Double Auction-based Pricing Mechanism for Autonomous Vehicle Public Transportation System

Article · February 2018

DOI: 10.1109/TIV.2018.2804161

3 authors:

James J.Q. Yu
The University of Hong Kong

Albert Lam
The University of Hong Kong

Zhiyi Lu
The University of Hong Kong

Double Auction-based Pricing Mechanism for Autonomous Vehicle Public Transportation System

James J.Q. Yu, Albert Y.S. Lam, and Zhiyi Lu

Abstract—**The Autonomous Vehicle (AV) is expected to be an important "building block" of the future smart city. Recently, an AV-based public transportation system has been successfully developed to provide precise, effective, and intelligent public transportation services. For better quality of service, the system encourages market competition by accommodating multiple AV operators. To facilitate the pricing process, a pricing mechanism was developed but it can only process one service request each time. This can significantly impair the overall passenger admissibility, especially when there are many outstanding requests to be processed. In this paper, we re-design the pricing mechanism for handling multiple requests simultaneously. To do this, we formulate the key component of the mechanism, i.e., request-AV allocation, as a double combinatorial auction-based process. We construct a new winner determination problem that can accommodate requests of different AV service types. We also investigate its duality to devise an efficient service charge determination rule. We evaluate the performance of the proposed mechanism and charging rule with extensive simulations. The results show that the mechanism can result in better social welfare than the original scheme. Moreover, we examine the computational time required and the percentage of successfully served passengers. The simulations demonstrate that the mechanism can make the AV public transportation system more practical.**

Index Terms—**Autonomous vehicle, combinatorial double auction, public transportation system, smart city.**

I. INTRODUCTION

DUE to its driverless and environmentally friendly properties, the Autonomous Vehicle (AV) is gaining increasing attention from the public and research community. Since the maneuvers are computer-controlled, an AV can better adapt to various road conditions with the assistance of available transportation information about its neighborhood [1]. With the introduction of AVs, the number of traffic accidents is expected to decrease significantly [2]. Besides the autonomous self-driving capability, the functionalities of AVs can be further enhanced with inter-vehicular communications [3]. Multiple AVs can constitute a vehicular network and share information with one another. Moreover, a control center can also be employed to systematically coordinate their routes and schedules in order to improve the social welfare[1] [4], resulting in less traffic congestion and more transportation throughput. This grants AVs higher controllability than traditional driver-controlled vehicles from the system-wise perspective.

It is believed that most future transportation applications will involve certain service models of shared fleet operation [5]. The gradual deployment of AVs is expected to lead to critical transformation of transportation system, bringing about new classes of vehicle routing, scheduling, and fleet management problems for AVs [5]. More research on AV fleet management is needed to enhance the efficiency of new services and this work investigates a certain form of AV fleet management. We adopt the recently proposed AV Public Transportation System (AVPTS) [6], in which AVs, as the transportation carriers, are employed for public transits in modern cities; and a control center manages a fleet of AVs to offer on-demand transportation services, with optional ride-sharing capability. Customers can place transportation requests through e-hailing with specific requirements, such as pickup and dropoff locations. The control center allocates appropriate AVs to serve the requests. By means of coordinated routing and scheduling, the system can achieve different objectives and economic benefits to both the system operators and passengers.

While the control center is dedicated to coordinating the allocation, routing, and scheduling of the AVs, a mature deregulated AVPTS should be able to accommodate multiple AV operators, each of which manages its own AVs, constituting an *AVPTS market*. This market introduces competitions among the operators so as to improve service quality and lower operational cost, which are favorable to the passengers. Moreover, more operators contribute more transportation resources to the system so that more requests are likely to be admitted. To operate a multi-tenant system, an additional pricing process is needed to settle the service charges by considering multiple customer requests and operator offers simultaneously. The pricing mechanism should be fair to all participants while maximizing the social welfare.

In this paper, we design a pricing mechanism for the multi-tenant AVPTS which is capable of handling multiple service requests simultaneously. Inspired by typical combinatorial double auctions [7], we modify the Winner Determination Problem (WDP) to optimally allocate available AVs to serve these requests. We consider the duality to settle the service charges. The contribution of this work is summarized as follows:

- We propose a new pricing mechanism for handling multiple service requests in AVPTS;
- We develop a new WDP for request-AV allocation involving multiple service types;

This work was supported in part by the National Natural Science Foundation of China, under Grant No. 51707170.

J.J.Q. Yu and Z. Lu are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong (e-mail: jqyu@eee.hku.hk, zylu@eee.hku.hk).

A.Y.S. Lam is with Fano Labs and the Department of Electrical and Electronic Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong (e-mail: albert@fano.ai).

[1] Social welfare refers to the well-being of the entire society, including but not limited to both vehicle operators and customers.

- We develop an effective mechanism to determine the service charges; and
- We evaluate the performance of the proposed pricing mechanism with extensive simulations.

The rest of this paper is organized as follows. We introduce concepts of AVPTS and propose our pricing mechanism in Section III. We formulate the AV pricing process and develop its WDP in Section IV. A duality-based service charge rule is developed in Section V. In Section VI, we evaluate the system performance with a series of simulations. Finally, we conclude this paper in Section VII.

II. Background

AVPTS was introduced in [6], in which AVs are employed to provide public transit services. The system comprises three types of entities, including customers, AVs, and a control center. Customers submit transportation requests to the control center with the necessary information, such as pickup and dropoff locations, number of passengers, etc. The control center then performs *admission control* to screen out the inadmissible and non-profitable requests and to maximize the social welfare. The system also performs *scheduling* to determine appropriate routes and schedules for the available AVs to accomplish the admitted requests. Those AVs with assigned tasks then execute the instructed service plans to serve the admitted requests. Through the admission control and scheduling processes, the efficiency and capability of the system can be enhanced and other social welfare objectives can also be achieved [6].

In the original design given in [6], the control center is assumed to have full control of all participating AVs. Although this model enables the only operator to maximize the social welfare easily, the pricing issue is overlooked. Customers have to accept any service charges proposed by the operator regardless of their own valuations. This constitutes a monopoly and the operator is encouraged to manipulate the service charges, which sacrifices the welfare of the customers.

Alternatively, like many modern deregulated public transportation systems, it is common to have multiple operators, i.e., business entities. Moreover, thanks to the unmanned nature of AVs and the e-hailing operating paradigm, as used by Uber [8], individual AV owners are strongly incentivized to lease their vehicles for profit during their “off-time”. Therefore the multi-tenant AVPTS are more pragmatic in real-world applications.

There is some related work on designing pricing mechanisms for AVPTS, e.g., [9] and [10]. [9] focused on designing a pricing mechanism for a single transportation request competed by multiple AV operators. A strategy-proof Vickrey-Clarke-Groves (VCG)-based scheme was developed to decide the service charge. [10] focused on the security concerns in the pricing process. A core-selecting auction-based pricing mechanism is designed to address the false-name and shill bidding vulnerabilities of the design proposed in [9]. However, both of them can only handle one request per execution. When it comes to a large number of incoming requests which is probable in a large AVPTS, the order of request handling (i.e., which request is handled sooner and which is later) may significantly influence their admittability and so as the overall social welfare. As stated in [9] and [10] where each request constitutes an individual optimization problem, handling the requests sequentially is not likely to maximize the overall social welfare, which should be attained by jointly considering multiple requests at the same time. Therefore, a pricing mechanism that can handle a collection of requests simultaneously is needed to make the multi-tenant AVPTS more practical.

In this work, we propose a new pricing mechanism for the multi-tenant AVPTS capable of handling multiple requests at the same time. This mechanism is modeled as a double auction, in which multiple bidders compete for multiple items. There is a plethora of literature studying such auctions with promising solutions. For instance, [7] illustrated typical forms of double auctions with performance comparison of several solutions. Double auction has also been employed in modeling other scientific and engineering research problems, e.g., service allocation in mobile cloud computing [11]. The interested reader may refer to [12] and [13] for the details and developments of double auctions.

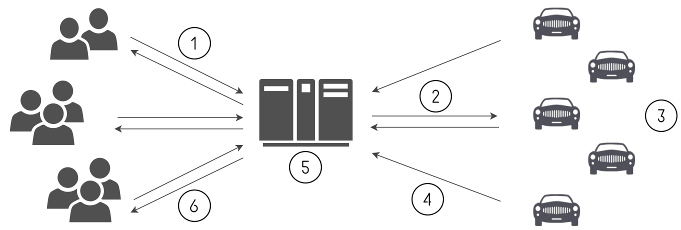


Fig. 1. The pricing process in the multi-tenant AVPTS. Circled numbers indicate the steps in this process.

III. AVPTS Pricing Mechanism

In the multi-tenant AVPTS, the pricing process is used to perform the AV-request allocation and to settle the service charges for transportation requests. The whole process can be broken into several steps, which are schematically illustrated in Fig. 1. In each pricing process, the customers first submit their service requests, with specific maximum service charges, to the control center, say through e-hailing (Step 1). After a certain number of requests have been collected or a pre-defined period for request gathering has expired, the control center distributes the request information to the AV operators (Step 2). Next, each operator individually evaluates the operating cost induced by its governed AVs for serving the requests (Step 3), and use their own strategies to suggest minimum service charges for their AVs to serve the requests (Step 4). After gathering all the bidding proposals, the control center finalizes the service charges for the admissible requests while dropping the inadmissible ones (Step 5). Finally, the AV allocation plan is reported back to the customers (Step 6). When determining the pricing results, various objectives can be achieved. In this work, we focus on maximizing the social welfare.

The pricing mechanism can be implemented in a quasi-online fashion, where the request gathering and processing

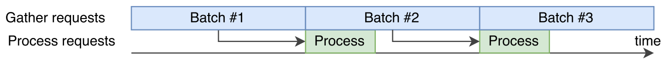


Fig. 2. Online auction mechanism with request gathering and processing executed in parallel.

phases can be executed simultaneously. As depicted in Fig. 2, while gathering a new batch of customer service requests, the control center can be processing the previous batch. For instance, after gathering requests in Batch #1 (Step 1 in Fig. 1), the mechanism will start matching requests with AVs subject to their requirements and availabilities (Steps 2 to 6). At the same time, the newly submitted requests will be collected in Batch #2. The length of request collecting period can be dynamically adjusted by the control center based on the availability of request submissions. As long as the time for Step 1 is longer than that of the remaining steps, the mechanism is stable, i.e., all requests can be processed in finite time.

Based on [9], transportation requests can be categorized into three types, i.e., the splittable, non-splittable, and private services. For the splittable service, passengers originated from a single request may be split into groups, each being served by a different AV. For the non-splittable service, all passengers in the same request must be served by one AV. For these two service types, ride-sharing is permitted, i.e., passengers belonging to different requests may share the same ride. Finally, for the private service, passengers of the same request will occupy the whole vehicle for the ride. In order to utilize vehicular resources effectively, the control center should determine the optimal routes and schedules collectively for the governed AVs to accommodate the requests of different service types. Through ride-sharing, while serving some requests, those AVs with available seats can be further utilized for other requests. This improves the system throughput and decreases the operational cost imposed on each passenger. Thus lower service charges can be expected. Moreover, unlike [9] and [10], the three service types are handled altogether in this work. This can avoid the infeasible situation that a seat is reserved by multiple requests of different service types.

To summarize, the unique features of the proposed auctioneering mechanism that are not realized in [9] and [10] are listed as follows:

- Multiple customers and AV operators are considered concurrently;
- All three service types are handled simultaneously; and
- We develop a charging rule dedicated to the proposed pricing mechanism.

The customers and the operators are generally autonomous entities. Without further assumptions, we cannot manipulate their own pricing strategies. Instead, we can at most facilitate an effective matching between the service requests and AVs at the control center, as an intermediary, to maximize the system welfare. This corresponds to Step 5 in Fig. 1. An effective AV pricing mechanism should be able to handle complex bidding strategies and the performance should not be influenced substantially based on the bidding strategies. To develop such a mechanism, we need to allocate appropriate AVs to the passengers and also determine the final service charges properly. We will explain their designs in Sections IV and V, respectively.

It should be noted that the proposed pricing mechanism is not limited to AVPTS. Actually, the mechanism can be generalized to other transportation systems where multiple vehicles and customers are involved. In this work, we adopt AVPTS to demonstrate the implementation and performance of the proposed mechanism. We will further investigate its application to other systems in our future research.

IV. Double Auction-based Allocation

As discussed above, we try to provide the customers with seat occupancies, which are leased by the AV operators through bidding. The control center acts as an auctioneer/broker in an auction. Similar to [9] and [10], we can model the request-AV allocation as an auction. In economics, a double auction refers to a process of trading goods where buyers and sellers submit their bids and their ask prices, respectively, to an auctioneer, who then configures a trading price to clear the market [12]. A combinatorial auction describes an auction involving combinations of discrete items rather than "individual items or continuous quantities" [14]. Recall that in the AVPTS pricing process, the customers compete with each other to get served while the AV operators race to provide service for profit. By considering combinations of seat occupancies as items to be traded, the allocation process is in fact double combinatorial auction-like.

A. Auction Setting

Consider that multiple transportation requests are submitted to the control center for settlement and each of them can bear one or more passengers. There are also multiple AV operators, each of which manages an independent fleet of AVs, competing to serve the requests. During this process, seat occupancies are regarded as the items to be traded. For instance, an AV with several available seats intends to "sell" these seat occupancies to potential customers, who may be originated from different transportation requests, subject to various transportation requirements. The control center acts as an auctioneer to perform the matching between the vehicles and the requests. In general, there are multiple "buyers" (i.e., customers) intended to to acquire multiple items (i.e., seat occupancies) from multiple "sellers" (i.e., AVs operators). This configuration analogizes a typical combinatorial double auction, and we call this the *AV auction* in the sequel.

Consider a set of requests $\mathcal{R} = \mathcal{R}^{\mathrm{S}} \cup \mathcal{R}^{\mathrm{N}} \cup \mathcal{R}^{\mathrm{P}}$ composed of three different kinds of requests, where $\mathcal{R}^{\mathrm{S}}$, $\mathcal{R}^{\mathrm{N}}$, and $\mathcal{R}^{\mathrm{P}}$ stand for the set of splittable, non-splittable, and private service requests, respectively. Each request $r \in \mathcal{R}$ is described by a 3-tuple $\langle q_r, c_r, \mathcal{I}_r \rangle$, where q_r is the number of seats required, c_r is the maximum expected service charge for the trip proposed by the customer, and $\mathcal{I}_r$ represents the other necessary information such as service type, pickup, dropoff locations, etc. for the admission control and scheduling purposes [6]. We assume that a request can either be completely served (by one or multiple AVs) or not served. In other words, a partially served request is not allowed. Since a request may end up to be served by multiple vehicles, the utility of the customer with respect to Request r, u_r, is given as follows:

$$u_r = \begin{cases} c_r - \sum_{k\in\mathcal{K}} p_{k,r}(\mathcal{S}_{k,r}) & r \text{ can be served} \\ 0 & \text{otherwise,} \end{cases} \quad (1)$$

where $\mathcal{S}_{k,r}$ is the set of seats of AV k allocated to serve r after the pricing process, and $p_{k,r}(\mathcal{S}_{k,r})$ is the corresponding service charge. $p_{k,r}$ is determined by the control center based on the auction result and it has a zero value when k is not assigned to serve r.

Let $\mathcal{K}$ be the set of AVs available for service in a particular pricing process. AV $k \in \mathcal{K}$ has total $\bar{q}_k$ seats with $\hat{q}_k \leq \bar{q}_k$ seats available. Similar to [9] and [10], we assume that all seats are homogeneous. Therefore, AV k is considered to provide different seat combinations[2] $\mathcal{Q}_k = \{\{1\}, \{1,2\}, \cdots, \{1,2,\cdots,\hat{q}_k\}\}$ for the customers to choose from, where $1, 2, \cdots, \hat{q}_k$ are the identifiers of seats [9]. For each k, its operator can calculate its operational cost of each combination $\mathcal{S} \in \mathcal{Q}_k$ to serve (part of) r through Step 3 in Fig. 1. This results in its valuation denoted by $v_k(\mathcal{S}, r)$. Based on $v_k(\mathcal{S}, r)$, the AV operator places a bid $b_k(\mathcal{S}, r)$ for k to serve r with $\mathcal{S}$. Assuming that all AV operators are rational, their reported bid values are no less than their actual operational costs (bid valuations), i.e., $b_k(\mathcal{S}, r) \geq v_k(\mathcal{S}, r) > 0$. The total utility of AV operator with respect to its governing AV k is defined by

$$\mu_k = \begin{cases} \sum_{r\in\mathcal{R}} p_{k,r}(\mathcal{S}_{k,r}) - \\ \quad \sum_{r\in\mathcal{R}} b_k(\mathcal{S}_{k,r}, r) & k \text{ serves any request,} \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

B. Winner Determination Problem

We determine the winners of the auction, i.e., matching between the requests and AVs, via optimization. We formulate an optimization problem by jointly considering all requests and bids, and the problem is called WDP. WDP is a widely adopted methodology to determine the winning bids in an auction – See [7], [11], [12] as examples. In this work, we follow the practice to determine the winner(s) of the auction via WDP. It is constructed by jointly considering all requests and bids as an optimization problem [12], [13], [15]. However, the structures of the utilities defined in (1) and (2) may not be helpful in formulating the problem. To overcome this, we transform them into quadratic forms, which can give an integer quadratic program (IQP). Moreover, we can also convert it to a tractable integer linear program (ILP). The details are illustrated as follows.

[2]In a combinatorial auction, each item is considered unique. Multiple items (seat occupancies in our case) are grouped into a combination (i.e., seat combination) for trading. Such combination is considered as a trading unit and this facilitates the buyers and sellers to trade multiple items simultaneously. Since all seats in a vehicle are assumed to be homogeneous, each vehicle can at most offer one combination of one seat, one combination of two seats, and so forth, up the combination of the vehicle capacity.

We introduce binary variables $x_k(\mathcal{S}, r) \in \{0, 1\}$ to indicate if k is allocated to serve r with seats $\mathcal{S}$. Then u_r in (1) and μ_k in (2) can be transformed into:

$$u_r = \big(1 - \prod_{k\in\mathcal{K}} \prod_{\mathcal{S}\in\mathcal{Q}_k} [1 - x_k(\mathcal{S}, r)]\big) c_r - \sum_{k\in\mathcal{K}} p_{k,r}(\mathcal{S}_{k,r}), \quad (3)$$

$$\mu_k = \sum_{r\in\mathcal{R}} p_{k,r}(\mathcal{S}_{k,r}) - \sum_{r\in\mathcal{R}} \sum_{\mathcal{S}\in\mathcal{Q}_k} b_k(\mathcal{S}, r) x_k(\mathcal{S}, r), \quad (4)$$

respectively. In (3), the term $\big(1 - \prod_{k\in\mathcal{K}} \prod_{\mathcal{S}\in\mathcal{Q}_k} [1 - x_k(\mathcal{S}, r)]\big)$ indicates whether r can be served in the auction. If not, all $x_k(\mathcal{S}, r)$ will turn zero resulting in this term equal to zero. On the other hand, if AV k can serve, we will have $x_k(\mathcal{S}, r) = 1$, making this term equal to one. Hence, the first term in (3) will be equal to c_r if r can be served, and zero otherwise. Therefore, (1) and (3) are equivalent. Similarly, we have $\sum_{\mathcal{S}\in\mathcal{Q}_k} b_k(\mathcal{S}, r) x_k(\mathcal{S}, r) = b_k(\mathcal{S}_{k,r}, r)$ when k serves r. Thus (2) and (4) are equivalent.

We formulate WDP to determine which items each seller should trade with each buyer. In the AV auction, we aim to maximize the total utility of all participants, including both AV operators and customers. Thus the objective is posed as follows:

$$\begin{aligned} &\sum_{r\in\mathcal{R}} u_r + \sum_{k\in\mathcal{K}} \mu_k \\ &= \sum_{r\in\mathcal{R}} \big[\big(1 - \prod_{k\in\mathcal{K}} \prod_{\mathcal{S}\in\mathcal{Q}_k} [1 - x_k(\mathcal{S}, r)]\big) c_r - \sum_{k\in\mathcal{K}} p_{k,r}(\mathcal{S}_{k,r})\big] \\ &+ \sum_{k\in\mathcal{K}} \big[\sum_{r\in\mathcal{R}} p_{k,r}(\mathcal{S}_{k,r}) - \sum_{r\in\mathcal{R}} \sum_{\mathcal{S}\in\mathcal{Q}_k} b_k(\mathcal{S}, r) x_k(\mathcal{S}, r)\big] \\ &= \sum_{r\in\mathcal{R}} \big(1 - \prod_{k\in\mathcal{K}} \prod_{\mathcal{S}\in\mathcal{Q}_k} [1 - x_k(\mathcal{S}, r)]\big) c_r \\ &- \sum_{r\in\mathcal{R}} \sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} b_k(\mathcal{S}, r) x_k(\mathcal{S}, r). \end{aligned} \quad (5)$$

Then WDP for the AV auction is formulated as:

Problem 1 (Winner Determination Problem)**.**

$$\begin{aligned} \text{maximize } & (5) \\ \text{subject to } & \sum_{r\in\mathcal{R}} \sum_{\mathcal{S}\in\mathcal{Q}_k} |\mathcal{S}| x_k(\mathcal{S}, r) \leq \hat{q}_k, \forall k \in \mathcal{K}, & (6a) \\ & \sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} |\mathcal{S}| x_k(\mathcal{S}, r) \geq q'_r, \forall r \in \mathcal{R}, & (6b) \\ & \sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} x_k(\mathcal{S}, r) \leq 1, \forall r \in \mathcal{R}^{\mathrm{N}} \cup \mathcal{R}^{\mathrm{P}}, & (6c) \\ & \sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}'_k} x_k(\mathcal{S}, r) = 0, \forall r \in \mathcal{R}^{\mathrm{P}}, & (6d) \\ & x_k(\mathcal{S}, r) \in \{0, 1\}, \forall r \in \mathcal{R}, k \in \mathcal{K}, \mathcal{S} \in \mathcal{Q}_k, & (6e) \end{aligned}$$

where $|\cdot|$ stands for the cardinality of a set, $q'_r = q_r\big(1 - \prod_{k\in\mathcal{K}} \prod_{\mathcal{S}\in\mathcal{Q}_k} [1 - x_k(\mathcal{S}, r)]\big)$ is the actual number of seats required by r, and $\mathcal{Q}'_k = \mathcal{Q}_k \setminus \{\mathcal{S} \in \mathcal{Q}_k | |\mathcal{S}| = \bar{q}_k\}$. (6a) guarantees that each AV k does not offer more seats than the available, i.e., $\hat{q}_k$. (6b) ensures that if Request r is served, there will be sufficient seats offered to accommodate the passengers associated to r. (6c) states that the total number of AVs employed to serve a non-splittable or private service request must be no greater than one, which is imposed by the nature of the service types. (6d) excludes those bids associated to non-empty vehicles for the private service.

Note that Problem 1 is an IQP. Solving an IQP can be computationally expensive even with a moderate problem size. To reduce the complexity, we introduce the variable $y_r \in \{0,1\}$ to replace the quadratic terms in Problem (1) as

$$y_r = 1 - \prod_{k\in\mathcal{K}} \prod_{\mathcal{S}\in\mathcal{Q}_k} [1 - x_k(\mathcal{S}, r)], \forall r \in \mathcal{R}. \tag{7}$$

y_r also indicates whether Request r can be served by any AV in the system. In other words, y_r is one if and only if there exists a $\{k, \mathcal{S}\} \in \mathcal{K} \times \mathcal{Q}_k$ such that $x_k(\mathcal{S}, r) = 1$.

By introducing (7) to Problem (1), the quadratic terms in (5) and (6b) can be replaced by y_r. However, (7) should also be included as a new constraint to confine the relationship between y_r and $x_k(\mathcal{S}, r)$. As (7) is quadratic, which is computationally expensive, we replace it with the following linear constraint [16]:

$$y_r \le \sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} x_k(\mathcal{S}, r). \tag{8}$$

If any $x_k(\mathcal{S}, r)$ is set to one, then we have $\sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} x_k(\mathcal{S}, r) \ge 1$. In Problem 1 we aim to maximize (5), which now becomes $\sum_{r\in\mathcal{R}} c_r y_r - \sum_{r\in\mathcal{R},k\in\mathcal{K},\mathcal{S}\in\mathcal{Q}_k} b_k(\mathcal{S}, r)x_k(\mathcal{S}, r)$ after introducing y_r. Thus $c_r y_r$ should be maximized, and the binary variable y_r should attain one by maximization. On the other hand, if $\sum_{k\in\mathcal{K},\mathcal{S}\in\mathcal{Q}_k} x_k(\mathcal{S}, r) = 0$ due to all related $x_k(\mathcal{S}, r)$ are zero, y_r will be confined to zero. Therefore, constraints (7) and (8) are interchangeable with respect to Problem 1. By introducing (7) and including (8), we can transform Problem 1 into an ILP:

Problem 2 (Transformed Winner Determination Problem)**.**

$$\text{maximize} \sum_{r\in\mathcal{R}} c_r y_r - \sum_{r\in\mathcal{R}} \sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} b_k(\mathcal{S}, r)x_k(\mathcal{S}, r) \tag{9a}$$

$$\text{subject to} \sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} |\mathcal{S}|x_k(\mathcal{S}, r) \ge q_r y_r, \forall r \in \mathcal{R}, \tag{9b}$$

$$x_k(\mathcal{S}, r), y_r \in \{0,1\}, \forall r \in \mathcal{R}, k \in \mathcal{K}, \mathcal{S} \in \mathcal{Q}_k, \tag{9c}$$

(6a),(6c), (6d), and (8).

To simplify the notation, we call Problem 2 WDP in the sequel whenever this is no confusion.

As an ILP, Problem 2 can also be hard to solve when the problem size gets large. To overcome this, we can investigate the properties of Problem 2 to reduce its feasible region further for potential computational speedup.

Lemma 1. *Each AV can attain at most one winning bid for each request if all bids $b_k(\mathcal{S}, r)$ are concave with respect to the number of seats $|\mathcal{S}|$.*

Proof. We need to show that $\sum_{\mathcal{S}\in\mathcal{Q}_k} x_k(\mathcal{S}, r) \le 1, \forall r \in \mathcal{R}, k \in \mathcal{K}$ if all bids $b_k(\mathcal{S}, r)$ are concave. For non-splittable and private services, from (6c), we have $\sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} x_k(\mathcal{S}, r) \le 1$, which implies $\sum_{\mathcal{S}\in\mathcal{Q}_k} x_k(\mathcal{S}, r) \le 1$. Thus the lemma holds. For splittable service, we prove the result by contradiction. Assume $\sum_{\mathcal{S}\in\mathcal{Q}_k} x_k(\mathcal{S}, r) = 2$, i.e., an AV serves a splittable service request with two bids with seats $\mathcal{S}'$ and $\mathcal{S}''$. As the bid values are concave with respect to $|\mathcal{S}|$, there always exists seats $\mathcal{S}^*$ such that $|\mathcal{S}^*| = |\mathcal{S}'| + |\mathcal{S}''|$, and $b_k(\mathcal{S}^*, r) < b_k(\mathcal{S}', r) + b_k(\mathcal{S}'', r)$. In this case, y_r is of value equal to one, thus the first summation in (9a) is constant. The auctioneer should try to minimize $\sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} b_k(\mathcal{S}, r)x_k(\mathcal{S}, r)$ in Problem 2, which makes it favors $\mathcal{S}^*$ over $\mathcal{S}'$ and $\mathcal{S}''$. This contradicts the assumption that k wins with $\mathcal{S}'$ and $\mathcal{S}''$. $\sum_{\mathcal{S}\in\mathcal{Q}_k} x_k(\mathcal{S}, r) > 2$ can be proved using the same pattern. ■

Although adopting Lemma 1 would introduce extra constraints to Problem 2, the feasible region is reduced. This may result in a decrease in computation time to solve the problem. We will verify this in Section VI.

Lemma 2. *For the splittable and non-splittable service requests, if $b_k(\mathcal{S}, r)$ is increasing with respect to the number of seats $|\mathcal{S}|$, no extra seats will be offered to r.*

Proof. Mathematically, this is equivalent to

$$\sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} |\mathcal{S}|x_k(\mathcal{S}, r) = q_r y_r, \forall r \in \mathcal{R}^{\mathrm{S}} \cup \mathcal{R}^{\mathrm{N}}. \tag{10}$$

If r is not served, both sides of (10) will become zero. For the case that r is served, we prove this lemma by contradiction. To avoid ambiguity, we define $\mathrm{B}_k(\mathcal{S}, r)$ as the bid associated with bid value $b_k(\mathcal{S}, r)$. Assume $\sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} |\mathcal{S}|x_k(\mathcal{S}, r) = q_r y_r + 1$, and $\mathcal{W} = \{\mathrm{B}_k(\mathcal{S}, r)|x_k(\mathcal{S}, r) = 1\}$ is the set of winning bids for serving r. For any $\mathrm{B}_{k'}(\mathcal{S}', r) \in \mathcal{W}$, we have

$$|\mathcal{S}'| + \sum_{\mathrm{B}_k(\mathcal{S},r)\in\mathcal{W}'} |\mathcal{S}| = q_r y_r + 1, \tag{11}$$

where $\mathcal{W}' = \mathcal{W} \setminus \mathrm{B}_{k'}(\mathcal{S}', r)$. There exist two cases:

1) $|\mathcal{S}'| > 1$: The auctioneer can select another bid $\mathrm{B}_{k'}(\mathcal{S}^*, r)$, such that $|\mathcal{S}'| = |\mathcal{S}^*| + 1$. As k' can provide at least $|\mathcal{S}'|$ seats, this bid always exists. Therefore, we have

$$|\mathcal{S}'| + \sum_{\mathrm{B}_k(\mathcal{S},r)\in\mathcal{W}'} |\mathcal{S}| > |\mathcal{S}^*| + \sum_{\mathrm{B}_k(\mathcal{S},r)\in\mathcal{W}^*} |\mathcal{S}| = q_r y_r,$$

where $\mathcal{W}^* = \mathcal{W} \setminus \mathrm{B}_{k'}(\mathcal{S}^*, r)$. By Lemma 1, the auctioneer should minimize $\sum_{k\in\mathcal{K}} \sum_{\mathcal{S}\in\mathcal{Q}_k} b_k(\mathcal{S}, r)x_k(\mathcal{S}, r)$ since $b_{k'}(\mathcal{S}^*, r) < b_{k'}(\mathcal{S}', r)$. Therefore, it favors $\mathcal{S}^*$ over $\mathcal{S}'$. This contradicts the assumption that $\mathrm{B}_{k'}(\mathcal{S}', r)$ is a winning bid.
2) $|\mathcal{S}'| = 1$: The auctioneer prefers another set of winning bids $\mathcal{W}' = \mathcal{W} \setminus \mathrm{B}_{k'}(\mathcal{S}', r)$. By removing $\mathrm{B}_{k'}(\mathcal{S}', r)$, we still have $\sum_{\mathrm{B}_k(\mathcal{S},r)\in\mathcal{W}'} |\mathcal{S}| = q_r y_r$. So enough seats are provided for r and the total service charge is reduced. This contradicts the assumption that $\mathrm{B}_{k'}(\mathcal{S}', r)$ is a winning bid.

The proof of $\sum_{k\in\mathcal{K}}\sum_{\mathcal{S}\in\mathcal{Q}_k}|\mathcal{S}|x_k(\mathcal{S},r)=q_r y_r+n, n>1$ can be done similarly[3]. ∎

Lemma 2 can further reduce the feasible region of WDP by introducing (10) to the problem. We will also show how Lemma 2 helps improve the computational time in Section VI.

C. Discussion

In this work we present a generalized WDP for the proposed pricing mechanism of AVPTS. It is worth noting that the formulated optimization problem can be modified to satisfy specific formulation and/or performance requirements for particular transportation systems. For instance, in Appendix A we discuss a possible solution to incorporate passenger waiting time into the optimization problem. In addition, the solution space of WDP serving a large city can be huge rendering long computational time. We may change the investigating time span and service area to influence of the solution space, leading to performance improvement. Through pre-processing, we can also cluster the vehicles and requests into groups based on their temporal and spatial conditions. In this way, the original WDP can be divided into multiple sub-problems and thus a notably reduced computation time can be expected. How to employ such methods to improve the problem solving performance will be investigated in our future research.

Moreover, be noted that the original design of AVPTS actually does not consider any security issues, which may happen during the pricing and vehicle-assignment processes. So the system is vulnerable to security attacks. For example, intruders may adversely forbid specific vehicles from service. However, since the proposed pricing mechanism is conducted in batches as demonstrated in Fig. 2, certain security issue can be resolved by some "fallback" approaches. Assume that some vehicle operators are compromised in a particular batch. One may immediately re-conduct the auction with the compromised vehicles removed and backup vehicles included. If the time to process each batch is short enough or the customer waiting time is not an important issue, one may also discard the result of the current batch and skip to the next (so two batches are processed altogether). The above solutions actually require AVPTS to provide certain backup vehicles for security measures, which is a possible extension for AVPTS enhancement.

V. Duality-based Service Charge Determination

We need to determine the service charges to complete the pricing process. In this section, by investigating the duality of Problem 2, we determine the shadow prices to be the service charges.

A. Primal and Dual Problems

We call Problem 2 the primal problem in the following. As will be illustrated in Section VI, mainly the constraints defined by (6a) make the problem more computationally expensive. Therefore, we consider to relax (6a) to construct the dual problem.

[3]This lemma cannot be applied to private service requests as each AV has at most one feasible bid, which consumes all seats of the vehicle. Hence the alternative bid $\mathrm{B}_{k'}(\mathcal{S}^*,r)$ does not exist.

Let $\Lambda=[\lambda_k]^{\intercal}_{k\in\mathcal{K}}\geq 0$ be the vector of Lagrangian multipliers corresponding to (6a), and let Ψ be the solution space of Problem 2 without (6a). We consider the following relaxed problem:

$$Z(\Lambda)\triangleq\underset{\{x_k(\mathcal{S},r),y_r\}\in\Psi}{\text{maximize}}\; c_r y_r-\sum_{r\in\mathcal{R}}\sum_{k\in\mathcal{K}}\sum_{\mathcal{S}\in\mathcal{Q}_k}b_k(\mathcal{S},r)x_k(\mathcal{S},r) -\sum_{k\in\mathcal{K}}\lambda_k\Big(\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}|\mathcal{S}|x_k(\mathcal{S},r)-\hat{q}_k\Big). \tag{12}$$

Let $\{x^*_k(\mathcal{S},r),y^*_r\}_{r\in\mathcal{R},k\in\mathcal{K},\mathcal{S}\in\mathcal{Q}_k}\in\Psi$ be an optimal solution of Problem 2 and Z_{IP} be the corresponding optimal objective function value. As $\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}|\mathcal{S}|x^*_k(\mathcal{S},r)-\hat{q}_k\leq 0$, we have

$$\begin{aligned} Z_{\text{IP}}&\triangleq c_r y^*_r-\sum_{r\in\mathcal{R}}\sum_{k\in\mathcal{K}}\sum_{\mathcal{S}\in\mathcal{Q}_k}b_k(\mathcal{S},r)x^*_k(\mathcal{S},r)\\ &\leq c_r y^*_r-\sum_{r\in\mathcal{R}}\sum_{k\in\mathcal{K}}\sum_{\mathcal{S}\in\mathcal{Q}_k}b_k(\mathcal{S},r)x^*_k(\mathcal{S},r)\\ &-\sum_{k\in\mathcal{K}}\lambda_k\Big(\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}|\mathcal{S}|x^*_k(\mathcal{S},r)-\hat{q}_k\Big).\end{aligned}$$

Therefore, $Z(\Lambda)\geq Z_{\text{IP}}$ for any arbitrary Λ and Problem (12) provides an upper bound for Problem 2.

To find the optimal Λ^* that yields the tightest bound, we consider the following problem:

Problem 3 (Lagrangian Dual Problem)**.**

$$\text{minimize } Z(\Lambda) \text{ subject to } \Lambda\geq 0. \tag{13}$$

Let Z_{D} be the optimal value of (13). $Z(\Lambda)$ is concave and piecewise-linear. It is clear that the weak duality theorem holds: $Z_{\text{D}}\geq Z_{\text{IP}}$. As Ψ is constituted by binary variables, the solution space is finite. Therefore, many techniques can be employed to efficiently solve Problem 3, whose optimal value provides an upper bound for Problem 2.

B. Service Charge Determination

In fact, we can interpret the Lagrangian multipliers λ_k as shadow prices. In this context, they represent the service charges per seat imposed by $\mathcal{K}$. For the dual problem, we can specify the sub-gradient as follows:

$$g'_{\lambda_k}=\frac{\partial Z(\Lambda)}{\partial\lambda_k}=\hat{q}_k-\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}|\mathcal{S}|x_k(\mathcal{S},r). \tag{14}$$

We first consider an arbitrary λ_k, denoted by λ'_k. When $g'_{\lambda_k}|_{\lambda_k=\lambda'_k}$ is greater than zero, $Z(\Lambda)$ increases with λ_k at λ'_k. So the optimal λ_k, denoted by λ^*_k, should be smaller than λ'_k since $Z(\Lambda)$ is convex [17]. At the same time, $g'_{\lambda_k}|_{\lambda_k=\lambda'_k}>0$ also implies that $\hat{q}_k>\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}|\mathcal{S}|x_k(\mathcal{S},r)$. In other words, the capacity of k is larger than the number of seats required. This suggests a smaller service charge for each seat. On the other hand, when the subgradient at λ'_k is negative, λ^*_kshould be larger than λ'_k. The negative g'_{λ_k} also implies that the capacity is smaller than the number of seats required, leading to a larger service charge for each seat. Consequently,

Λ^* reflects the optimal service charge per seat, which also results in the smallest duality gap between the primal and dual problem.

Lagrangian multipliers have been employed to construct auction prices in some previous work (see [11] for example). In this paper, we follow a similar methodology to establish a service charging rule. As Λ^* represents the final service charge per seat of the AVs, the total service charge by each AV k can be calculated accordingly with the winning bid results from Problem 2 as $\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}\lambda_k^*|\mathcal{S}|x_k(\mathcal{S},r)$. However, the minimum service charge proposed by k, i.e., $\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}b_k(\mathcal{S},r)x_k(\mathcal{S},r)$, is less than or equal to the income $\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}\lambda_k^*|\mathcal{S}|x_k(\mathcal{S},r)$ decided by the auction. To determine the value for each $p_k(\mathcal{S},r)$ such that the summation of all service charges of its serving requests $\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}p_k(\mathcal{S},r)x_k(\mathcal{S},r)$ equals to the decided income, one can first set $p_k(\mathcal{S},r)$ to their corresponding $b_k(\mathcal{S},r)$ values, and then gradually increase them to increase $\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}p_k(\mathcal{S},r)x_k(\mathcal{S},r)$ to $\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}\lambda_k^*|\mathcal{S}|x_k(\mathcal{S},r)$. In this work, all requests served by a particular AV will have the same degree of increment, denoted by C_k. As

$$\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}\lambda_k^*|\mathcal{S}|x_k(\mathcal{S},r)=\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}C_kb_k(\mathcal{S},r)x_k(\mathcal{S},r),$$

we can set

$$C_k=\frac{\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}\lambda_k^*|\mathcal{S}|x_k(\mathcal{S},r)}{\sum_{r\in\mathcal{R}}\sum_{\mathcal{S}\in\mathcal{Q}_k}b_k(\mathcal{S},r)x_k(\mathcal{S},r)}, \tag{15}$$

and the final service charge is determined as:

$$p_k(\mathcal{S},r)=C_kb_k(\mathcal{S},r)x_k(\mathcal{S},r). \tag{16}$$

For example, AV k wins an auction for providing two seats $\mathcal{S}$ to Request r and three seats $\mathcal{S}'$ to Request r' with bid values $b_k(\mathcal{S},r)=4$ and $b_k(\mathcal{S}',r')=5$, respectively. The suggested charge per seat for k is $\lambda_k^*=2.5$. Then we have $C_k=[\lambda_k^*(|\mathcal{S}|+|\mathcal{S}'|)]/[b_k(\mathcal{S},r)+b_k(\mathcal{S}',r')]\approx 1.389$. The service charges of r and r' are $1.389\times b_k(\mathcal{S},r)=5.556$ and $1.389\times b_k(\mathcal{S}',r')=6.945$, respectively. In this case, the final service charge is calculated based on the bid value and λ_k^*, and the total income by k accords with the auction result.

VI. Performance Evaluation

We conduct simulations to evaluate the performance of our proposed pricing mechanism in five different perspectives. We first examine the achievable social welfare under different scales of the system. Then we investigate the computational time required to complete a pricing mechanism. Next we study the efficacy of the proposed service charge rule to advocate market competition. After that, we investigate how likely passengers can be served under different problem scales. Last but not least, we inspect the effect of different request handling methods by looking into sequential and batch processing of the service requests.

Random test cases are generated to simulate the system of different scales. We consider different values for $|\mathcal{R}|\in\{5,10,20,50,100\}$, and for $|\mathcal{K}|\in\{10,20,50,100,200,500,1000\}$. For each combination of $|\mathcal{R}|$ and $|\mathcal{K}|$, we produce 25 random cases respectively and thus there are 875 test cases in total. For each test case, $|\mathcal{R}|$ random customers are created, in which each customer requires a random number of seats in the range of $[1,8]$. The distance between pickup and dropoff locations d_r is randomly chosen in the range of $[1,3]$ units. Unless otherwise stated, 60%, 30%, and 10% of the requests belong to the splittable, non-splittable, and private services, respectively. We determine the maximum expected service charge for each customer by multiplying the distance d_r with number of passengers in the request q_r and a random factor drawn from a normal distribution $\mathcal{N}(0,0.05^2)$. For AV $k\in\mathcal{K}$, $\bar{q}_k$ is randomly selected in $[4,8]$ and $\hat{q}_k$ in $[1,\bar{q}_k]$. Similarly, the bid $b_k(\mathcal{S},r)$ is calculated by multiplying the distance d_r with the number of passengers and a random factor drawn from $\mathcal{N}(0,0.05^2)$. The result is further multiplied by 0.9 so as to make most bid values smaller than their corresponding customer maximum expected service charge.

All tests are performed on a computer with an Intel Core i7-3770 CPU at 3.40GHz and 16 GB RAM. The test code is developed in Python 3, and all optimization problems are solved with Gurobi [18].

A. Social Welfare

We first evaluate the achievable social welfare under different scales of AVPTS. The average computed results of social welfare are presented in Table I. We also provide the analytical upper bounds discussed in Section V for reference and they allow us to see the impact of (6a) on the complexity of solving the WDP. In Table I, "AV Auction" and "Dual bound" label the results computed by our proposed AV Auction and the bounds achieved through the dual, respectively. The "Utility" columns indicate the total utilities (or social welfare) of the computed AV Auction results.

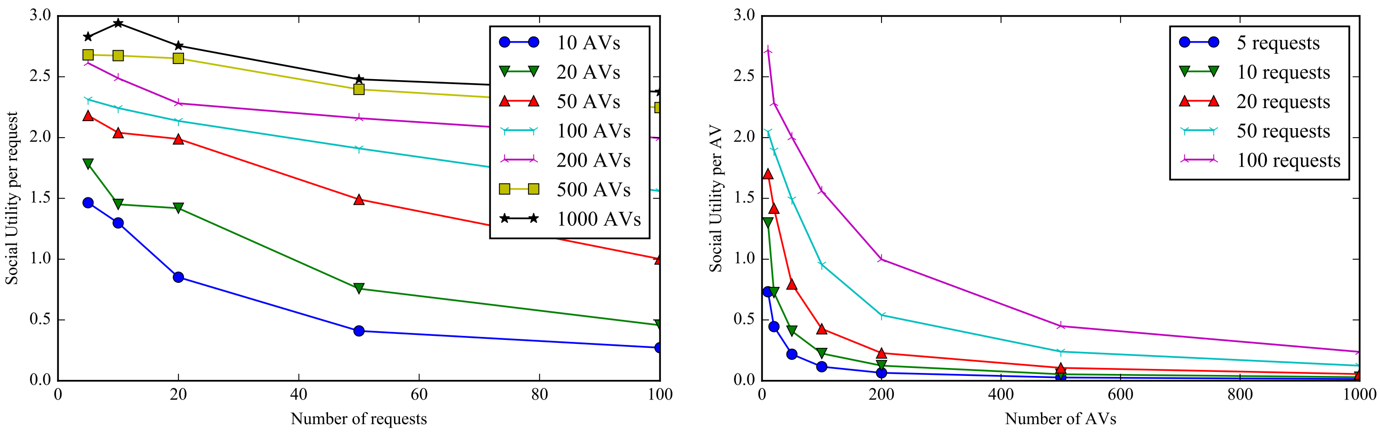
It can be observed that the total utility increases with the numbers of AVs and requests. This confirms our intuition that the social welfare should benefit from a larger market with more AV operators and/or customers involved.

More specifically, the market size grows with both the sizes of the request pool and the AV fleet. To better understand the relationship between market size and achievable social welfare, we investigate the change of average utility from the perspectives of requests and AVs separately. The results are illustrated in Fig. 3, in which each data point corresponds to the average of the results of 25 test cases. Fig. 3a shows how the average utility contributed by each request changes with the number of available requests for several fixed AV fleet sizes. In general the average utility decreases in the presence of more requests. The trend is similar with different fleet size but higher average utility can be achieved with more AVs. More requests result in severer competition and thus the individual utility is reduced. However, as previously shown in Table I, the total utility is still improved with more customers/requests. The influence of the AV fleet size can be more easily depicted in Fig. 3b.

Fig. 3b illustrates how the average utility contributed by each AV changes with the number of AVs for fixed numbers of requests. In general the average utility decreases with the

TABLE I
PERFORMANCE EVALUATION OF THE PROPOSED PRICING MECHANISM

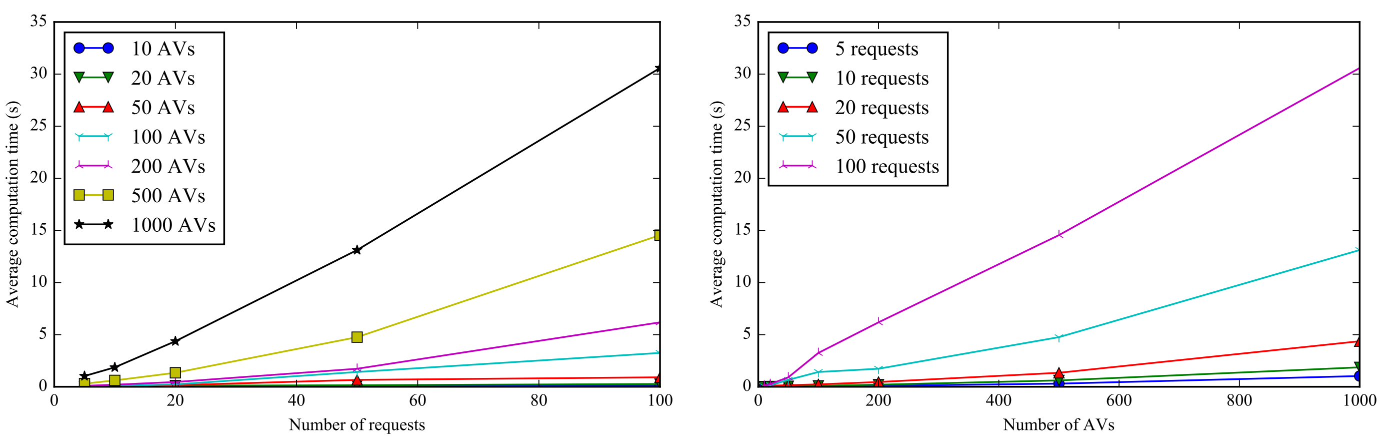
Number of AVs		Number of Requests									
		5		10		20		50		100	
		Utility	Time (s)	Utility	Time (s)	Utility	Time (s)	Utility	Time (s)	Utility	Time (s)
10	AV Auction	7.32e+00	7.84e-03	1.30e+01	1.63e-02	1.70e+01	3.56e-02	2.05e+01	5.44e-02	2.72e+01	1.16e-01
	Dual bound	7.41e+00	3.37e-03	1.32e+01	5.22e-03	1.73e+01	1.33e-02	2.07e+01	3.15e-02	2.75e+01	8.38e-02
20	AV Auction	8.91e+00	1.01e-02	1.45e+01	2.80e-02	2.84e+01	1.03e-01	3.79e+01	1.48e-01	4.57e+01	2.66e-01
	Dual bound	8.95e+00	4.67e-03	1.46e+01	9.29e-03	2.86e+01	2.16e-02	3.81e+01	6.16e-02	4.59e+01	1.46e-01
50	AV Auction	1.09e+01	1.76e-02	2.04e+01	5.06e-02	3.98e+01	1.51e-01	7.46e+01	6.61e-01	1.00e+02	9.08e-01
	Dual bound	1.09e+01	1.06e-02	2.05e+01	2.33e-02	3.99e+01	5.05e-02	7.48e+01	1.56e-01	1.00e+02	3.90e-01
100	AV Auction	1.16e+01	3.87e-02	2.24e+01	8.82e-02	4.27e+01	2.29e-01	9.55e+01	1.42e+00	1.56e+02	3.25e+00
	Dual bound	1.16e+01	2.57e-02	2.25e+01	5.07e-02	4.28e+01	1.09e-01	9.57e+01	3.45e-01	1.56e+02	8.76e-01
200	AV Auction	1.31e+01	9.09e-02	2.49e+01	1.98e-01	4.56e+01	4.56e-01	1.08e+02	1.73e+00	2.00e+02	6.18e+00
	Dual bound	1.31e+01	6.55e-02	2.49e+01	1.23e-01	4.57e+01	2.57e-01	1.08e+02	7.98e-01	2.00e+02	2.14e+00
500	AV Auction	1.34e+01	3.01e-01	2.67e+01	6.11e-01	5.30e+01	1.34e+00	1.20e+02	4.76e+00	2.25e+02	1.45e+01
	Dual bound	1.34e+01	2.51e-01	2.68e+01	5.03e-01	5.31e+01	1.04e+00	1.20e+02	3.28e+00	2.25e+02	7.66e+00
1000	AV Auction	1.41e+01	1.03e+00	2.94e+01	1.86e+00	5.51e+01	4.37e+00	1.24e+02	1.31e+01	2.38e+02	3.06e+01
	Dual bound	1.42e+01	9.19e-01	2.95e+01	1.60e+00	5.52e+01	3.75e+00	1.24e+02	1.02e+01	2.38e+02	2.18e+01



(a) Average utility per request on different number of requests

(b) Average utility per AV on different number of AVs

Fig. 3. Change of average utility.



(a) Computational time change on different number of requests

(b) Computation time change on different number of AVs

Fig. 4. Change of average computation time.

TABLE II
IMPACT OF CONSTRAINT INTRODUCED BY LEMMAS 1 AND 2 ON COMPUTATION TIME

	Problem 2 with Lemma 1			Problem 2 without Lemma 1		
	Avg.	Min.	Max.	Avg.	Min.	Max.
Problem 2 with Lemma 2	100%	100%	100%	103.74%	100.59%	112.02%
Problem 2 without Lemma 2	102.15%	100.63%	106.85%	105.43%	100.91%	115.67%

number of vehicles. The trend is similar with different number of requests but higher average utility can be obtained in the presence of more requests. Increasing competition among vehicles reduces individual AV utility. However, with more AVs involved in the auction, customers have more transit options and thus the total utility can also be improved.

B. Computational Time

We investigate the computational time required to determine the result of an AV auction. The times to generate the corresponding results are also provided in Table I. In general, the computational time increases with the market size since the formulated WDP is getting bigger with more constraints. In addition, the computational time required for dual bound is significantly smaller than that for the AV Auction. This validates our claim that the complexity of the WDP is mainly stemmed from Constraint (6a), since the major difference between the original and the dual problems is on the relaxation of (6a).

Fig. 4 illustrates the change of computational time in the perspectives of requests and vehicles separately. In Fig. 4a, for a fixed fleet size, the average computational time grows roughly linearly with the number of requests. Similarly, as shown in Fig. 4b, for a fixed request pool size, the computational time also grows roughly linearly with the number of AVs. However, as shown in Table I, when more customers and AV operators participate in the AV auction, the computational time will increase rapidly. In spite of that, our largest test case, which contains 1000 AVs and 100 requests, requires merely around 30 seconds to be solved on an ordinary computer. This suggests that the proposed AV pricing mechanism is highly practical.

Recall that Lemmas 1 and 2 presented in Section IV introduce additional constraints to Problem 2 resulting in different formulations. These constraints can reduce the feasible region of the problem. Here we also verify that both lemmas can contribute to computational speedup. To do this, we examine the test cases and compute the results based on three variants of Problem 2: (i) Problem 2 with Lemma 1, (ii) Problem 2 with Lemma 2, and (iii) Problem 2 with both the lemmas. For our test cases, all formulations are equivalent resulting in the same optimal values. So we focus on the computational time. We set Formulation (iii) as the benchmark and the relative average, minimum, and maximum computational times for solving the test cases are presented in Table II. From the table, it can be concluded that both lemmas can reduce the computational time as analyzed in Section IV. In the worst case scenarios, i.e., the "Min." columns in Table II, the computational time can still be reduced by introducing either lemma. The differences in computational time become more obvious in the best-case scenarios, i.e., the "Max." columns. It is worth mentioning that both lemmas can be incorporated into Problem 2 at no additional cost. Therefore, enforcing Lemmas 1 and 2 is generally beneficial.

C. Service Charge

We examine the impact of AVPTS system size on the service charge. Fig. 5 shows the change of average service charge per seat with respect to the numbers of available requests and AVs, respectively. Fig. 5a depicts that the service charge per seat increases with the number of requests when only a small number of AVs are available. When the fleet size grows (e.g., of 200 AVs), the service charge per seat becomes insensitive to the number of requests. This is due to the fact that competition among customers exists when there are not enough AVs to accommodate the requests and the competition leads to an increase in average service charge. On the other hand, when there are sufficient AVs available, all the requests can be entertained and competition is not likely to happen. Note that the average service charge experiences little fluctuation with 10 requests, which are caused by the randomness of the generated cases. We can also come up with a similar conclusion for the competitions among AV operators from Fig. 5b. In fact, fewer requests result in a smaller AVPTS market. When there are fewer buyers (customers), the sellers (AV operators) need to reduce the prices in order to sell their seat occupancies. However, with more requests (customers), the service charge can be slightly increased without losing in the auction for rationality reasons. This explains why the average service charge per seat for five requests are slightly lower than that for more requests.

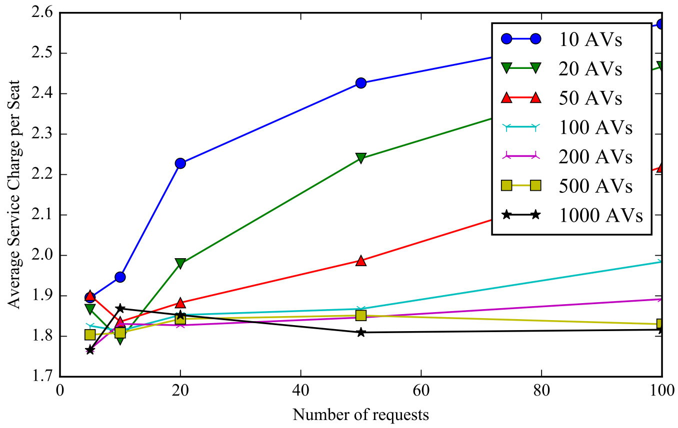
D. Successfully Served Customers

Here we explore how many passengers can be successfully served under different problem sizes. We express the results in terms of percentage which indicates how many passengers can belong to the successfully served requests among the passengers of all possible requests. The averaged percentage results are depicted in Fig. 6. From Fig. 6a we can observe that the percentage of served passengers decreases with number of requests for fixed AV fleet sizes. This is due to the fact that with a constant number of AVs, more requests lead to severer competition among customers. So the AVs are encouraged to serve those requests with potentially higher profit with higher priority. In addition, when the number of AVs is large enough, e.g., 500 or 1000 AVs for serving 100 requests, almost all requests can be served resulting in a minuscule percentage decrease.

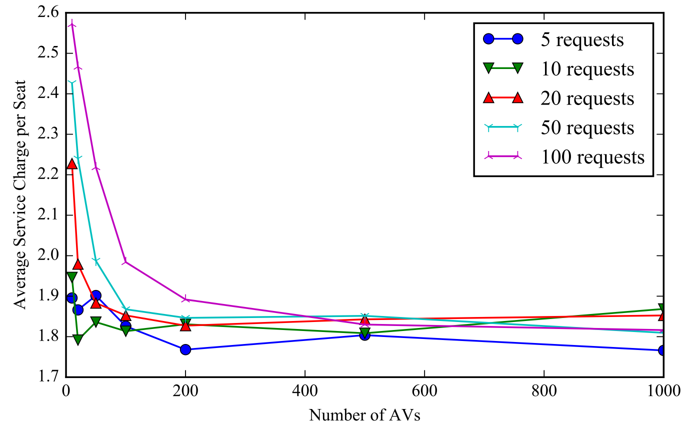
To be complete, we also present the percentage change with respect to different number of AVs in Fig. 6b, which demonstrates a similar trend. With an increase of AV fleet size, the percentage of served passengers also increases. This is consistent with our previous explanation for Fig. 6a.

E. Service Request Processing

In an AVPTS pricing process, there can be two modes to process all service requests $\mathcal{R}$. We can either handling all requests in a batch or one-by-one sequentially based on their arrival times. While the mechanism proposed in this work targets batch processing, we can also consider that each batch consists of only one request and apply our proposed mechanism $|\mathcal{R}|$ times for processing all the requests (called single-request AV auction). For sequential processing, we adopt the VCG mechanism proposed in [9] for the illustrative

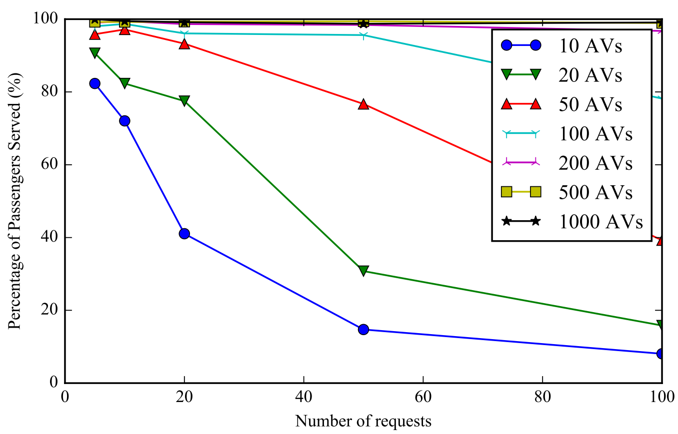


(a) Average service charge per seat on different number of requests

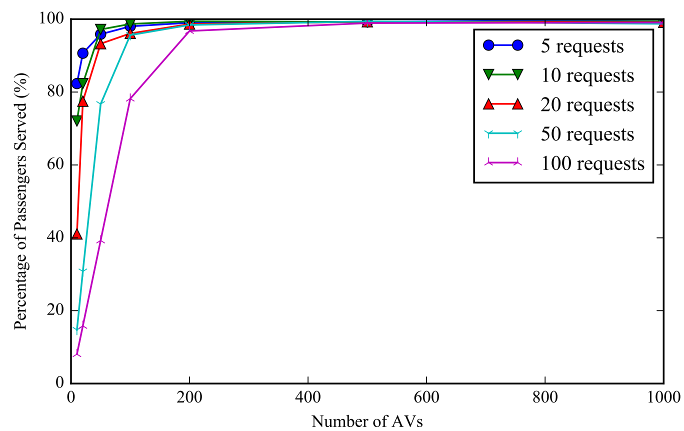


(b) Average service charge per seat on different number of AVs

Fig. 5. Change of average service charge.



(a) Percentage of served passengers on different number of requests



(b) Percentage of served passengers on different number of AVs

Fig. 6. Change of served passenger percentage.

TABLE III
COMPARISON OF THE PROPOSED PRICING MECHANISM WITH VCG AND SINGLE REQUEST MECHANISMS

Number of AVs		Number of Requests									
		5		10		20		50		100	
		Utility	Time	Utility	Time	Utility	Time	Utility	Time	Utility	Time
10	VCG	56.1%	162.3%	60.5%	109.7%	50.5%	71.7%	38.6%	78.5%	33.6%	67.4%
	WDP-1	65.4%	118.3%	61.7%	103.5%	51.9%	116.1%	39.1%	353.2%	34.3%	602.4%
20	VCG	51.3%	159.4%	53.0%	101.1%	55.6%	44.5%	43.7%	49.2%	40.2%	43.6%
	WDP-1	58.7%	111.7%	55.5%	83.6%	58.1%	53.1%	44.7%	143.5%	41.4%	276.9%
50	VCG	48.4%	160.6%	48.2%	121.5%	50.8%	67.0%	56.4%	31.3%	44.3%	31.3%
	WDP-1	57.9%	119.5%	56.7%	94.5%	57.9%	63.3%	60.5%	48.9%	46.2%	98.5%
100	VCG	46.0%	167.1%	45.4%	145.5%	49.9%	94.2%	52.3%	35.8%	55.7%	24.2%
	WDP-1	55.1%	105.1%	55.7%	102.4%	57.1%	73.8%	58.2%	37.0%	58.6%	39.5%
200	VCG	40.9%	156.4%	46.3%	144.2%	47.0%	113.7%	50.4%	71.2%	52.7%	35.4%
	WDP-1	51.0%	101.6%	53.6%	97.8%	55.1%	79.3%	57.4%	59.3%	60.1%	38.8%
500	VCG	33.8%	159.4%	39.3%	158.9%	41.8%	144.5%	46.7%	104.0%	49.4%	64.1%
	WDP-1	50.3%	97.9%	51.6%	96.9%	52.7%	91.0%	55.0%	70.2%	58.0%	49.1%
1000	VCG	42.4%	148.7%	45.0%	138.3%	41.4%	133.4%	43.7%	106.4%	43.9%	85.6%
	WDP-1	46.1%	84.4%	49.7%	84.0%	49.8%	78.1%	51.6%	66.5%	54.8%	58.2%

purpose. Therefore, we can compare three request processing methods.

We consider all the 875 random cases again. The simulation results are presented in Table III. In the table, we set the batch processing method as the basis and we list the relative performance of the sequential VCG mechanism (VCG) and the single-request AV Auction (WDP-1) with respect to the batch processing method. For example, if batch processing can generate an average social welfare at 100 and VCG can make 80, the corresponding entry is shown as 80%. From the table, it is clear that both sequential methods produce results with utility much worse than the batch processing. The performance gap increases with the market size. This shows that the arrival sequence can influence the social welfare and the batch processing can get rid of this influence and improve the performance. When only the two sequential processing methods are compared, WDP-1 still outperforms VCG in terms of utility. This observation is concurred with the literature that VCG cannot guarantee maximum customer utilities [12], [19], [20].

When it comes to computational time, no single method has advantages over all circumstances. For small systems, the batch-processing AV auction requires less computational time than the sequential counterparts but it takes longer then problem size gets large. It is due to the increased complexity of WDP when the number of AVs or requests becomes large. Despite this, the maximum simulation time required for the largest system, as illustrated in Section VI-A, is around 30 seconds, which is insignificant in practice. Therefore the AVPTS can still provide a near real-time customer experience.

VII. Conclusion

In a large multi-tenant AVPTS, the pricing process is essential to allocate the AVs to serve the service requests and to settle customer service charges. Previous related studies can only handle one request in each execution, which may suppress the social welfare of the system. In this paper, we construct a new pricing mechanism, which can handle multiple requests in a batch, resulting in better service plans. We first illustrate the complete mechanism, including information exchange and decision making. Then we propose an double combinatorial auction-based AV pricing scheme to optimally determine the request-AV allocation. The devised auction can accommodate multiple requests and AVs at same the time, in the presence of different service types. We formulate WDP of the auction as an IQP, which is then transformed into an ILP for the ease of computation. In addition, we analyze the properties of WDP with duality and the results are engaged to develop a service charge determination rule. The proposed pricing mechanism is evaluated with a wide range of randomly generated cases of different sizes, and the simulation results demonstrate its superiority in maximizing social welfare. In addition, the proposed mechanism significantly outperforms the previous AVPTS pricing mechanisms with higher utility and comparable computational time.

The future work can be generally classified into two topics. As discussed in Section IV-C, how to properly handle the huge problem solution space (leading to long computational time) for AVPTS deployed in large cities is an interesting problem to be investigated. Another possible extension is to study the security issues in the proposed pricing mechanism, e.g., false-bidding. Although [10] can resolve some of these issues, how to incorporate security measures to handle multiple simultaneous requests is still an open problem.

Appendix A
Passenger Waiting Time

In our proposed pricing mechanism, the waiting time of requests (passengers) are not considered in WDP. This is because that in this paper, we mainly consider the economical concerns in the pricing process of AVPTS, and the main objective is to maximize the social welfare of the system. In spite of this, the waiting time can be modeled as a hard constraint to prevent overly long waiting time experienced by the passengers. In Problem 2, consider that request r can be kept on hold for a maximum duration of $\bar{t}$ and the driving time for $k \in \mathcal{K}$ to arrive at r is $t_{k,r}$. Then we can have the following to satisfy the customer waiting time requirement:

$$x_k(\mathcal{S}, r) = 0, \forall r \in \mathcal{R}, k \in \mathcal{K}^-, \mathcal{S} \in \mathcal{Q}_k, \tag{17}$$

where $\mathcal{K}^- = \{k | k \in \mathcal{K}, t_{k,r} > \bar{t}\}$. This constraint forbids those vehicles that cannot arrive at r within $\bar{t}$ from serving r.

References

[1] R. Hult, G. R. Campos, E. Steinmetz, L. Hammarstrand, P. Falcone, and H. Wymeersch, "Coordination of Cooperative Autonomous Vehicles: Toward safer and more efficient road transportation," *IEEE Signal Process. Mag.*, vol. 33, no. 6, pp. 74–84, Nov. 2016.

[2] K. Zheng, Q. Zheng, H. Yang, L. Zhao, L. Hou, and P. Chatzimisios, "Reliable and efficient autonomous driving: The need for heterogeneous vehicular networks," *IEEE Commun. Mag.*, vol. 53, no. 12, pp. 72–79, Dec. 2015.

[3] J. Nie, J. Zhang, W. Ding, X. Wan, X. Chen, and B. Ran, "Decentralized Cooperative Lane-Changing Decision-Making for Connected Autonomous Vehicles," *IEEE Access*, vol. 4, pp. 9413–9420, 2016.

[4] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt, M. Sokolsky, G. Stanek, D. Stavens, A. Teichman, M. Werling, and S. Thrun, "Towards fully autonomous driving: Systems and algorithms," in *2011 IEEE Intelligent Vehicles Symposium (IV)*, Jun. 2011, pp. 163–168.

[5] M. F. Hyland and H. S. Mahmassani, "Taxonomy of Shared Autonomous Vehicle Fleet Management Problems to Inform Future Transportation Mobility," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2653, pp. 26–34, Jan. 2017.

[6] A. Y. S. Lam, Y. W. Leung, and X. Chu, "Autonomous-Vehicle Public Transportation System: Scheduling and Admission Control," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 5, pp. 1210–1226, May 2016.

[7] M. Xia, J. Stallaert, and A. B. Whinston, "Solving the combinatorial double auction problem," *European Journal of Operational Research*, vol. 164, no. 1, pp. 239–251, Jul. 2005.

[8] K. Michael, "When Uber Cars Become Driverless: "They Won't Need No Driver" [Editorial]," *IEEE Technol. Soc. Mag.*, vol. 35, no. 2, pp. 5–10, Jun. 2016.

[9] A. Y. S. Lam, "Combinatorial Auction-Based Pricing for Multi-Tenant Autonomous Vehicle Public Transportation System," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 3, pp. 859–869, Mar. 2016.

[10] J. J. Q. Yu and A. Y. S. Lam, "Core-Selecting Auctions for Autonomous Vehicle Public Transportation System," submitted for publication.

[11] K. Xu, Y. Zhang, X. Shi, H. Wang, Y. Wang, and M. Shen, "Online combinatorial double auction for mobile cloud computing markets," in *2014 IEEE 33rd International Performance Computing and Communications Conference (IPCCC)*, Dec. 2014, pp. 1–8.

[12] R. McAfee and J. McMillan, "Auctions and Bidding," *J. Econ. Lit.*, vol. 25, no. 2, pp. 699–738, 1987.

[13] V. Krishna, *Auction Theory*, 2nd ed. Burlington, MA: Academic Press, Aug. 2009.
[14] T. Sandholm, "Approaches to winner determination in combinatorial auctions," *Decision Support Systems*, vol. 28, no. 1–2, pp. 165–176, Mar. 2000.
[15] V. L. Smith, *Combinatorial Auctions*, P. Cramton, Y. Shoham, and R. Steinberg, Eds. The MIT Press, Jan. 2010.
[16] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, Mass: Athena Scientific, Sep. 1999.
[17] S. Boyd and L. Vandenberghe, *Convex Optimization*, 1st ed. Cambridge, UK ; New York: Cambridge University Press, Mar. 2004.
[18] "Gurobi Optimization," http://www.gurobi.com/.
[19] M. Yokoo, Y. Sakurai, and S. Matsubara, "The effect of false-name bids in combinatorial auctions: New fraud in internet auctions," *Games and Economic Behavior*, vol. 46, no. 1, pp. 174–188, Jan. 2004.
[20] R. W. Day and S. Raghavan, "Fair Payments for Efficient Allocations in Public Sector Combinatorial Auctions," *Management Science*, vol. 53, no. 9, pp. 1389–1406, Jul. 2007.