Solving the Winner Determination Problem in Combinatorial Auctions for Fractional Ownership of Autonomous Vehicles

Mahdi Takalloo, Aigerim Bogyrbayeva, Hadi Charkhgard, and Changhyun Kwon*

Department of Industrial and Management Systems Engineering, University of South Florida, Tampa, FL, 33620

{mtakalloo, aigerimb, hcharkhgard, chkwon}@usf.edu

July 31, 2020

Abstract

The introduction of autonomous vehicles to consumer markets will expedite the trend of car-sharing and enable co-owning or co-leasing a car. In this paper, we consider a combinatorial auction market for fractional ownership of autonomous vehicles, which is unique in two aspects. First, items are neither pre-defined nor discrete; rather, items are continuous time slots defined by bidders. Second, the spatial information of bidders should be incorporated within the winner determination problem so that sharing a vehicle is indeed a viable plan. The consideration of spatial information increases the computational complexity significantly. We formulate the winner determination problem, which plays a critical role in various auction designs and pricing schemes, for both discrete- and continuous-time settings. In terms of social welfare maximization, we show that the continuous-time model is superior to the discrete-time model. We provide a conflict-based reformulation of the continuous-time model, for which we develop an effective solution approach based on a heuristic and maximal-clique based reformulations. Using samples of the 2010–2012 California Household Travel Survey, we verify that the proposed solution methods provide effective computational tools for the combinatorial auction with bidder-defined items.

Keyword: Fractional Ownership; Combinatorial Auction; Bidder-Defined Items; Clique

1 Introduction

People are not interested in owning a car as much as they once were due to several factors, including congestion, parking problems, and the cost of maintaining and operating the vehicle. It is no

*Corresponding author.

C Kwon acknowledges partial support through a University of South Florida Nexus Initiative (UNI) Award.

(a) Conventional vehicles (b) Autonomous Vehicles

Figure 1: Fractional ownership under conventional vehicles and autonomous vehicles

surprise that peer to peer ride-sharing services like Lyft and Uber and car-sharing services like ZipCar and Car2go have grown significantly in recent years. In response to these recent changes, fractional ownership has also gained attention as a viable alternative to traditional car ownership. By doing so, people can split the cost of leasing. Moreover, given that people have such varying needs, someone who takes two or three trips a week may find fractional ownership even cheaper than car-sharing or ride-sharing services. Note that ride-sharing services, car-sharing services, and fractional vehicle ownership correspond to different needs. While ride-sharing services may be preferred for unplanned trips and car-sharing services may be more suitable for occasional and short-term use, the fractional ownership may be a better and cheaper option for long-term planned trips. On the contrary to ride-sharing and car-sharing services, the fractional ownership does not require customers to look for reservations every time or occasion they commute but instead provides a predictable long-term service with fixed rates. Moreover, in contrast to the ride-sharing services and some car-sharing services, the availability of vehicles and surge pricing and the inconvenience of locating a car is not an issue under fractional vehicle ownership. In addition, fractional ownership provides a cleaner option compare to ride-sharing and car-sharing services as only a few customers use the vehicle.

The introduction of autonomous vehicles (AVs) to consumer markets will expedite this trend. Even with extreme congestion on the road, “driving” an AV can be a pleasant experience: one can sleep or watch a movie in the car. Moreover, since AVs can travel without passengers, sharing vehicles will become easier. The fractional vehicle ownership may not be practical under conventional vehicles because it needs re-positioning cars from one customer’s location to another or using another form of transportation for some customers. In contrast, fractional vehicle ownership can potentially become a viable option under AVs. Consider the example in Figure 1. Suppose two customers are willing to share a vehicle. As Figure 1a shows, under conventional vehicles, customer 1 goes from her origin to her destination using a conventional vehicle. Then, customer 2 must come to the customer 1 destination to pick the vehicle up, which requires another form of mobility (M). However, as Figure 1b shows, using an autonomous vehicle, the co-owned AV can travel autonomously from customer 1 destination to customer 2 origin, and there is no need for another form of mobility. Therefore, we envision that AVs can be co-owned more easily, and new markets can be created accordingly.

Co-owning an item is not a new idea. In the private jet market, fractional ownership has been popular (Hicks et al., 2005). One can buy a share in a jet and use it for an allocated amount of time. In the real-estate market, especially for resort condominiums, a similar form of ownership exists, known as *timeshare* (Terry, 1994). Multiple owners hold the rights to use a particular property and each owner is allocated to a certain number of periods of time. The price each owner pays depends on the length of time periods and the season.

Various car manufacturers with conventional vehicles have experimented fractional ownership with some restrictions on the location. Ford Credit Link was a pilot program for co-leasing vehicles in Austin, Texas, but few people registered. Audi had a similar pilot fractional ownership program 'Audi Unite' in Stockholm, Sweden. After selecting an Audi model and forming a group to co-lease a vehicle of the chosen Audi model, one could use a smart phone application to reserve a time slot to drive the vehicle. Each customer splits the leasing cost either in proportion to the usage time or at the fixed rate. Nissan Intelligent Get & Go Micra, in Paris, France, used a matching algorithm to form car sharing communities, which partly own a vehicle. All these programs are no longer active.

While the future of this new form of vehicle ownership is unclear with conventional vehicles, it certainly becomes a viable option with AVs. Inconveniences of having to locate and walk to a car can be overcome with the self-driving capability. As typical cars are parked 95% of the time (Shoup, 2005), there is a great potential for AV fractional ownership to change the current structure of car ownership. We note that when people share AVs, *co-leasing* may be a more viable service due to maintenance and insurance issues. In this paper, 'fractional ownership' means that customers co-lease a vehicle and 'co-owners' mean co-lessees.

For a practical fractional AV ownership model to be successful, there must be little to no time conflicts among co-owners. Therefore, a suitable mechanism is needed to match customers with non-overlapping time-schedules together and avoid conflicts. With these elements in mind, we consider an auction market for fractional AV ownership as an alternative to the traditional full ownership model.

1.1 Combinatorial Auctions

Combinatorial auctions are suitable mechanisms to sell items or allocate resources in packages, instead of single items separately. Due to complementarity and substitutability among items, the value of a package of items may not be the same as the summation of the individual values of those items. Furthermore, in many situations, bidders participating in combinatorial auctions have preferences only for a package of items, not for individual items separately. Combinatorial auctions capture these unique characteristics through bidding languages, winner determination methods, and pricing schemes that are specifically designed. For general descriptions of combinatorial auctions, we refer readers to De Vries and Vohra (2003) and Pekeč and Rothkopf (2003).

Combinatorial auctions have been used across various industry sectors. Federal communication commission initiated combinatorial auctions in 1994 to allocate the spectrum right licenses to telecommunication companies. Combinatorial auctions have also been proposed for Internet pricing (Hershberger and Suri, 2001) and assigning contracts to private companies for meal delivery to schools (Epstein et al., 2004) and resource allocation in cloud computing (Samimi et al., 2016). In transportation and logistics, combinatorial auctions have gained attention for selling airport departure and arrival slots (Rassenti et al., 1982), truckload transportation (Zhang et al., 2015), city bus route market (Cantillon and Pesendorfer, 2006), and railway industry (Kuo and Miller-Hooks, 2015). Collaborative transportation has gained attention as an effective approach to exchange transportation requests and save the cost (Guajardo and Rönnqvist, 2016; Gansterer and Hartl, 2017). Researchers have used the combinatorial auctions in collaborative vehicle-routing for carriers as well (Gansterer and Hartl, 2016, 2018; Li et al., 2016). Combinatorial auctions have also been proposed in ride-sharing market for designing a more efficient shared mobility system (Hara and Hato, 2018). For more information regarding combinatorial auction and its applications, see Smith et al. (2006); Porter et al. (2003); Abrache et al. (2007).

In this paper, we consider a new application of combinatorial auctions, the fractional AV ownership market. In particular, we study single-round combinatorial auctions, in which the auctioneer is a car manufacturer or leasing company that wants to sell AVs, and the bidders are customers who are interested in co-leasing a car. The auctioneer sells time-slot packages to bidders through an auction that gives the winners the right to use the same vehicle in these time-slots within a week for a certain period.

Each time-slot package includes several time-slots covering the travel needs of customers. For example, one package can offer the right to use the vehicle for Monday and Tuesday between 8 AM to 9 AM, and Thursday from 6 PM to 7 PM.

In the designed auction market, bidders submit multiple bids for various packages, and at most, one of them will be accepted. For instance, if a customer wishes to use the vehicle on Monday between 8 AM to 9 AM, and Tuesday or Thursday from 6 PM to 7 PM, she will submit two bids, one for Monday between 8 AM to 9 AM, and Tuesday from 6 PM to 7 PM, and another one for Monday between 8 AM to 9 AM, and Thursday from 6 PM to 7 PM. Each bid includes a time-slot package and the bidding price offered by the bidder of that package. The bidders must submit their spatial information at the beginning and ending of the time-intervals they intend to use the vehicle. Considering the available vehicles and the spatial information of the bidders, the company determines the set of winning bids to maximize social welfare. Next, the auctioneer determines the payment amount of each winner following some pricing rules.

Other mechanisms, such as stable or maximal matching, can also be used to match the customers (Zhang et al., 2020). The leasing company, however, needs to price each possible package of time slots as well as generate packages with pre-determined time intervals for these mechanisms. Both pricing and package generation are complicated tasks for the leasing company. In auction mechanisms, customers can determine the packages and the bidding prices by themselves.

1.2 Unique Challenges and Contributions

The setting of the problem under study is unique in several aspects. First, in most existing combinatorial auctions, items are pre-defined and discrete. In our proposed auction market, items are neither pre-defined nor discrete; instead, we consider *bidder-defined continuous-time items.* Every possible time interval is a potential item; it becomes an item indeed when a bidder finds the time interval valuable. As we will discuss later, if we define the items as fixed pre-defined time-slots, we obtain lower social welfare compared to the case where we consider flexible bidder-defined time slots. Second, the fractional vehicle ownership model incorporates the bidder's location as well. If the customers are far away and the time gap between their trips is short, they cannot co-own a car together. A suitable formulation for the model requires not only to consider these unique characteristics but also to provide a suitable framework for designing effective solution approaches as well.

Determining the set of winners of the auction is another challenge. The winner determination problem (WDP) can usually be formulated as a weighted set packing problem, which is known to be NP-hard (Rothkopf et al., 1998). We can categorize methods for solving the WDP into three groups based on the used solution approaches. Given the intractability of finding optimal allocation in combinatorial auctions, the first group of studies (Rothkopf et al., 1998; Tennenholtz, 2000) restricts the set of bids; thereby the problem is solvable in polynomial time. Unfortunately, restricting bids is not possible in many applications and applying these methods will result in a loss of efficiency. The second group of studies solves the problem to optimality using various branching methods (Fujishima et al., 1999; Sandholm et al., 2005) The third group of studies develops heuristics approaches, which include virtual simultaneous auctions (Fujishima et al., 1999), equilibrium heuristics (Tsung et al., 2011), and Tabu search heuristics (Wu and Hao, 2015).

In this study, considering the spatial information of bidders, we first formulate the WDP in a discrete-time setting where the items are fixed and predefined time slots. In the discrete-time setting, the period under the study is divided into equally-sized time-slots, and bidders can only bid on these predefined time-slots. We use simple examples to show the weakness of the discrete-time formulation. Next, we formulate the problem in a continuous-time setting, where the bidders define items, and we represent its superiority compared to the discrete-time setting. For the continuous-time WDP, we develop a conflict-based formulation, which enables us not only to generate the model more effectively, but also to solve it for relatively large-sized instances exactly. To solve large-scale problems, we design the Sequential Single-Vehicle Decomposition (SSVD) heuristic to find a high-quality primal solution. We also develop a maximal-clique based relaxation of the problem to obtain a dual bound for the WDP.

We summarize the contributions of this paper as follows. First, to the best of our knowledge, this paper is the first study considering combinatorial auction approaches for the fractional vehicle ownership of AVs. While researchers have studied combinatorial auctions in several areas of transportation, the problem of AV fractional ownership markets has not been addressed yet. Second, we design a new type of combinatorial auctions with unique characteristics, namely, *com-*

binatorial auctions with bidder-defined items. Instead of using predefined items (time slots in the combinatorial auction market), bidders define them based on their travel needs. Moreover, we incorporate the spatial information of bidders in our formulation and examine whether a package of bids is feasible considering this information. Third, we devise an effective computational method for solving the WDP, which considers bidder-defined items the spatial information. Since the WDP is NP-hard, and it may not be possible to solve it to optimality, we design the Sequential Single-Vehicle Decomposition (SSVD) heuristic for finding a high-quality primal solution. We also propose a maximal-clique based relaxation to find a dual bound for the WDP. We demonstrate the performance of our solution approach through numerical experiments. While we do not consider a particular payment scheme in this paper, efficient solutions methods for the WDP are important as the WDP solution provides a basis for various auction mechanisms. We also want to emphasize that we can apply the formulations and the solution approaches proposed in this paper to some other combinatorial auction problems when the continuous-time setting and bidder-defined continuous-time items are considered. Such applications may include arrival and departure slots allocation in airports, matching drivers and customers in ride-sharing transportation systems, and resource allocation for cloud computing.

The remainder of this paper is organized as follows. In Section 2, we introduce the winner determination problem in combinatorial auctions. In Section 3, we describe the setting of the fractional AV ownership market. We present the WDP under discrete-time and continuous-based settings and compare them in Section 4. In Section 5, we describe solution approaches to solve the WDP. In Section 6, we present numerical experiments based on California 2010–2012 travel survey datasets. We finally conclude this paper in Section 7.

2 The Winner Determination Problem (WDP)

Suppose $\mathcal{I}$ denotes the set of bidders, and set $\mathcal{K}$ represents the set of distinct items. Each bidder $i \in \mathcal{I}$, submits a set of bids $\mathcal{B}_i$. Each bid b_j is a 2-tuple $(c_j, \mathbf{a}_j)$, where c_j is the bidding price for bid j and $\mathbf{a}_j$ is a binary vector of elements a_{kj}, where $a_{kj} = 1$ if item k is in bid j, and is 0 otherwise. We assume that at most one bid from each customer can be determined as a winning bid. The auctioneer pools the submitted bids and determines the winning bids by solving the following WDP:

$$
\begin{aligned}
Z^*_{\text{WDP}} = \max_{x_j} \quad & \sum_{j \in \mathcal{J}} c_j x_j && && (1)\\
\text{s.t.} \quad & \sum_{j \in \mathcal{J}} a_{kj} x_j \le 1 && \forall k \in \mathcal{K} && (2)\\
& \sum_{j \in \mathcal{B}_i} x_j \le 1 && \forall i \in \mathcal{I} && (3)\\
& x_j \in \{0,1\} && \forall i \in \mathcal{I}, \forall j \in \mathcal{B}_i && (4)
\end{aligned}
$$

which is a special case of the weighted set packing problem. Binary variable x_j specifies whether package j is awarded to the corresponding bidder or not. Objective function (1) maximizes the social welfare (the summation of winning bidding prices). Constraint (2) ensures that each item assigned to at most one bid. Constraint (3) specifies that at most one bid from each bidder can be a winning bid. The optimal solution $\mathbf{x}^*$ to the above problem specifies the optimal allocation.

After determining the set of winners, the auctioneer calculates the payment amount for each winner according to some payment rules. Choosing a suitable payment rule is an important part of the auction design, as it influences the bidding strategy of the bidders and the revenue for the auctioneer. Note that different pricing rules can be used to determine the payments for the winners. The first-price rule, the VCG payment rule (Vickrey, 1961), and the core-selecting pricing (Day and Cramton, 2012) are among the proposed payment methods for combinatorial auctions. In the first-price auction, the winners pay the same amount as they bid. Under the VCG payment rule, which incentivizes bidders to bid truthfully (incentive compatibility) in contrast to first-price auction, bidder i pays the amount:

$$p_i = Z^*_{\mathrm{WDP}_{-i}} - \left(Z^*_{\mathrm{WDP}} - \sum_{j \in \mathcal{B}_i} c_j x_j^* \right) \tag{5}$$

where $Z^*_{\mathrm{WDP}_{-i}}$ is the optimal social welfare when bidder i's bids are removed from the set of all bids. Although the VCG mechanism satisfies incentive compatibility, it does not perform well concerning the auctioneer's revenue (Sandholm and Likhodedov, 2015). Moreover, for large problems, we may not be able to solve the WDP to optimality, and under sub-optimal allocation, VCG mechanism does not satisfy incentive compatibility. Day and Cramton (2012) propose quadratic core-selecting payment rules to improve the revenue of the auctioneer while staying close to the VCG payments.

While finding a suitable payment rule is beyond the scope of this study, some well-known pricing rules, including VCG payment rule and quadratic core-selecting payment method involves solving the WDP multiple times. Therefore, it is essential to provide an effective solution approach to solve the WDP as it not only determines the winners of the auction but also affects the payments of the winners in the auction. Note that the solution approaches proposed for the WDP in this study can be used under any payment rules and auction mechanisms. For a more detailed description and comparison of the different payment rules for the combinatorial auction for fractional ownership of AVs, readers are referred to Bogyrbayeva et al. (2020).

3 Combinatorial Auction Design for Fractional Ownership of Autonomous Vehicles

In this section, we describe the combinatorial auction setting for the fractional vehicle ownership market.

3.1 Items and bidders

Items are basic units of goods or services offered in the auction. In the fractional AV ownership market, the item is the right to use a vehicle at a particular time slot within a specific day each week for a certain period, which we call a leasing period. For instance, the right to use the vehicle on Monday from 8 PM to 10 PM is a potential item. There are two possible settings for defining the time-slots. In the first setting, the auctioneer defines these time slots, and the bidders can only bid on these time-slots. For instance, the auctioneer may divide each day into the following time slots: 12 AM–6 AM, 6 AM–12 PM, 12 PM–6 PM and 6 PM–12 AM. In this example, the right to use a vehicle on Tuesday between 6 AM–12 PM is a valid item, while the right to use a vehicle on Wednesday between 11 AM–3 PM is not an acceptable item. We call this setting the discrete-time setting. In the second, continuous-time, setting, bidders define time-slots. In a continuous-time setting, bidders can bid on any possible time intervals rather than the restricted discrete fixed predefined time-slots. We will later compare these two settings. We are considering a leasing period from one month to six months for the fractional ownership market. Moreover, we assume a homogeneous vehicle fleet for the proposed auction. If the vehicles are not homogeneous, we can design an auction for each vehicle model separately. Each bidder needs to specify the following information in the submitted bids: *Trip schedule:* Set of items (time intervals) in which the bidder wishes to use the vehicle; *Bidding price:* The price offered by the bidder for the corresponding trips; and *Location information:* The spatial information of the vehicle at the origin and the destination of each trip. Spatial information will be used to estimate the commuting time between bidders and consequently find the feasible matches. There is no limit on the number of bids a bidder can submit. However, at most, one bid from each customer can be accepted.

3.2 The Auction Format

In this study, we consider a one-shot sealed bid auction format in which bidders simultaneously submit bids on any individual items or any packages of the items, without having any information about the bids submitted by the others. Next, the auctioneer pools all the bids and solve the WDP by specifying the best allocation of packages to the bidders. Afterward, the auctioneer computes the payment of each winner according to some payment rules such as the first-price scheme, the VCG payment rule, and the core-selecting pricing. While designing practical auction mechanisms and payment rules are beyond the scope of this study, the formulations and the solution approaches introduced in Sections 4 and 5 can be used in various practical mechanisms, including iterative auctions with different pricing rules, as in Bogyrbayeva et al. (2020).

4 Formulating the Winner Determination Problems

In this section, we formulate the winner determination problem under both the discrete-time setting with predefined time slots and the continuous-time setting with bidder-defined time slots.

4.1 The Discrete-Time Winner Determination Problem

Suppose $\mathcal{I}$ denotes the set of bidders participating in the fractional vehicle ownership combinatorial auction. Each bidder $i \in \mathcal{I}$ submits a set of bids $\mathcal{B}_i$, which includes the bidding price c_j, the corresponding time-slots, and the location of the bidder at the origin and destination of each trip. The constant a_{jt} denotes whether a bid j includes a time slot t or not; a_{jt} is 1 if a time-slot $t \in \mathcal{T}$ is included in a bid j and is 0 otherwise. We denote the set of all time slots by $\mathcal{T}$ and the length of each time-slot by Δt. Extracted from bidder's location data, the parameter $r_{iktt'}$ represents the time it takes to commute from bidder i's location at time slot t to a bidder k's location at time slot t' (note that bidders can be in different locations at different times). We formulate the discrete-time WDP as follows:

$$
\begin{aligned}
(\mathsf{P0}) \quad \max_{x_{jv}} \quad & \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{B}_i}\sum_{v\in\mathcal{V}} c_j x_{jv} && && (6)\\
\text{s.t.} \quad & \sum_{j\in\mathcal{J}} a_{jt} x_{jv} \le 1 && \forall t \in \mathcal{T}, v \in \mathcal{V} && (7)\\
& \sum_{j\in\mathcal{B}_i}\sum_{v\in\mathcal{V}} x_{jv} \le 1 && \forall i \in \mathcal{I} && (8)\\
& t\Delta t + r_{iktt'} x_{jv} \le (t'-1)\Delta t + M(1 - x_{lv}) && \forall i,k \in \mathcal{I}, j \in \mathcal{B}_i, l \in \mathcal{B}_k, v \in \mathcal{V} && (9)\\
& && t,t' \in \mathcal{T} : t' > t, a_{jt}a_{lt'} = 1 && \\
& x_{jv} \in \{0,1\} && \forall i \in \mathcal{I}, j \in \mathcal{B}_i && (10)
\end{aligned}
$$

where M is a big positive constant. In the formulation above, the decision variable x_{jv}, is 1 if bid j is assigned to vehicle v and is 0 otherwise. The objective function (6) maximizes social welfare. Constraint (7) ensures that each time slot for each vehicle is assigned to at most one bid. Constraint (8) states that at most, one bid from each bidder can be determined as a winner. Constraint (9) ensures that conflicting bids do not match. As an aside, formulation (P0) can be improved in practice by removing unnecessary time slots from $\mathcal{T}$. In other words, for each trip in each bid, $\mathcal{T}$ can only contain the starting time slot and ending time slot corresponding to that trip. For example, if a given trip in a given bid contains time slots 3–6, then $\mathcal{T}$ should contain time slots 3 and 6, but it is not necessary to insert time slots 4 and 5 in $\mathcal{T}$.

Although the discrete-time WDP provides a suitable framework, it is inferior compared to the continuous-time formulation in terms of social welfare. Moreover, increasing the number of time slots does not necessarily improve social welfare. The following examples specify the drawbacks of the discrete-time formulation.

Example 1. Consider two bidders, bidder A, and bidder B, who are bidding in a fractional ownership market with a single vehicle. Suppose the distance between the two bidders is negligible. Moreover, assume that the period under study $[t_0, t_3]$ is divided into three time-slots $[t_0, t_1]$, $[t_1, t_2]$, $[t_2, t_3]$. Figure 2 represents the trip schedules of bidder A and B. Bidder A bids c_A on time-slot $[t_0, t_1]$, and bidder B bids c_B on time slots $[t_0, t_1]$, $[t_1, t_2]$, $[t_2, t_3]$. Under the discrete-time setting,

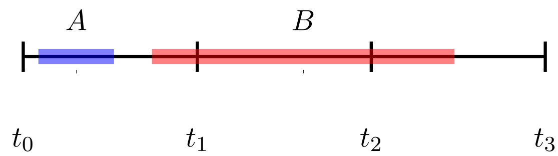
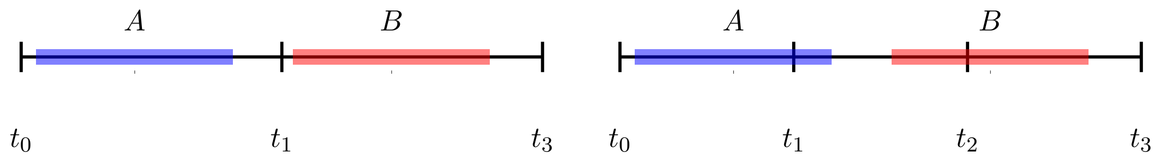


Figure 2: Trip schedules of bidder A and B in Example 1



(a) with two time slots

(b) with three time slots

Figure 3: The trip schedules of bidder A and B in Example 2

the social welfare is equal to $\max\{c_A, c_B\}$, while in the continuous-time setting with bidder-defined items, social welfare is equal to $c_A + c_B$ which is strictly greater than $\max\{c_A, c_B\}$ under positive bidding prices.

Example 2. Consider the auction in Example 1. Suppose bidders A and B have trip schedules represented in Figure 3a. Assume that the period under study $[t_0, t_3]$ has been divided into two time-slots $[t_0, t_1]$ and $[t_1, t_3]$. Under the discrete-time setting, bidder A bids on the time-slot $[t_0, t_1]$, and bidder 2 bids on the time slot $[t_1, t_3]$. Social welfare is equal to $c_A + c_B$. Now suppose, we increase the number of time-slots from 2 to 3, as depicted in Figure 3b. In this case, bidder A bids on time-slot $[t_0, t_1]$ and $[t_1, t_2]$, and bidder B bids on $[t_1, t_2]$ and $[t_2, t_3]$. The social welfare is equal to $\max\{c_A, c_B\}$ which is less than or equal to $c_A + c_B$. Therefore, increasing the number of time-slots may cause even less welfare.

4.2 The Continuous-Time Winner Determination Problem

In the continuous-time setting, bidders submit the exact departure and the arrival times of trips along with bidding price and spatial information. Note that each bid includes several trips. Two bids conflict with each other if there is at least one trip in one bid that overlaps with at least one trip in another bid. Set $\mathcal{T}_j$ denotes the set of trips in bid j. Each trip $n \in \mathcal{T}_j$ is represented by a pair (s_n, e_n), where s_n is the departure time/start time and e_n is the arrival time/end time of that trip. A parameter $r_{ike_ms_n}$ represents the time it takes for an AV to drive from bidder's i location at time e_m to bidder's k location at time s_n. With these notations, we can formulate the combinatorial auction problem under the continuous-time setting as

$$
\begin{aligned}
(\mathsf{P1}) \quad \max_{x_{jv}} \quad & \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{B}_i} \sum_{v \in \mathcal{V}} c_j x_{jv} && && (11)\\
\text{s.t.} \quad & \sum_{j \in \mathcal{B}_i} \sum_{v \in \mathcal{V}} x_{jv} \leq 1 && \forall i \in \mathcal{I} && (12)\\
& e_m x_{jv} \leq s_n + M(1 - x_{lv}) && \forall i, k \in \mathcal{I}, j \in \mathcal{B}_i, l \in \mathcal{B}_k, && (13)\\
& && \quad m \in \mathcal{T}_j, n \in \mathcal{T}_l : s_m \leq s_n \leq e_m && \\
& e_m + r_{ike_m s_n} x_{jv} \leq s_n + M(1 - x_{lv}) && \forall i, k \in \mathcal{I}, j \in \mathcal{B}_i, l \in \mathcal{B}_k, && (14)\\
& && \quad m \in \mathcal{T}_j, n \in \mathcal{T}_l : s_n \geq e_m && \\
& x_{jv} \in \{0, 1\} && \forall i \in \mathcal{I}, j \in \mathcal{B}_i && (15)
\end{aligned}
$$

Constraints (13) and (14) are equivalent to (7) and (9) in the discrete-time setting. Constraint (13) states that if any two trips from different bids overlap, then those two trips cannot be assigned to a single vehicle. Constraint (14) considers the commuting time between the bidder's location and ensures the feasibility of the match.

It is worth mentioning that in practice, the start time and end time of trips are subject to some uncertainty. For instance, bidder k may have some delay and finish her trip at time $e'_m > e_m$. As a result, we may not be able to serve bidder i at the pre-scheduled time s_n. To deal with this uncertainty, the auctioneer can set $r_{ike_m s_n}$ to a value that is greater than the commuting time between bidders' locations by some margin.

Problem (P1) is a binary problem with many constraints and variables. Generating the optimization model for (P1), specially Constraint (13) and Constraint (14) is time consuming. Moreover, these two constraints include big-M constraints, which make the problem more difficult to solve. In the next section, we propose a conflict-based approach to eliminate these constraints and construct and solve the model faster.

5 Computational Methods

In this section, we introduce an equivalent conflict-based formulation for the continuous-time WDP. To solve the conflict-based problem, we need to design an algorithm that finds conflicting bids efficiently. We introduce such an algorithm in Section 5.2. We introduce a heuristic to find a high-quality feasible solution for the WDP in a short time. We also introduce the so-called maximal-clique based relaxation of the problem to find a dual bound for the problem.

5.1 The Conflict-based Reformulation of the WDP

We can rewrite (P1) in an equivalent conflict-based formulation. Consider graph $G = (\mathcal{N}, \mathcal{A})$, in which $\mathcal{N}$ represents the set of nodes, and set $\mathcal{A}$ represents the set of arcs. Each node $n_j \in \mathcal{N}$

Table 1: Time-slots and bidding prices for submitted packages in Example 3

Package	Time Intervals			Bidding Price
1	6:00 AM–9:00 AM			45
2	6:00 AM–6:30 AM	2:30 PM–3:30 PM		25
3	8:00 AM–8:30 AM	3:00 PM–4:00 PM		20
4	2:00 PM–4:00 PM			20
5	10:00 AM–12:00 PM	2:00 PM–4:00 PM	6:00 PM–7:30 PM	55

represents a bid and has a weight, which is equal to the corresponding bidding price c_j. Each edge connects two nodes with conflicting bids. Conflicting bids have trips that overlap with each other.

Example 3. Suppose there are five bidders in a fractional vehicle ownership combinatorial auction. Table 1 shows the time-slots associated with each bidder's bid as well as the bidding price for those packages. Suppose that the commuting times between bidder's locations is negligible. The set of all non-matchable pairs is

$$\big\{\{1,2\},\{1,3\},\{2,3\},\{2,4\},\{2,5\},\{3,4\},\{3,5\},\{4,5\}\big\}.$$

We can have the following conflict-based constraints for this problem:

$$\begin{aligned}
&x_1 + x_2 \leq 1, \quad x_1 + x_3 \leq 1, \quad x_2 + x_3 \leq 1, \quad x_2 + x_4 \leq 1,\\
&x_2 + x_5 \leq 1, \quad x_3 + x_4 \leq 1, \quad x_3 + x_5 \leq 1, \quad x_4 + x_5 \leq 1,\\
&x_i \in \{0,1\} \quad \forall i \in \{1,2,3,4,5\}.
\end{aligned}$$

In the light of Example 3, Problem $(\mathsf{P1})$ can simply be written as follows:

$$\begin{aligned}
(\mathsf{P2}) \quad \max_{x_{jv}} \quad & \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{B}_i} \sum_{v\in\mathcal{V}} c_j x_{jv} && (16)\\
\text{s.t.} \quad & \sum_{j\in\mathcal{B}_i} \sum_{v\in\mathcal{V}} x_{jv} \leq 1 \quad \forall i \in \mathcal{I} && (17)\\
& x_{jv} + x_{lv} \leq 1 \quad \forall v \in \mathcal{V}, i,k \in \mathcal{I}, j \in \mathcal{B}_i, l \in \mathcal{B}_k : j,l \text{ are conflicting} && (18)\\
& x_{jv} \in \{0,1\} \quad \forall j \in \mathcal{B}, \forall v \in \mathcal{V} && (19)
\end{aligned}$$

Constraint (18) is a conflicting constraint, which ensures that two conflicting bids do not share a vehicle. Two bids do not match if they have at least one overlapping intervals. Mathematically speaking, bid $b_j \in \mathcal{B}_i$ *conflicts* with bid $b_l \in \mathcal{B}_k$ if and only if there exist $m \in \mathcal{T}_j, n \in \mathcal{T}_l$ such that:

$$s_m \leq e_n + r_{ike_ns_m}, \quad s_n \leq e_m + r_{ike_ms_n} \tag{20}$$

Constraint (18) can be easily generated in relatively short time compared to (13)–(14). Moreover, it makes the problem easier to solve. Next, we prove that formulations $(\mathsf{P2})$ and $(\mathsf{P1})$ are equivalent. Proofs for all propositions are provided in the appendix.

Proposition 1. (P2) *is equivalent to* (P1).

To use formulation (P2), we need to find the set of conflicting bids.

5.2 Detecting Conflicting Bids

As described above, formulation (P2) requires finding all overlapping bids. Each bid includes several trips. Two bids conflict if they have overlapping trips. A simple algorithm can be used for finding the set of conflicting bids. The algorithm works as follows: For each possible pair of bids, we sort the start time and the end time of the trips in those bids in nondecreasing order. Next, we check condition (20).

Proposition 2. *Let p be the total number of bids in a fractional vehicle ownership combinatorial auction, i.e., $p = |\mathcal{B}|$. Also, let q be the maximum number of trips in a bidding package. The time complexity of the algorithm for checking the set of conflicting bids is $O(p^2 q \log q)$.*

Once we determine the conflicting bids, we can solve problem (P2) by CPLEX or any other integer programming solver.

5.3 Sequential Single-Vehicle Decomposition (SSVD) heuristic

It usually takes a considerable amount of time to find a high-quality feasible solution for problem (P2) for large instances. In this section, we introduce SSVD to find a high-quality solution in a relatively short time.

Since we assume a homogeneous fleet of vehicles, we can decompose the combinatorial auction problem to a $|\mathcal{V}|$-round single vehicle combinatorial auction. At each round, considering the set of remaining bidders, we solve the winner determination problem for a single vehicle and find the winners. Then, we update the set of bidders by excluding the winners from the list of bidders and go to the next round. This procedure continues until we assign all the vehicles to the bidders. The SSVD algorithm steps are presented in Algorithm 1 pseudo-code.

Algorithm 1: SSVD heuristics for computing a high-quality feasible solution

Input: Set of vehicles $\mathcal{V}$, set of bidders $\mathcal{I}$, and set of bids $\mathcal{B}_i$ for each $i \in \mathcal{I}$,
Output: Set of winners $\mathcal{W}$
Initialization $\mathcal{W} \leftarrow \emptyset$;
for $v = 1$ **to** $|\mathcal{V}|$ **do**
    Consider an auction with a single vehicle v;
    Solve problem (P2) with the set of bidders $\mathcal{I}$;
    $\hat{\mathcal{W}} \leftarrow$ the set of winners of the auction ;
    $\mathcal{W} \leftarrow \mathcal{W} \cup \hat{\mathcal{W}}$;
    $\mathcal{I} \leftarrow \mathcal{I} \setminus \hat{\mathcal{W}}$;

We can provide a bound for the solution of the SSVD algorithm as the following proposition presents:

Proposition 3. *Given the same set of bids and corresponding bidding prices, let $Z^*_{\mathcal{V}}$ and $Z^G_{\mathcal{V}}$ be the optimal objective function value for* (P2) *and the objective value of the solution found by the SSVD when the set of vehicles is $\mathcal{V}$. Let $Z^*_{\mathcal{V}}$ be the optimal objective value when the set of vehicles is $\mathcal{V}'$, where $|\mathcal{V}| > |\mathcal{V}'|$. Then we have*

$$Z^*_{\mathcal{V}} - Z^G_{\mathcal{V}} \leq \frac{|\mathcal{V}|}{|\mathcal{V}'|} Z^*_{\mathcal{V}'} - Z^G_{\mathcal{V}}. \tag{21}$$

When the fleet size in vehicle fractional ownership combinatorial auction ($|\mathcal{V}|$) is large , we fail to solve (P2) to optimality. However, we are able to solve (P2) to the optimality when the fleet size is relatively small. Proposition 3 utilizes this fact to bound the solutions found by SSVD. In the first round, SSVD will find $Z^*_{\{v_1\}}$, with v_1 being the first vehicle the algorithm considers; this information can bound the performance of the algorithm as follows:

$$Z^*_{\mathcal{V}} - Z^G_{\mathcal{V}} \leq |\mathcal{V}| Z^*_{\{v_1\}} - Z^G_{\mathcal{V}}.$$

In general, however, the bound in (21) does not provide a high-quality dual bound. In Section 5.4, we introduce a relaxation of (P2), which is based on finding maximal cliques, for finding a better dual bound for the problem.

Note that problem (P2) has symmetry issues. An integer linear program is symmetric if its variables can be rearranged without structural changes in the problem (Margot, 2010). To see this, consider Example 3 and suppose the fleet size for the auction is 2. One possible solution is to assign bids $\{1, 5\}$ to vehicle 1, and $\{2\}$ to vehicle 2. However, we obtain the same objective value by assigning $\{1, 5\}$ to vehicle 2 and $\{2\}$ to vehicle 1. This symmetry problem, which arises from constraint (17), makes the problem difficult to solve as the fleet size increases. Since SSVD solves the WDP with a single vehicle in each iteration, the symmetry issue is eliminated. In Section 5.4, we develop a relaxation for (P2), which breaks the symmetry.

5.4 The Maximal-Clique Based Relaxation for Finding Dual Bounds

To assess the quality of the solution obtained by solving SSVD, finding a high-quality dual bound for the WDP is necessary. To find a high-quality dual bound, we formulate a maximal-clique-based relaxation problem for the WDP. Using maximal cliques enables us to tighten the feasible region and obtain a high-quality dual bound.

Note that each conflict constraint (18) can be viewed as a simple *clique* constraint. In graph theory, a clique is a complete subgraph of a given graph. By this definition, each conflict constraint (18) is constructed based on a clique with cardinality of two in the conflicting graph. This observation enables us to derive a better formulation by replacing constraints of the form (18) with stronger constraints based on cliques with larger cardinalities. To this end, we can use maximal cliques. Note that a maximal clique is a clique that cannot be extended by adding any other node. In Example 3, we can find two maximal cliques, namely $\{1, 2, 3\}$ and $\{2, 3, 4, 5\}$. Accordingly, we can replace conflict constraints in Example 3 by the following stronger constraints:

$$x_1 + x_2 + x_4 \leq 1, \qquad x_2 + x_3 + x_4 + x_5 \leq 1.$$

These two constraints give us a stronger formulation. For instance, $\mathbf{x} = (0.5, 0.5, 0, 0.5, 0)$ does not satisfy the above constraints, while it satisfies constraint (18).

In light of the above, we can obtain a stronger formulation by using maximal cliques:

$$\begin{aligned}
(\mathsf{P3}) \quad \max_{x_{jv}} \quad & \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{B}_i}\sum_{v\in\mathcal{V}} c_j x_{jv} && (22)\\
\text{s.t.} \quad & \sum_{j\in\mathcal{B}_i}\sum_{v\in\mathcal{V}} x_{jv} \leq 1 \quad \forall i \in \mathcal{I} && (23)\\
& \sum_{j\in\mathcal{C}_m} x_{jv} \leq 1 \quad \forall v \in \mathcal{V}, m \in \mathcal{M} && (24)\\
& x_{jv} \in \{0,1\} \quad \forall i \in \mathcal{I}, j \in \mathcal{B}_i, \forall v \in \mathcal{V} && (25)
\end{aligned}$$

where $\mathcal{M}$ is the set of all maximal cliques, and $\mathcal{C}_m$ is the set of nodes in maximal clique $m \in \mathcal{M}$. The LP relaxation of (P3) is superior to the LP relaxation of (P2). Commercial solvers such as CPLEX add similar cliques to the problem in the preprocessing phase to tighten LP feasible region of the problem.

Based on (P3), we now develop a relaxation problem to compute dual bounds. To do so, we use an aggregation technique to reduce the number of decision variables. Note that problem (P3) still suffers from the symmetry issue because of having homogeneous vehicles. Therefore, by aggregating on all vehicles, we can not only reduce the size of the problem but also break the symmetry. Specifically, we define $y_j := \sum_{v\in\mathcal{V}} x_{jv}$ for each $j \in \mathcal{B}$ and introduce the following relaxation of problem (P3):

$$\begin{aligned}
(\mathsf{R}) \quad \max_{y_j} \quad & \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{B}_i} c_j y_j && (26)\\
\text{s.t.} \quad & \sum_{j\in\mathcal{B}_i} y_j \leq 1 \quad \forall i \in \mathcal{I} && (27)\\
& \sum_{j\in\mathcal{C}_m} y_j \leq |\mathcal{V}| \quad \forall m \in \mathcal{M} && (28)\\
& y_j \in \{0,1\} \quad \forall i \in \mathcal{I}, j \in \mathcal{B}_i && (29)
\end{aligned}$$

where Constraint (28) states that at most $|\mathcal{V}|$ bids can be determined as winner among a group of conflicting bids, since there are only $|\mathcal{V}|$ available vehicles.

Proposition 4. *Problem* (R) *is a relaxation of problems* (P2) *and* (P1).

Now, one can solve problem (R) to compute a dual bound. It is evident that this problem is expected to be solved quickly because of breaking symmetry and having less decision variables. However, a key issue is that it is not always possible to add all maximal cliques to the problem because (1) finding all maximal cliques is NP-hard (Östergård, 2002), and (2) the number of maximal cliques increases exponentially. Therefore, we choose a subset of maximal cliques randomly. We use the following procedure to generate maximal cliques: We start to create a clique by adding one random bid to it. Next, we attempt to add all the other bids in a random order, if possible. We add a bid to the clique if it conflicts with all the existing bids in the clique. We repeat this process multiple times and impose a time limit as the termination condition. Also, we make sure that all the bids will initiate a clique at least once in our procedure. The time complexity for this simple procedure has been presented in Proposition 5 (Assuming that whether two bids conflict with each other or not is already known) As we will show in Section 6, problem (P3) gives us a better dual bound compare to the CPLEX dual bound for problem (P2) for large instances.

Proposition 5. *Let p be the number of maximal cliques generated, and q be the total number of bids. The time complexity of the algorithm for generating the maximal clique is $O(p.q^2)$.*

6 Numerical Experiments

The numerical experiments have concentrated on comparing the discrete and continuous models, comparing basic and conflict-based continuous formulation, and testing the solution approaches for solving the WDP (SSVD algorithm and maximal-clique relaxation). These numerical experiments are aligned with the contributions of this paper, which are formulating the WDP considering bidder-defined items and spatial information and proposing solution methods to solve the WDP effectively.

All computational experiments have been carried out on a Dell Precision Tower 7910 with 2 Intel Xeon E5-2630 2.2 GHz 10-Core Processors (30MB), 32GB RAM, and the Ubuntu 18.04.4 LTS operating system, and using 5 threads. We use the Julia programming language to implement all formulations and algorithms. In all computational studies, we use CPLEX 12.5 to solve integer programs.

Throughout this section, we use the performance profile for comparing the formulations, and the solution approaches. A performance profile presents the cumulative distribution functions of a performance index for a set of algorithms or formulations (Dolan and Moré, 2002). For example, to construct the solution time performance profile for a set of algorithms, we need to compute the ratio of the solution time for each algorithm and for each instance, and the minimum of the solution time of all algorithms on the instance. The runtime performance profile then shows the ratios on the horizontal axis and the fraction of instances with a ratio that is less than or equal to the ratio on the horizontal axis, on the vertical axis, for each algorithm. We can construct the performance profile for other performance metrics similarly. Overall, in a performance profile graph, the values in the upper left-hand corner of the graph indicate the best performance.

To test our results, we use the 2010–2012 California Household Travel Survey (https://trid.trb.

org/view/1308918), which includes the travel information of 2908 vehicles in a week. The survey dataset includes the date, departure time, arrival time, duration, mileage, and origin and destination of each trip for every vehicle under the study. We test the performance of the proposed formulation and algorithms by generating some instances based on 2010–2012 California Household Travel Survey. To generate the different bids other than the travel pattern for each bidder, we assume that bidders drop one or two trips randomly. To generate the bidding prices for a leasing period of four weeks, we use the following formula:

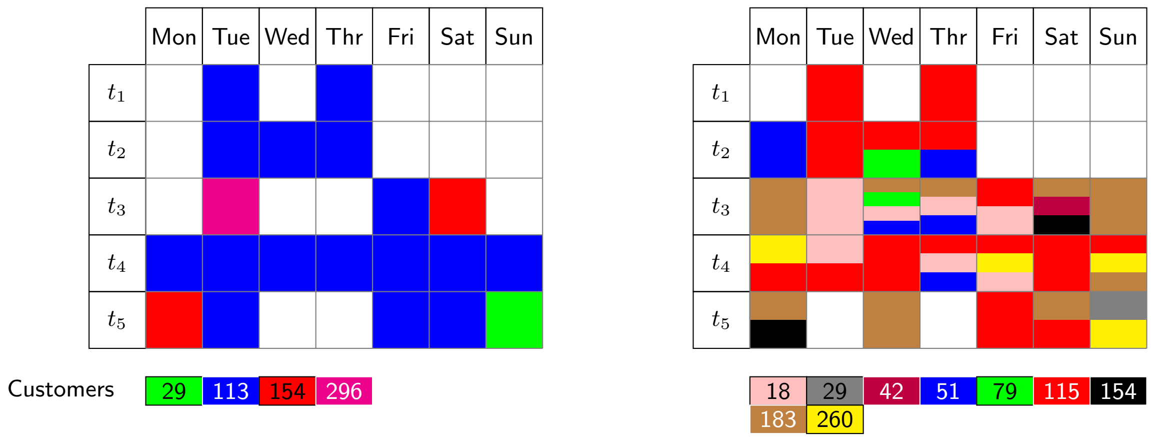
$$c_b = 4 \times (5 + 0.76\, \tilde{a}_b m_b)$$

where we assume the fixed leasing cost for each week is \$5 and the average cost per mile is \$0.76 (American Automobile Association, 2015), and $\tilde{a}_b$ is randomly drawn from the interval $[0.8, 1.2]$ and m_b is the total mileage of all the trips in the package. The location distances of each pair of customers are randomly selected between 0 and 60 minutes. The number of bidders and the fleet size in the experiments vary, depending on the classes of instances used. We will describe these classes when we present the numerical results for each experiment.

To compare the discrete-time and the continuous-time model, we consider an auction with 300 bids and a vehicle. We consider two settings. In the first setting, items are discrete time-slots, and each day is divided into five equally-length time-slots (Figure 4a). In the second setting, items are continuous time-slots (Figure 4b). As Figure 4a represents, in the discrete-time case, only 4 customers can share the vehicle because of the deficiency in the market resulted from using discrete time-slots. The travel distance commuted by the AV is 768 miles during this period. Moreover, in 15 out of 35 possible time slots, the vehicle is not being used. In the continuous-time case, the vehicle is being used by more customers and more frequently. As Figure 4b shows, 9 customers can use the vehicle in this case. Moreover, in contrast to the discrete-time setting, in each discrete time-slot, several customers can use the vehicle. Moreover, the total travel distance is 1132 miles in this case, which shows a 47 percent increase compare to the discrete-time case. In summary, by using continuous time-slots, we can increase efficiency, since more customers share an AV, and each AV travels more often.

Figure 5 highlights the importance of using continuous time-slots by comparing the social welfare under discrete-time problem (P0) and the continuous-time problem (P1) for a simulated fractional ownership auction with a single vehicle and 100 bidders. As presented, the continuous-time model is always superior to discrete-time model with respect to social welfare. Moreover, increasing the number of time slots in the discrete-time setting does not necessarily improve social welfare as it has been explained in Example 2.

Figure 6a and Figure 6b represent the effectiveness of developing the conflict-based formulation by comparing the basic formulation (P1) and the conflict-based formulation (P2). We compare the model building time (the time that Julia takes to generate the model) and the solution time of (P1) and (P2) to show the superiority of (P2). For the comparison, we generate ten classes of instances based on the fleet size and the number of bids. We set the number of bidders to 100, 150, 200, 250



$distance = 768$ $distance = 1132$

(a) Discrete-time setting (b) Continuous-time setting

Figure 4: Fractional ownership under conventional vehicles and autonomous vehicles

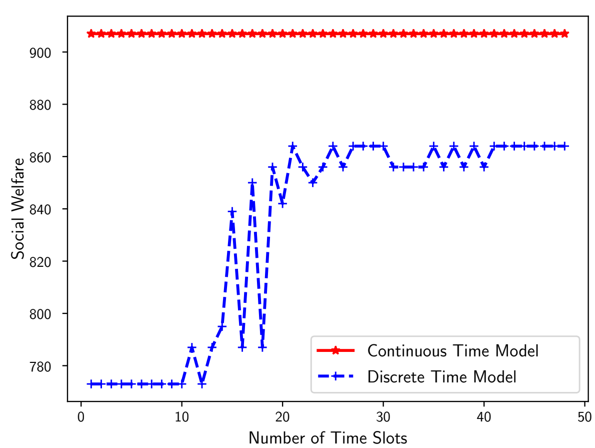
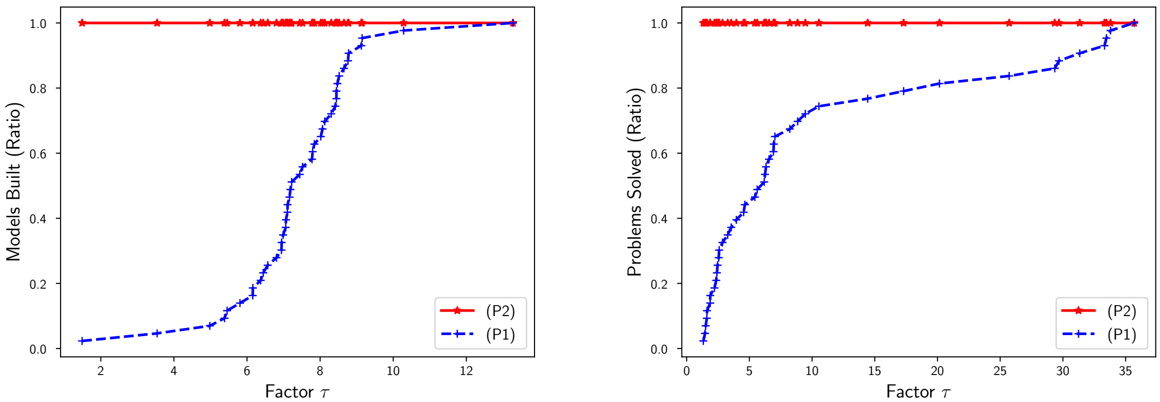


Figure 5: Comparing social welfare under the continuous-time setting and the discrete-time setting



(a) Model Building Time

(b) Model Computation Time

Figure 6: Comparing basic formulation (P1) and clique-based formulation (P2)

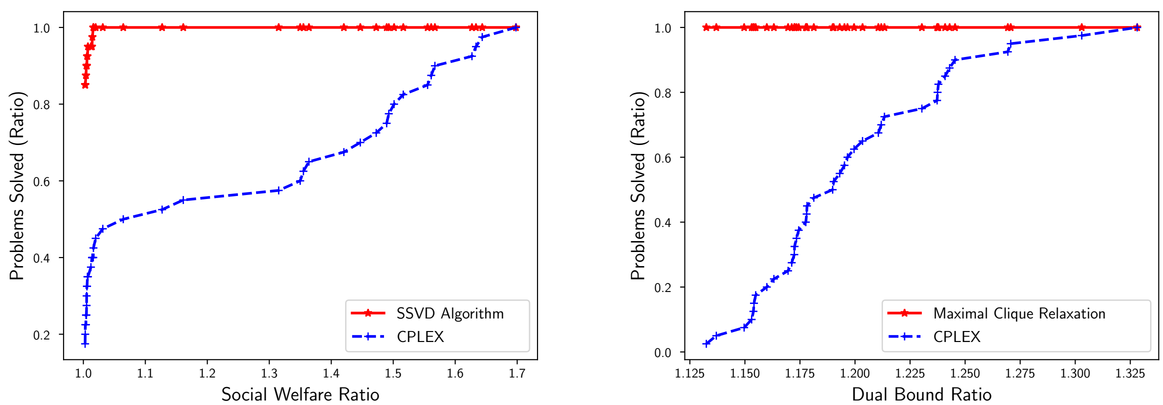
and 300, and the number of vehicles to 1 and 5. We generated five instances for each class, and use the performance profile to compare the building time and the solution time of the basic formulation (P1) and the conflict-based formulation (P2). We construct the performance profile in Figure 6a for comparing model building time as follows. For each formulation and for each problem instance, we compute the ratio between the model building time of the formulation on the instance and the minimum of the model building time of two formulations on the instance. The performance profile in Figure 6a then shows for each formulation, on the vertical axis, the fraction of instances with a ratio that is less than or equal to the factor τ shown on the horizontal axis. The performance profile in Figure 6b for solution time can be created similarly.

For almost all instances, the model building time and the solution time for the conflict-based formulation (P2) is smaller than the basic continuous formulation (P1). We could generate the conflict-based model 13 times faster compared to the basic continuous model in (P1). Moreover, we could solve (P1) 35 times faster than (P2). The reason is that formulation (P1) has big-M constraints, which makes the problem difficult to solve. However, the conflict-based formulation has only clique constraints. CPLEX utilizes these clique constraints to generate stronger constraints and solves the model much faster.

Next, we examine the performance of the solution approaches developed in Section 5 to solve (P2). We present the results for small instances (Table 2) and large instances (Figure 7 and Table 3) separately. Table 2 summarizes our results for smaller instances. We consider ten classes of instances based on the number of bidders and fleet size, as can be seen in Table 2, and generate ten instances for each class. We report the SSVD average computation time $\mathrm{T}^{\mathrm{SSVD}}$, the CPLEX average computation time for the problems solved to optimality within one hour $\mathrm{T}^{\mathrm{CPLEX}}$, and the ratio of instances in each class solved to optimality (R) by CPLEX within an hour , the gap between the primal bound obtained by CPLEX and SSVD and the CPLEX dual bound. When the number of vehicles is relatively small ($|\mathcal{V}| = 5$), 44 out of 50 instances solved to optimality within an hour

Table 2: Computational time and optimality gap for small instances

$\lvert\mathcal{I}\rvert$	$\lvert\mathcal{B}\rvert$	$\lvert\mathcal{V}\rvert$	R	$\mathrm{T}^{\mathrm{SSVD}}$(sec)	$\mathrm{T}^{\mathrm{CPLEX}}$(sec)	$\mathrm{Gap}^{\mathrm{SSVD}}$(%)	$\mathrm{Gap}^{\mathrm{CPLEX}}$(%)
100	292.6	5	1	2.41	6.47	0.83	0
150	438.8	5	1	8.30	32.72	0.87	0
200	582.8	5	1	20.28	120.77	1.58	0
250	730.0	5	1	37.79	535.22	1.37	0
300	876.2	5	0.4	55.62	1526.69	4.09	1.93
100	292.6	10	0.9	3.28	162.47	1.49	0.04
150	438.8	10	0.3	12.41	275.44	3.20	1.35
200	582.8	10	0.1	31.75	3379.70	5.03	2.93
250	730.0	10	0	64.86	>3600	6.95	5.76
300	876.2	10	0	100.39	>3600	13.26	11.53



(a) Primal Solution Ratio

(b) Dual Bound Ratio

Figure 7: Comparing primal solution and dual solution with CPLEX solution

by CPLEX, however, when the number of vehicles is larger ($|\mathcal{V}| = 10$), the symmetry becomes more severe, and CPLEX fails to solve the problem to optimality. However, SSVD can find a primal solution within 2 minutes. Moreover, as presented in Table 2, the gap between the primal bound obtained by CPLEX and SSVD is negligible (within 2.2%), which suggests that SSVD provides a high-quality primal solution. The optimality gap of SSVD is under 10% for all classes of instances except the last class, which has the highest average number of bids ($|\mathcal{B}| = 876.2$) and the number of vehicles ($|\mathcal{V}| = 10$). Hence, the SSVD can provide a high-quality solution in a short time for small instances.

Figure 7 shows the effectiveness of the SSVD and maximal-clique relaxation by comparing their performance with CPLEX for larger instances. For comparison purposes, we consider four classes of instances based on the fleet size. We set the fleet size to 15, 20, 25 and 30. We set the number of bidders to 300 for all classes. We generate ten instances for each class, and compare the solution obtained by the SSVD heuristics with the one found by CPLEX through the performance profile presented in Figure 7a. We set the CPLEX solution time limit to one hour. As Figure 7a shows, the

Table 3: Optimality gap for large instances

$\lvert\mathcal{I}\rvert$	$\lvert\mathcal{B}\rvert$	$\lvert\mathcal{V}\rvert$	T^{SSVD}(sec)	T^{CPLEX}(sec)	$\frac{Z^{\text{SSVD}}}{Z^{\text{CPLEX}}}$	Gap$^{\text{SSVD}}$(%)
300	879.2	15	147.77	>3600	1.00	15.00
300	879.2	20	178.08	>3600	1.10	13.97
300	879.2	25	200.32	>3600	1.36	12.74
300	879.2	30	217.48	>3600	1.38	11.49

SSVD heuristics outperforms in 33 out of 40 instances. The solution found by the SSVD heuristics can be as high as 1.7 times of the CPLEX solution. Figure 7b compares the dual bound found by (R) with CPLEX dual bound, found within an hour, For all instances, maximal clique relaxation (R) outperforms CPLEX dual bound. The dual bound found by (R) can outperform CPLEX by more than 30 %.

Table 3 reports the average solution time for each class, and the optimality gap for large instances obtained by comparing SSVD primal solution and maximal clique dual solution. It also reports the SSVD to CPLEX objective value ratio $\left(\frac{Z^{\text{SSVD}}}{Z^{\text{CPLEX}}}\right)$. As it can be seen in the table, the SSVD primal bound outperforms CPLEX. Moreover, SSVD is much faster and can find the primal bound in less than 4 minutes for all classes of instances. As seen in Table 3, while the optimality gap is considerable for larger instances (from 11.49% to 15%), the SSVD primal bound outperforms CPLEX.

7 Concluding Remarks

In this paper, we design a new auction market for fractional vehicle ownership market that considers spatial information of the bidders. We propose the use of bidder-defined items, which enables bidders to bid on any time intervals they wish. We formulate the WDP as a problem with conflict constraints, which enables us to generate the model and solve it faster. For solving the problem for larger instances, we design a greedy algorithm to find a primal solution and provide a maximal-clique relaxation of the problem to obtain a high-quality dual bound for that. We compare the performance of the proposed formulations and solution approaches through extensive numerical experiments.

The main contributions of this paper are introducing a new market for fractional ownership of AVs, formulating the WDP effectively considering bidder-defined items and spatial information, and proposing solution methods to solve the WDP. While designing a more practical auction mechanism is beyond the scope of this study, it is certainly an interesting future research direction. The formulations and the solution approaches introduced in this paper can be used in developing various practical mechanisms, including iterative auctions with different pricing rules, as in Bogyrbayeva et al. (2020), for example.

We can extend this paper in several other directions as well. First, the methodology of this paper can be applied to other combinatorial auctions, including resource allocation in cloud computing and ride-sharing markets with some variations, etc. Second, We can improve the solution approaches by designing algorithms based on maximal-clique based relaxation (P3). Third, we can improve the vehicle co-ownership auction market by considering other features. For instance, we can consider the uncertainty in the departure and arrival time of the bidders in the auction market.

References

Abrache, J., Crainic, T.G., Gendreau, M., Rekik, M., 2007. Combinatorial auctions. *Annals of Operations Research* 153, 1, 131–164.

American Automobile Association, 2015. Behind the numbers. https://exchange.aaa.com/wp-content/uploads/2015/04/Your-Driving-Costs-2015.pdf.

Bogyrbayeva, A., Takalloo, M., Charkhgard, H., Kwon, C., 2020. An Iterative Combinatorial Auction Design for Fractional Ownership of Autonomous Vehicles. *Working Paper, submitted to International Transactions in Operational Research*

Cantillon, E., Pesendorfer, M., 2006. Auctioning bus routes: The London experience. *Combinatorial auctions*

Day, R.W., Cramton, P., 2012. Quadratic core-selecting payment rules for combinatorial auctions. *Operations Research* 60, 3, 588–603.

De Vries, S., Vohra, R.V., 2003. Combinatorial auctions: A survey. *INFORMS Journal on Computing* 15, 3, 284–309.

Dolan, E.D., Moré, J.J., 2002. Benchmarking optimization software with performance profiles. *Mathematical Programming* 91, 2, 201–213.

Epstein, R., Henríquez, L., Catalán, J., Weintraub, G.Y., Martínez, C., Espejo, F., 2004. A combinatorial auction improves school meals in chile: a case of or in developing countries. *International Transactions in Operational Research* 11, 6, 593–612.

Fujishima, Y., Leyton-Brown, K., Shoham, Y., 1999. Taming the computational complexity of combinatorial auctions: Optimal and approximate approaches. In *IJCAI*, Vol. 99, pp. 548–553.

Gansterer, M., Hartl, R., 2016. Combinatorial auctions in collaborative vehicle routing. *IFORS News* 10, 4, 15–16.

Gansterer, M., Hartl, R.F., 2017. Collaborative vehicle routing: a survey. *European Journal of Operational Research* 268, 1–12.

Gansterer, M., Hartl, R.F., 2018. Centralized bundle generation in auction-based collaborative transportation. *OR Spectrum* 40, 3, 613–635.

Guajardo, M., Rönnqvist, M., 2016. A review on cost allocation methods in collaborative transportation. *International Transactions in Operational Research* 23, 3, 371–392.

Hara, Y., Hato, E., 2018. A car sharing auction with temporal-spatial OD connection conditions. *Transportation Research Part B: Methodological* 117, 723–739.

Hershberger, J., Suri, S., 2001. Vickrey prices and shortest paths: What is an edge worth? In *Foundations of Computer Science, 2001. Proceedings. 42nd IEEE Symposium on*, IEEE, pp. 252–259.

Hicks, R., Madrid, R., Milligan, C., Pruneau, R., Kanaley, M., Dumas, Y., Lacroix, B., Desrosiers, J., Soumis, F., 2005. Bombardier flexjet significantly improves its fractional aircraft ownership operations. *Interfaces* 35, 1, 49–60.

Kuo, A., Miller-Hooks, E., 2015. Combinatorial auctions of railway track capacity in vertically separated freight transport markets. *Journal of Rail Transport Planning & Management* 5, 1, 1–11.

Li, Y., Chen, H., Prins, C., 2016. Adaptive large neighborhood search for the pickup and delivery problem with time windows, profits, and reserved requests. *European Journal of Operational Research* 252, 1, 27–38.

Margot, F., 2010. Symmetry in integer linear programming. In *50 Years of Integer Programming 1958-2008.* Springer, pp. 647–686.

Östergård, P.R., 2002. A fast algorithm for the maximum clique problem. *Discrete Applied Mathematics* 120, 1-3, 197–207.

Pekeč, A., Rothkopf, M.H., 2003. Combinatorial auction design. *Management Science* 49, 11, 1485–1503.

Porter, D., Rassenti, S., Roopnarine, A., Smith, V., 2003. Combinatorial auction design. *Proceedings of the National Academy of Sciences* 100, 19, 11153–11157.

Rassenti, S.J., Smith, V.L., Bulfin, R.L., 1982. A Combinatorial Auction Mechanism for Airport Time Slot Allocation. *The Bell Journal of Economics* 13, 2, 402–417.

Rothkopf, M.H., Pekeč, A., Harstad, R.M., 1998. Computationally manageable combinational auctions. *Management Science* 44, 8, 1131–1147.

Samimi, P., Teimouri, Y., Mukhtar, M., 2016. A combinatorial double auction resource allocation model in cloud computing. *Information Sciences* 357, C, 201–216.

Sandholm, T., Likhodedov, A., 2015. Automated design of revenue-maximizing combinatorial auctions. *Operations Research* 63, 5, 1000–1025.

Sandholm, T., Suri, S., Gilpin, A., Levine, D., 2005. CABOB: A fast optimal algorithm for winner determination in combinatorial auctions. *Management Science* 51, 3, 374–390.

Shoup, D.C., 2005. *The High Cost of Free Parking*, Vol. 206. Planners Press Chicago.

Smith, V., Crampton, P., Shoham, Y., Steinberg, R., 2006. *Combinatorial Auctions* (1st edn.), Vol. 1. MIT Press.

Tennenholtz, M., 2000. Some tractable combinatorial auctions. In *AAAI/IAAI*, pp. 98–103.

Terry, G.A., 1994. Resort timesharing: Growth and situation analysis. *Journal of Travel & Tourism Marketing* 3, 1, 99–114.

Tsung, C., Ho, H., Lee, S., 2011. An equilibrium-based approach for determining winners in combinatorial auctions. In *Parallel and Distributed Processing with Applications (ISPA), 2011 IEEE 9th International Symposium on*, IEEE, pp. 47–51.

Vickrey, W., 1961. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance* 16, 1, 8–37.

Wu, Q., Hao, J.K., 2015. Solving the winner determination problem via a weighted maximum clique heuristic. *Expert Systems with Applications* 42, 1, 355–365.

Zhang, A., Kang, J., Kwon, C., 2020. Generalized stable user matching for autonomous vehicle co-ownership programs. *Service Science* Accepted, https://www.chkwon.net/papers/zhang_generalized.pdf.

Zhang, B., Yao, T., Friesz, T.L., Sun, Y., 2015. A tractable two-stage robust winner determination model for truckload service procurement via combinatorial auctions. *Transportation Research Part B: Methodological* 78, 16–31.

A Appendix A

Proof of Proposition 1. Formulations (P1) and (P2) possess the same objective function. Moreover, constraints (12) and (17) are exactly the same. Therefore, we only need to show that (18) is equivalent to (13)–(14).

1. First, we show that if either (13) or (14) is violated, then (18) is also violated. First, suppose that $\bar{\mathbf{x}}$ violates (13); i.e., there exist j, v, n, and m such that

$$e_m \bar{x}_{jv} > s_n + M(1 - \bar{x}_{lv}) \tag{30}$$

$$s_m \leq s_n \leq e_m \tag{31}$$

where we abbreviated the index descriptions, $i, k \in \mathcal{I}, j \in \mathcal{B}_i, l \in \mathcal{B}_k, m \in \mathcal{T}_j, n \in \mathcal{T}_l$, for simplicity. For (30), $\bar{x}_{jv} = \bar{x}_{lv} = 1$ is the only possibility. From (31), we observe:

$$s_n \le e_m \implies s_n \le e_m + r_{ike_m s_n} \text{ and } s_m \le s_n \implies s_m \le s_n \le e_n \le e_n + r_{ike_n s_m}$$

Therefore, when (31) holds, (20) also holds. In this case, $\bar{\mathbf{x}}$ does not satisfy (18).

Similarly, when $\bar{\mathbf{x}}$ violates (14), we can show that $\bar{\mathbf{x}}$ also violates (18).

2. We show that if (18) is violated, then (13)–(14) are also violated. Suppose that $\bar{\mathbf{x}}$ violates (18); i.e., there exist j, v, n, and m such that

$$\begin{aligned} \bar{x}_{jv} = \bar{x}_{lv} &= 1, \\ s_m &\le e_n + r_{ike_n s_m}, \\ s_n &\le e_m + r_{ike_m s_n}. \end{aligned}$$

If $s_n \le s_m$, we obtain $s_n \le s_m \le e_m \le e_m + r_{ike_m s_n}$, and as a result $\bar{\mathbf{x}}$ violates (13)–(14). If $s_m \le s_n$, then $s_m \le s_n \le e_n \le e_n + r_{ike_n s_m}$, and consequently, $\bar{\mathbf{x}}$ violates (13)–(14).

Hence, (18) is equivalent to (13)–(14). □

Proof of Proposition 2. Since we have p bids, there are $\binom{p}{2}$ pairs of possible bids. Moreover, since the maximum number of trips in a submitted bid is q, vector L has a maximum possible length of $2q$. Therefore, the time complexity of sorting L is $O(q \log q)$. After sorting $O(q)$ comparisons should be made. Consequently, the complexity of Algorithm 2 is

$$\binom{p}{2}[O(2q \log q) + O(q)] = \binom{p}{2}[O(q \log q)].$$

□

Proof of Proposition 3. Let $Z^{*,v}_{\mathcal{V}}$ be the social welfare associated with the v-th vehicle; that is, $Z^{*,v}_{\mathcal{V}} = \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{B}_i} c_j x^*_{jv}$ where $\mathbf{x}^*$ is an optimal solution to (P2) with $\mathcal{V}$. Without loss of generality, let us assume that $Z^{*,v}_{\mathcal{V}} \ge Z^{*,v'}_{\mathcal{V}}$ if $v \le v'$. Since $Z^*_{\mathcal{V}'}$ is optimal, from $Z^*_{\mathcal{V}'} \ge \sum_{v=1}^{|\mathcal{V}'|} Z^{*,v}_{\mathcal{V}}$ we have

$$\frac{Z^*_{\mathcal{V}'}}{|\mathcal{V}'|} \ge \frac{\sum_{v=1}^{|\mathcal{V}'|} Z^{*,v}_{\mathcal{V}}}{|\mathcal{V}'|} \tag{32}$$

Moreover, since $Z^{*,v}_{\mathcal{V}} \ge Z^{*,v'}_{\mathcal{V}}$ if $v \le v'$, we have

$$\frac{Z^*_{\mathcal{V}'}}{|\mathcal{V}'|} \ge \frac{\sum_{v=1}^{|\mathcal{V}'|} Z^{*,v}_{\mathcal{V}}}{|\mathcal{V}'|} \ge \frac{\sum_{v=|\mathcal{V}'|+1}^{|\mathcal{V}|} Z^{*,v}_{\mathcal{V}}}{|\mathcal{V}| - |\mathcal{V}'|} \tag{33}$$

Using the weighted average of the right-hand-side of (32) and (33), we obtain the following:

$$\frac{Z^*_{\mathcal{V}'}}{|\mathcal{V}'|} \geq \frac{|\mathcal{V}'|}{|\mathcal{V}|}\frac{\sum_{v=1}^{|\mathcal{V}'|} Z^{*,v}_{\mathcal{V}}}{|\mathcal{V}'|} + \frac{|\mathcal{V}|-|\mathcal{V}'|}{|\mathcal{V}|}\frac{\sum_{v=1+|\mathcal{V}'|}^{|\mathcal{V}|} Z^{*,v}_{\mathcal{V}}}{|\mathcal{V}|-|\mathcal{V}'|} = \frac{\sum_{v=1}^{|\mathcal{V}|} Z^{*,v}_{\mathcal{V}}}{|\mathcal{V}|} = \frac{Z^*_{\mathcal{V}}}{|\mathcal{V}|} \tag{34}$$

Furthermore, from the sub-optimality of SSVD, we have $Z^G_{\mathcal{V}} \leq Z^*_{\mathcal{V}}$, which along with (34) leads to

$$Z^*_{\mathcal{V}} - Z^G_{\mathcal{V}} \leq \frac{|\mathcal{V}|}{|\mathcal{V}'|} Z^*_{\mathcal{V}'} - Z^G_{\mathcal{V}}.$$

□

Proof of Proposition 4. Suppose $\bar{\mathbf{x}}$ is a feasible point for problem (P3). We let $\bar{y}_j = \sum_{v\in\mathcal{V}} \bar{x}_{jv}$. From (23) we have

$$\sum_{j\in\mathcal{B}_i}\sum_{v\in\mathcal{V}} \bar{x}_{jv} = \sum_{j\in\mathcal{B}_i} \bar{y}_j \leq 1.$$

Getting summation over v in (24) we have

$$\sum_{v\in\mathcal{V}}\sum_{j\in\mathcal{C}_m} \bar{x}_{jv} = \sum_{j\in\mathcal{C}_m} \bar{y}_j \leq |\mathcal{V}|.$$

Therefore, $\bar{\mathbf{x}}$ is a feasible solution for problem (R). As (P3) and (R) share the same objective function: $(\sum_{j\in\mathcal{B}}\sum_{v\in\mathcal{V}} c_j x_{jv} = \sum_{j\in\mathcal{B}} c_j y_j)$, problem (R) is a relaxation of problem (P3). □

Proof of Proposition 5. To add a new bid to the current clique, we need to check whether the new bid conflicts with each bid in the current cliques or not. This needs $0+1+2+,...,q-1$ operations at most. Hence the time complexity of generating a maximal clique is $O(q^2)$, and the time complexity of generating p maximal cliques is $O(pq^2)$. □