The proposed methodology separates request allocation and battery charging from vehicle routing when determining the request scheduling plans, and utilizes the service time of each vehicle to calculate its detailed route.

We conduct comprehensive simulations to assess the performance of the proposed request scheduling problem and two-stage scheduling methodology for AVLS. We first investigate the solution quality of the two-stage methodology on the formulated optimization. The simulation results clearly show that the two-stage methodology can generate near-optimal request schedules with notably decreased computation time. In addition, the solution quality is significantly better than the previous work. We also test the impact of instance size and changing requests on the solution quality as well as the computation time, and the results imply that the proposed two-stage methodology has a good scalability and can adapt to request changes well. Last but not least, a parameter sensitive test is conducted to provide guidelines for setting the control parameters.

The future work can be divided into two parts. On the one hand, the first stage of the proposed two-stage methodology can be transformed into a distributed optimization problem, which may further reduce the computation time. On the other hand, we will extend the deterministic optimization formulation with a stochastic model, and investigate possible multi-objective solution techniques to AVLS request scheduling problem.

References

[1] F.-Y. Wang, "Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 630–638, Sep. 2010.

[2] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[3] H. Menouar, I. Guvenc, K. Akkaya, A. S. Uluagac, A. Kadri, and A. Tuncer, "UAV-enabled intelligent transportation systems for the smart city: Applications and challenges," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 22–28, Mar. 2017.

[4] L. Ulrich, "Top 10 tech cars: 2018 [top 10 tech cars]," *IEEE Spectr.*, vol. 55, no. 4, pp. 30–41, Apr. 2018.

[5] D. Mohr, H.-W. Kaas, P. Gao, D. Wee, and T. Möller, "Automotive revolution–perspective towards 2030: How the convergence of disruptive technology-driven trends could transform the auto industry," McKinsey & Company, New York, NY, USA, Tech. Rep., 2016.

[6] D. Cattaruzza, N. Absi, D. Feillet, and J. González-Feliu, "Vehicle routing problems for city logistics," *EURO J. Transp. Logistics*, vol. 6, no. 1, pp. 51–79, 2017.

[7] M. Joerss, J. Schröder, F. Neuhaus, C. Klink, and F. Mann, "Parcel delivery—The future of last mile," McKinsey & Company, New York, NY, USA, Tech. Rep., Sep. 2016.

[8] J. J. Q. Yu and A. Y. S. Lam, "Autonomous vehicle logistic system: Joint routing and charging strategy," *IEEE Trans. Intell. Transp. Syst.*, to be published.

[9] H. Yang *et al.*, "Operational planning of electric vehicles for balancing wind power and load fluctuations in a microgrid," *IEEE Trans. Sustain. Energy*, vol. 8, no. 2, pp. 592–604, Apr. 2017.

[10] P. Mitra and G. K. Venayagamoorthy, "Wide area control for improving stability of a power system with plug-in electric vehicles," *IET Generat., Transmiss. Distrib.*, vol. 4, no. 10, pp. 1151–1163, Oct. 2010.

[11] A. Holzapfel, H. Kuhn, and M. G. Sternbeck, "Product allocation to different types of distribution center in retail logistics networks," *Eur. J. Oper. Res.*, vol. 264, no. 3, pp. 948–966, Feb. 2016.

[12] T. Huth and D. C. Mattfeld, "Integration of vehicle routing and resource allocation in a dynamic logistics network," *Transp. Res. C, Emerg. Technol.*, vol. 17, no. 2, pp. 149–162, 2009.

[13] R. A. Vasco and R. Morabito, "The dynamic vehicle allocation problem with application in trucking companies in Brazil," *Comput. Oper. Res.*, vol. 76, pp. 118–133, Dec. 2016.

[14] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an international standard for wireless access in vehicular environments," in *Proc. IEEE Veh. Technol. Conf. (VTC-Spring)*, May 2008, pp. 2036–2040.

[15] X. Zhang, G. G. Karady, and S. T. Ariaratnam, "Optimal allocation of CHP-based distributed generation on urban energy distribution networks," *IEEE Trans. Sustain. Energy*, vol. 5, no. 1, pp. 246–253, Jan. 2014.

[16] P. S. Georgilakis and N. D. Hatziargyriou, "Optimal distributed generation placement in power distribution networks: Models, methods, and future research," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 3420–3428, Aug. 2013.

[17] J. J. Q. Yu, Y. Hou, A. Y. S. Lam, and V. O. K. Li, "Intelligent fault detection scheme for microgrids with wavelet-based deep neural networks," *IEEE Trans. Smart Grid*, to be published.

[18] J. J. Q. Yu, A. Y. S. Lam, D. J. Hill, Y. Hou, and V. O. K. Li, "Delay aware power system synchrophasor recovery and prediction framework," *IEEE Trans. Smart Grid*, to be published.

[19] Y. Zhang, N. Gatsis, and G. B. Giannakis, "Robust energy management for microgrids with high-penetration renewables," *IEEE Trans. Sustainable Energy*, vol. 4, no. 4, pp. 944–953, Oct. 2013.

[20] J. J. Q. Yu, A. Y. S. Lam, and Z. Lu, "Double auction-based pricing mechanism for autonomous vehicle public transportation system," *IEEE Trans. Intell. Veh.*, vol. 3, no. 2, pp. 151–162, Jun. 2018.

[21] Z. He, P. Chen, H. Liu, and Z. Guo, "Performance measurement system and strategies for developing low-carbon logistics: A case study in China," *J. Cleaner Prod.*, vol. 156, pp. 395–405, Jul. 2017.

[22] G. Duarte, C. Rolim, and P. Baptista, "How battery electric vehicles can contribute to sustainable urban logistics: A real-world application in Lisbon, Portugal," *Sustain. Energy Technol. Assessments*, vol. 15, pp. 71–78, Jun. 2016.

[23] N. Nesterova and H. Quak, "State of the art of the electric freight vehicles implementation in city logistics—Update 2015," Freight Electr. Vehicles Urban Eur., Westminster City, U.K., Tech. Rep. D1.3 addendum 1, 2015.

[24] K. Fagerholt and M. Christiansen, "A travelling salesman problem with allocation, time window and precedence constraints—An application to ship scheduling," *Int. Trans. Oper. Res.*, vol. 7, no. 3, pp. 231–244, 2000.

[25] M. A. Krajewska, H. Kopfer, G. Laporte, S. Ropke, and G. Zaccour, "Horizontal cooperation among freight carriers: Request allocation and profit sharing," *J. Oper. Res. Soc.*, vol. 59, no. 11, pp. 1483–1491, Nov. 2008.
[26] L. C. Coelho, J. Renaud, and G. Laporte, "Road-based goods transportation: A survey of real-world logistics applications from 2000 to 2015," *Inf. Syst. Oper. Res.*, vol. 54, no. 2, pp. 79–96, 2016.
[27] A. A. Juan, C. A. Mendez, J. Faulin, J. de Armas, and S. E. Grasman, "Electric vehicles in logistics and transportation: A survey on emerging environmental, strategic, and operational challenges," *Energies*, vol. 9, no. 2, p. 86, 2016.
[28] W. Shi, X. Xie, C. C. Chu, and R. Gadh, "Distributed optimal energy management in microgrids," *IEEE Trans. Smart Grid*, vol. 6, no. 3, pp. 1137–1146, May 2015.
[29] K. Rahbar, C. C. Chai, and R. Zhang, "Energy cooperation optimization in microgrids with renewable energy integration," *IEEE Trans. Smart Grid*, vol. 9, no. 2, pp. 1482–1493, Mar. 2018.
[30] D. E. Olivares *et al.*, "Trends in microgrid control," *IEEE Trans. Smart Grid*, vol. 5, no. 4, pp. 1905–1919, Jul. 2014.
[31] *Gurobi Optimization—The State-of-the-Art Mathematical Programming Solver*. Accessed: Dec. 2017. [Online]. Available: http://www.gurobi.com/
[32] *CPLEX Optimizer—IBM Analytics*. Accessed: Dec. 2017. [Online]. Available: https://www.ibm.com/analytics/data-science/prescriptive-analytics/cplex-optimizer
[33] G. Dantzig, *Linear Programming and Extensions*. Princeton, NJ, USA: Princeton Univ. Press, 1963.
[34] J. R. Frazer, *Applied Linear Programming*. Upper Saddle River, NJ, USA: Prentice-Hall, 1968.
[35] J. R. Andrade and R. J. Bessa, "Improving renewable energy forecasting with a grid of numerical weather predictions," *IEEE Trans. Sustain. Energy*, vol. 8, no. 4, pp. 1571–1580, Oct. 2017.
[36] E. B. Ssekulima, M. B. Anwar, A. A. Hinai, and M. S. El Moursi, "Wind speed and solar irradiance forecasting techniques for enhanced renewable energy integration with the grid: A review," *IET Renew. Power Gener.*, vol. 10, no. 7, pp. 885–989, Jul. 2016.
[37] Y. Seyedi, H. Karimi, and S. Grijalva, "Distributed generation monitoring for hierarchical control applications in smart microgrids," *IEEE Trans. Power Syst.*, vol. 32, no. 3, pp. 2305–2314, May 2017.
[38] *Distributed Generation of Electricity and its Environmental Impacts | Energy and the Environment | U.S. EPA*. Accessed: Apr. 2018. [Online]. Available: https://www.epa.gov/energy/distributed-generation-electricity-and-its-environmental-impacts
[39] A. Lam, J. J. Q. Yu, Y. Hou, and V. O. K. Li, "Coordinated autonomous vehicle parking for vehicle-to-grid services: Formulation and distributed algorithm," *IEEE Trans. Smart Grid*, to be published.
[40] A. Y. Saber and G. K. Venayagamoorthy, "Plug-in vehicles and renewable energy sources for cost and emission reductions," *IEEE Trans. Ind. Electron.*, vol. 58, no. 4, pp. 1229–1238, Apr. 2011.
[41] S. Erdoğan and E. Miller-Hooks, "A green vehicle routing problem," *Transp. Res. E, Logistics Transp. Rev.*, vol. 48, no. 1, pp. 100–114, 2012.
[42] C. Lin, K. L. Choy, G. T. S. Ho, S. H. Chung, and H. Y. Lam, "Survey of Green vehicle routing problem: Past and future trends," *Expert Syst. Appl.*, vol. 41, no. 4, Part 1, pp. 1118–1138, 2014.
[43] Y. Dumas, J. Desrosiers, and F. Soumis, "The pickup and delivery problem with time windows," *Eur. J. Oper. Res.*, vol. 54, no. 1, pp. 7–22, 1991.
[44] H. Li and A. Lim, "A metaheuristic for the pickup and delivery problem with time windows," *Int. J. Artif. Intell. Tools*, vol. 12, no. 2, pp. 173–186, 2003.
[45] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Syst. Sci. Cybern.*, vol. SSC-4, no. 2, pp. 100–107, Jul. 1968.
[46] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numer. Math.*, vol. 1, no. 1, pp. 269–271, Dec. 1959.
[47] *OpenStreetMap*. Accessed: Dec. 2017. [Online]. Available: http://www.openstreetmap.org/
[48] *Vehicular Mobility Trace of the City of Cologne, Germany*. Accessed: Dec. 2017. [Online]. Available: http://kolntrace.project.citi-lab.fr/#trace
[49] S. Uppoor, O. Trullols-Cruces, M. Fiore, and J. M. Barcelo-Ordinas, "Generation and analysis of a large-scale urban vehicular mobility dataset," *IEEE Trans. Mobile Comput.*, vol. 13, no. 5, pp. 1061–1075, May 2014.
[50] *2017 Nissan LEAF Electric Car*. Accessed: Dec. 2017. [Online]. Available: https://www.nissanusa.com/electric-cars/leaf/
[51] *Model S—Tesla*. Accessed: Dec. 2017. [Online]. Available: https://www.tesla.com/models
[52] *Model X—Tesla*. Accessed: Dec. 2017. [Online]. Available: https://www.tesla.com/modelx
[53] C. De Cauwer, J. Van Mierlo, and T. Coosemans, "Energy consumption prediction for electric vehicles based on real-world data," *Energies*, vol. 8, no. 8, pp. 8573–8593, 2015.
[54] *Multiple Depot VRP Instances—Vehicle Routing Problem*. Accessed: Dec. 2017. [Online]. Available: http://neo.lcc.uma.es/vrp/vrp-instances/multiple-depot-vrp-instances/
[55] *Eastern and Western Data Sets—Grid Modernization—NREL*. Accessed: Dec. 2017. [Online]. Available: https://www.nrel.gov/grid/eastern-western-wind-data.html
[56] K. Strunz, E. Abbasi, and D. N. Huu, "DC microgrid for wind and solar power integration," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 2, no. 1, pp. 115–126, Mar. 2014.

James J. Q. Yu (S'11–M'15) received the B.Eng. and Ph.D. degrees in electrical and electronic engineering from The University of Hong Kong, Hong Kong, in 2011 and 2015, respectively. He is currently an Honorary Assistant Professor and a Post-Doctoral Fellow with the Department of Electrical and Electronic Engineering, The University of Hong Kong. He is also the Chief Research Consultant of GWGrid Inc. and Fano Labs. His research interests include smart city technologies, deep learning and big data industrial applications, power stability analysis, and evolutionary computation.