Generalized Stable User Matching for Autonomous Vehicle Co-ownership Programs

Anpeng Zhang[†], Jee Eun Kang[*†], and Changhyun Kwon[‡]

[†]Department of Industrial and Systems Engineering, University at Buffalo, SUNY
[‡]Department of Industrial and Management Systems Engineering, University of South Florida

April 12, 2020

Abstract

We investigate a new form of car sharing system that can be introduced in the market for autonomous vehicles, called *fractional ownership* or *co-ownership*. While dynamic ride sharing provides ad-hoc shared mobility services without any long-term commitment, we consider co-ownership programs with which users can still "own" a car with committed usages and ownership. We assume that an autonomous vehicle is shared by a group of users, which is only accessible by the group. We use stable matching to help users find an appropriate group to share an autonomous vehicle and present a generalized stable matching model that allows flexible sizes of groups as well as various alternative objectives. We also present a heuristic algorithm to improve computational time due to the combinatorial property of the problem.

Keywords: autonomous vehicles; stable matching; roommate matching; car sharing system; vehicle ownership; fractional ownership

1 Introduction

Recent introduction of autonomous technology, as well as the growing car-sharing market, could greatly improve people's travel experience and broaden their travel choices. Various players including General Motors, Nissan, Tesla, and Google are testing self-driving vehicles in different cities. Uncertainty on people's acceptance of autonomous vehicles, as well as the reshape of their travel patterns, arise along with the growing technologies (Lavieri et al., 2017; Winter et al., 2017). Currently, most travelers prefer owning a private vehicle in the US. Car sharing, as well as other means of transportation, can be used when private vehicles are not available. Car sharing provides users with short-term access to vehicles from car rental companies or other private vehicle owners, and the number of people using car sharing programs is growing. As of 2017, there were 40 active

[*]Corresponding author; `jeeeunka@buffalo.edu`

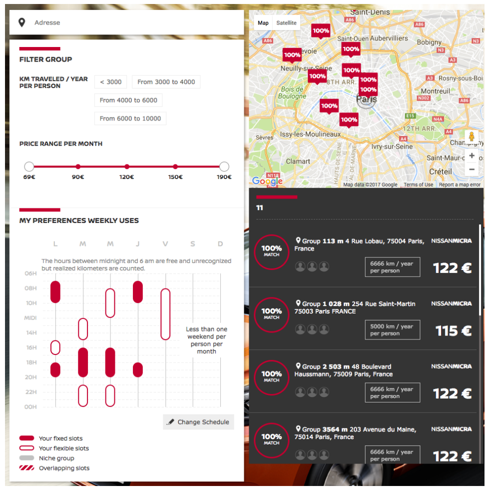


Figure 1: "Nissan Intelligent Get & Go Micra", translated from `www.nissangetandgo.fr`. Filter by group location and schedule. Red colored slots are fixed and empty slots are flexible in the schedule. After joining the group, one can use a mobile application to use the shared vehicle.

programs in North America and the total members reached approximately 1.93 million people (Shaheen et al., 2018). With the introduction of new self-driving technology, there will be less difference between car sharing programs and taxi or ride sourcing services [1] since drivers are no longer needed. Given the huge number of people using car sharing and ride sourcing services all over the world, it will greatly change vehicle ownership and the landscape of car sharing programs in the near future with AV technology. The vehicle ownership structure in the US has been one vehicle per one licensed driver for almost a century (Sivak, 2013; Litman, 2014) and it is worthwhile to investigate how AVs will change people's attitude towards vehicle ownership and what it entails for our transportation infrastructure and systems.

Considering the potential changes in how people currently own and operate vehicles, we investigate new forms of ownership that may be introduced in the market for AVs, called fractional ownership or co-ownership. We use the term of *vehicle co-ownership program* to refer to these kinds of programs that allow people sharing vehicles by owning it together (or *vehicle co-leasing program* for leasing vehicles together). These programs aim in the gray area between car sharing and private vehicle ownership.

Recently, car manufacturers have experimented with pilot co-ownership programs with some restrictions in different locations, including *Audi Unite*, *Ford Credit Link* and etc. *Nissan Intelligent Get & Go Micra* used intelligent social-media profile-matching to form car sharing communities, allowing users to co-own a new Nissan Micra (Nissan, 2018). As shown in Figure 1, customers were able to fill in their traveling area, budget, as well as the intended travel schedule to join (or create) a group with reasonable schedule overlaps and costs. These services are no longer operational as of May 2019.

[1] Ride sourcing, different from ride sharing where additional travelers are added to a pre-existing trip, is provided by transportation network companies like Uber and Lyft.

While the success of this new form of vehicle ownership was limited with conventional vehicles, it certainly becomes a more viable option with the coming AV technology. The distance between parking location and travel destination could be covered easily 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. Therefore, we envision that AVs will be co-owned widely and new markets will be created accordingly.

Motivated by this idea, we focus on the single-autonomous-vehicle ownership between multiple people in this paper. Although some recent works studied shared autonomous vehicle usage or ownership (Fagnant and Kockelman, 2014; Masoud and Jayakrishnan, 2016; Lavieri et al., 2017; Nair et al., 2017; Haboucha et al., 2017; Winter et al., 2017), we are the first to propose an autonomous vehicle co-leasing program as a new form of car sharing system and our work focus on the matching of users participating in these new shared vehicle ownership programs.

In this paper, we introduce a problem of matching multiple users to share an autonomous vehicle, and the number of people in a group is flexible. In our problem, we assume that a group of unknown number of people are willing to share autonomous vehicles with others and they need to be fairly assigned to different groups according to their daily travel schedule. We formulate our matching problem as a *generalized stable matching problem* to ensure the fairness of the match among all users, because evidence from previous works shows that stable matching helps ensure the acceptance of the market (Roth, 1984, 1991; Winter et al., 2017). Gale and Shapley (1962) first introduced stable matching along with three classic stable matching problems, including stable marriage problem, stable roommates problem and hospitals/residents problem. Stable marriage problem finds a stable marriage matching between equally sized sets of men and women, where no two people of the opposite sex both prefer each other than their current partners. Similarly, stable roommate problem finds a stable matching of two people sharing one room among an even-sized set of people. Given such unknown group size and total number of groups, our problem could be considered as a multi-dimensional extension of the stable roommate problem since we want to assign users to vehicles.

Although there are some previous works studying 3-dimensional stable matching problems, none of them provided a generalized multi-dimensional stable matching formulation or algorithm that allows both flexible group size and incomplete preference list. Knuth (1997) proposed a 3-dimensional stable marriage problem with three sets of agents. Ng and Hirschberg (1991) and Subramanian (1994) considered various definitions of stability and preference lists restrictions and shows the NP-completeness of the three dimensional models. Boros et al. (2004) and Eriksson et al. (2006) studied the complexity of cyclic 3D stable matching, while 3-dimensional stable roommate problem is also proved to be hard to solve (Huang, 2007; Iwama et al., 2007; Cordeau et al., 2007).

Recently, Wang et al. (2017) introduced stable matching for the two-sided market in dynamic ride-sharing. They focused on the bipartite matching model between drivers and riders, searching for stable or nearly stable matches with incomplete lists instead of extending the matching problem to multi-dimensional cases. Similarly, we believe it is reasonable to include matching stability to ensure the fair experience for users in autonomous vehicle co-ownership programs.

In our work, we focus on the matching between multiple users, and the size of the group is also flexible depending on the preference of users. We consider the stability of matching in our problem, trying to find an optimal group assignment that all people can be satisfied as much as possible. We summarize the major contributions as follows:

- We propose a mathematical formulation for the generalized, multi-dimensional stable matching problem, applied in user matching for autonomous vehicle co-ownership programs.
- We define the personal disutility in the same group, which is solved based on a single vehicle scheduling problem with soft time windows.
- We design a heuristic method for our problem and discuss the special cases that are ignored in our method.
- The matching results should provide insights on the reshape of vehicle ownership, the changes in travelers' decisions as well as the creation and design of new markets.

The remainder of this paper is organized as follows. In section 2, we introduce our generalized stable user matching model formulation as well as our autonomous vehicle scheduling problem to measure group members' disutility. In section 3, we present our heuristic method to solve the model with the introduction of the concept of valid groups. In section 4, we provide computational time analysis as well as scenario analysis based on various alternative matching objectives. Section 5 concludes the paper and provides directions for future research.

2 Model formulation

We present a generalized optimization formulation of the stable user matching problem for autonomous vehicle co-ownership programs. Consider a group of people as the potential users of autonomous vehicle co-ownership program, we will match the population as groups of multiple people. In this problem, we assign each person into a group so that members in the same group can share an autonomous vehicle. The problem is equivalent to the classic stable roommate problem if each group contains two members. However, it is reasonable to allow multiple people to stay in the same group. Thus, we assume that the number of people in a group is not limited and the sizes of groups can be different. Our problem considers the stability of group assignment (or the fairness of user matching) comparing to the classic set packing problem that maximizes the number of pairwise disjoint subsets.

2.1 Generalized formulation of stable user matching problem

Stable matching theory (Roth, 2008) has been a popular topic in different areas including economics, computer science, mathematics, etc. Algorithms of solving classic stable matching problems as well as their extensions have been widely studied. The stable marriage problem focuses on two-sided matching between men and women, and a deferred acceptance algorithm was proposed in Gale and Shapley (1962) to find a stable set of marriage. Later, an efficient polynomial-time algorithm was given in Irving (1985) to solve the stable roommate problem as well as to check the existence of a stable matching solution. Other extensions were also studied in previous works, including stable matching problems with incomplete preference lists and ties in preference lists (Ronn, 1986, 1990; Irving, 1994).

In addition to solving stable matching problems with algorithms, previous works also studied stable matching problems from the polyhedral perspective as well as fractional stable matching using linear programming (Vate, 1989; Roth et al., 1993; Abeledo and Rothblum, 1994; Abeledo and Blum, 1996; Teo and Sethuraman, 1998, 2000; Vohra, 2012). Iwama and Miyazaki (2008) presented a survey of stable matching problems as well as other studies of interesting extensions, including Abraham et al. (2005)'s work on the solution with a minimum number of blocking pairs and multi-dimensional stable matching problems.

Considering the flexibility in group size (multi-dimensional) as well as the acceptance of people in the same groups (incomplete preference lists), our user matching problem is more complicated than the classic stable roommate problem as well as its variants studied in previous literature. We extend the stable roommate problem to multi-dimensional cases and formulate our generalized stable user matching problem, based on the following definitions. The notation of our model is listed in Table 1.

Definition 1 (Match)**.** If each person is assigned to a group, the assignment of all disjoint groups containing all people is called a *match* Ω.

Definition 2 (Blocking group)**.** Suppose we have a group k that is not in the current match Ω. If all persons who constitute group k prefer to switch from their current assigned groups to group k, we call group k a *blocking group* under match Ω.

Definition 3 (Stable match)**.** A match $\overline{\Omega}$ is called a stable match if there is no blocking group under the current match.

Definition 4 (Disutility)**.** We use disutility δ to denote how people are dissatisfied with the current matched group. We use δ_m to denote the disutility for person m in a single-person group (staying alone); we use δ_m^k to denote the disutility for person m in a multiple-person group k. Obviously, we have $\delta_m = \delta_m^k$ for person m staying alone when $|k| = 1$.

Definition 5 (Preference)**.** Suppose we have two groups k_1 and k_2 both containing person m as a member, if we have $\delta_m^{k_1} < \delta_m^{k_2}$, we say person m *prefers* group k_1 to group k_2; if we have $\delta_m < \delta_m^{k_1}$, we say person m *prefers* staying alone to joining group k_1. We use $k_1 \succ_m k_2$ to denote that person m prefers group k_1 to group k_2.

Definition 6 (Acceptable group)**.** We say group k_1 is *acceptable* to person m when we have $\delta_m \geq \delta_m^{(k_1)}$, and group k_1 is an *acceptable group* when group k_1 is acceptable to all people in the group. Otherwise, the group is *unacceptable.*

Table 1: Notation for generalized stable user matching model

Ω	A current match
$\overline{\Omega}$	A stable match
M	Set of all people
G	Set of all possible groups
$\|k\|$	Total number of people in group k
G_n^k	Set of unfavored groups of person n, compared to group k
G_m	Set of groups containing person m
δ^k	Total disutility of all people in group k, $\delta^k = \sum_{m \in k} \delta_m^k$
x_k	Decision variable: $x_k = 1$ if group k is chosen in the current match Ω; otherwise $x_k = 0$

The unfavored group set G_n^k of person n in group k can be obtained as follows. Suppose we have the disutility of person n in group k as δ_n^k, then all groups k' satisfying the constraint $\delta_n^{k'} > \delta_n^k$ form the unfavored group G_n^k. That is,

$$G_n^k = \{k' \in G_n : k \succ_n k'\} = \{k' \in G_n : \delta_n^k < \delta_n^{k'}\}$$

In other words, person n prefers to stay in group k rather than staying in any of the groups in G_n^k.

We have the following proposition:

Proposition 1. Any group in a stable match $\overline{\Omega}$ is an acceptable group.

Proof. Suppose we have a current match Ω, person m will prefer to stay alone if he/she has higher disutility in the current assigned group. According to the definition of blocking groups, the group with only person m, which is apparently not in the current match, becomes a blocking group and prevents the current match from being stable. □

Roth et al. (1993) showed that a stable marriage matching could be expressed as an integer solution of an optimization problem. Later, the stable roommate problem presented in a linear inequality system were also studied by Abeledo and Rothblum (1994) and Teo and Sethuraman (2000). Based on our former notation, we can rewrite the mathematical formulation of stable roommate problem as follows:

$$\sum_{k \in G_m} x_k = 1 \qquad \forall m \in M \tag{1}$$

$$x_k + \sum_{n \in k} \sum_{p \in G_n^k} x_p \leq 1 \qquad \forall k \in G \tag{2}$$

$$x_k \in \{0, 1\} \qquad \forall k \in G \tag{3}$$

Note that here each group k always contains two people (a matching pair). For each group $k = \{i, j\}$, we can write (2) as

$$x_k + \sum_{p \in G_i^k} x_p + \sum_{p \in G_j^k} x_p \leq 1,$$

or

$$x_{\{i,j\}} + \sum_{l \in M : j \succ_i l} x_{\{i,l\}} + \sum_{l : i \succ_j l} x_{\{j,l\}} \leq 1$$

which is equivalent to the formulation of Teo and Sethuraman (2000).

We provide a generalized formulation of stable user matching problem for $|M|$ people based on the idea of stability constraint (2) as follows:

$$\min z = \sum_{k \in G} \delta^k x_k \tag{4}$$

subject to:

$$\sum_{k \in G_m} x_k = 1 \qquad \forall m \in M \tag{5}$$

$$x_k + \frac{1}{|k| - 0.5} \sum_{n \in k} \sum_{p \in G_n^k} x_p \leq 1 \qquad \forall k \in G \tag{6}$$

$$x_k \in \{0, 1\} \qquad \forall k \in G \tag{7}$$

The objective function (4) is to minimize the total disutility of all users in the optimal match given each user is assigned to a group. In other words, disutility is the weight of different groups and we are looking for a group assignment solution to satisfy all people as much as possible. Constraint (5) ensures that each person is assigned to only one group. Easily we know that all chosen groups ($x_k = 1$) are disjoint, since none of them contains the same person.

Note that the classic stable roommate problem can be formulated as a special case of our problem where all acceptable groups contain exactly two members based on the following proposition:

Proposition 2. When $|k| = 2$ for all $k \in G$, our generalized stable matching constraint (6) is equivalent to (2).

Proof. We first note that $G_n^k \subset G_n$. Therefore, (1) or (5) implies

$$\sum_{p \in G_n^k} x_p \leq 1 \quad \forall n \in k, k \in G. \tag{8}$$

When $x_k = 1$, both (6) and (2) hold if and only if $\sum_{n \in k} \sum_{p \in G_n^k} x_p = 0$. On the other hand, when $x_k = 0$, then the quantity $\sum_{n \in k} \sum_{p \in G_n^k} x_p$ needs to be less than or equal to 2, since (8) holds; therefore, there are three possibilities: 2, 1, or 0. We examine one by one.

When it is 2, we have constraint (6) as $0 + 2/1.5 > 1$ and constraint (2) as $0 + 2/1 > 1$, and both constraints are violated. When it is 1, we have constraint (6) as $0 + 1/1.5 \leq 1$ and constraint (2) as $0 + 1/1 \leq 1$, and both constraints are satisfied. When it is 0, we have constraint (6) as $0 + 0/1.5 \leq 1$

and constraint (2) as $0 + 0/1 \leq 1$, and both constraints are satisfied. Thus, constraints (6) and (2) are equivalent when we have $|k| = 2$ in a stable matching problem. □

In our formulation, we use $|k|-0.5$ in the denominator to prevent the denominator from becoming zero when $|k| = 1$, since we allow single-person groups in our user matching problem.

We also obtain the following proposition:

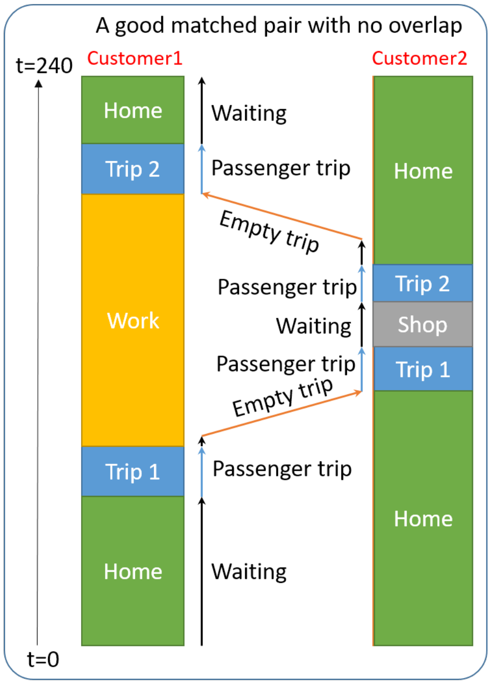
Proposition 3. Constraint (6) ensures that there will be no blocking group in any feasible match.

Proof. For each possible group k in any given feasible match, we have two cases: $x_k = 1$ or $x_k = 0$. In the first case, group k is in the given match and we have $x_k = 1$. Observe that all groups from G_n^k will share at least one person with group k for any $n \in k$, which implies that $\sum_{n \in k} \sum_{p \in G_n^k} x_p = 0$ according to constraint (5). Therefore, constraint (6) is satisfied.

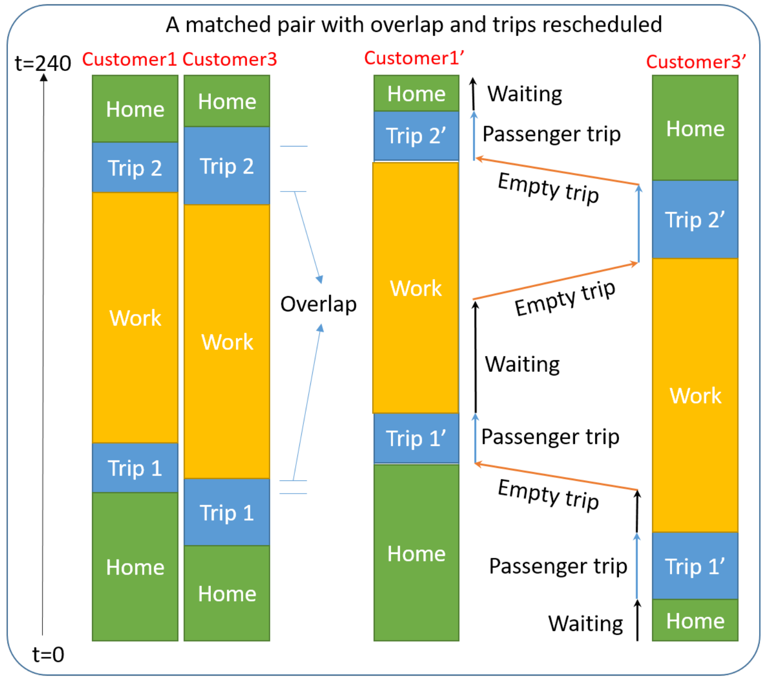
In the second case, group k is not in the given match and we have $x_k = 0$. We need to make sure that acceptable group k is not a blocking group. We need at least one person of group k preferring staying in the currently matched group to moving into group k. According to the definition of unfavored group set, if we have $\sum_{p \in G_n^k} x_p = 1$ for person n, then person n is actually assigned into a group where he/she has higher disutility and he/she will prefer to switch to the new group k. If all people in group k prefer to move from their current matching group to group k, we have $\sum_{n \in k} \sum_{p \in G_n^k} x_p = |k|$. This means $\frac{|k|}{|k|-0.5} > 1$ and the constraint (6) will be violated. Otherwise, we will have $\frac{|k|-1}{|k|-0.5} < 1$ if a person decides to stay in the current matching group and constraint (6) is satisfied. Obviously, constraint (6) always holds when multiple persons decide to stay in the current matching group. Thus, there will be no blocking groups when all constraints are met. □

Our generalized formulation allows more flexible settings in the matching problem and the maximum number of people allowed in a group can be easily adjusted by changing the set of G in our model. Mathematical formulations of the classic stable matching problems have been studied in previous studies (Roth et al., 1993; Wang et al., 2017) to find the optimal match among alternative stable matches. However, those works focused on the stability in two-sided matching markets or matches of two elements. We extend the problem so that our formulation can deal with flexible size of elements in each group of a match. Since our formulation is based on acceptable groups, the assumption of incomplete preference lists with ties are also considered.

The measuring of disutility in our user matching problem is easier since we can observe the time and monetary cost based on trip assignment results in each matching group. It is difficult to measure the quality of matching in classic stable matching problems due to lack of appropriate measuring of individual utilities in the system (Wang et al., 2017). Thus, the optimal stable matching results of our generalized problem are more meaningful than the optimal matching results based on preference rankings. Our current formulation of the stable matching problem as well as its possible variants with reasonable size can be easily solved in optimization solvers. We have no need to design different solving algorithms for variants of the user matching problem where only simple modifications are made. Thus, designing and implementing scenario analysis are much easier with our generalized formulation of stable user matching problem.



(a) A good match



(b) A match with overlap

Figure 2: Matching pair examples

2.2 Measuring disutility of users within a group: Autonomous Vehicle Scheduling Problem

With our generalized stable matching problem, we need a good measurement of users' disutility in order to find the optimal match that satisfies all people as much as possible. This disutility (or utility) represents potential inconveniences (conveniences) of sharing a vehicle, and there may exist multiple types of disutility, including time and monetary considerations from users. Assuming that travelers have a representative set of trips to make, and they can request pick-up and drop-off from a shared AV, there can be conflicting schedules between members from the same group, and disutility should be able to describe how users react to those conflicts. Consider a simple example of two people in a group, we present a visualization of two possible cases in Figure 2. As we can see in Figure 2a, there is no overlap between two persons, and their activity schedules remain the same. On the contrary, two persons need to negotiate to reschedule their activities so that one autonomous vehicle can serve all their trips in Figure 2b. People with a shifted schedule should be less satisfied, and we use the time penalty to consider this potential disutility of shifted activities. Consider another example that two people are sharing a vehicle, both people have short-distance trips while their trip locations are far away from each other. This may lead to a high fuel cost for both of them due to the long empty trips, and the mileage of the vehicle may increase faster than usual, leading to higher insurance and maintenance fees. Thus, empty trip costs should be seen as disutility as long as the fuel and vehicle maintenance fees are shared among all users.

We can easily extend these cases of trip scheduling for a group of multiple persons. In order to quantify the time and monetary cost of the users, we define the disutility δ_m of a person m in single-person group as well as the disutility δ_m^k of a person m in multiple-person group k as follows:

$$\delta_m = C_{\text{lease}} \tag{9}$$

$$\delta_m^k = \frac{\sigma_m}{\sigma^k}\left(C_{\text{lease}} + r_1 D_{\text{empty}}^k\right) + r_2 P_m^k \tag{10}$$

Here, we use C_{lease} to denote the fixed vehicle leasing cost. For a person in a single-person group, he/she is staying alone with no time penalty or empty vehicle trips, so there is only vehicle leasing cost in personal disutility. We assume that members in multiple-person groups will split the vehicle leasing cost C_{lease} and the cost of empty trips D_{empty}^k based on their percentage time/distance usage of the vehicle. More discussions on pricing rules are presented in Appendix A. Also, there will be additional time penalty cost P_m^k for each person m due to the rescheduling of the trips when overlap exists. We use r_1 to denote vehicle cost per mile or per time unit, r_2 to denote penalty of waiting, σ_m to denote the total time spent on traveling of person m and σ^k to denote the total time spent on traveling of all people in group k. Obviously, if a person has higher disutility, he/she is less satisfied with the matching result; if a person has lower disutility, he/she is more satisfied with the matching result. Based on the disutility of each user in all possible groups, we can find the acceptable groups in all possible groups to form an optimal stable match with our generalized stable matching model.

In order to optimally schedule all trips in a group and extract each member's disutility, we introduce an autonomous vehicle scheduling problem (AVSP) based on the pickup and delivery model of Sexton and Choi (1986). AVSP assume that travelers have a set of predefined trips, that is presentative of their typical schedules, to be served by the AV, and travelers are willing to change the trip starting times within some fairness conditions among the users in case of conflicts. In reality, this may not hold true. For example, some trips have hard time windows and some trips may be flexible as long as it is performed. We note that, the scheduling problem among multiple users with fairness consideration itself is a difficult problem (Allahviranloo and Chow, 2019; Khayati et al., 2019a,b), however, in the proposed stable user matching problem of AV co-ownership, the results of scheduling problem is represented with disutility (preferences) parameter values. Hence we resort to a base case of fixed trip request for establishing a user trip scheduling model, AVSP, for the illustration of the proposed research. In Section 5, we additionally discuss potential future research on how to replace this AVSP problem with more sophisticated scheduling models.

In the proposed AVSP, we apply the soft time window assumption that the starting time and ending time of people's activity can be shifted while the total time spent on the activity will remain unchanged. Given soft time window constraints, rescheduled trips are accepted with time penalty as part of personal disutility. Empty autonomous vehicles can travel between different locations themselves, and the travel cost is considered as empty trip costs in personal disutility. With the given disutility definition, our goal is to satisfy all users in a group and fairly reschedule all trips optimally. Ride sharing between the group members is not considered currently, which means only

Table 2: Notation for AVSP of group k

V_m	Set for all trips of person $m \in k$
V	Set for all trips of all people in the group, $V = \bigcup_{m \in k} V_m$
W_m	Set for all adjacent trip pairs of person $m \in k$
W	Set for all adjacent trip pairs of each person in the group, $W = \bigcup_{m \in k} W_m$.
N	Total number of trips
t_{ij}	Travel time from trip i's destination to trip j's origin
s_i	Travel time of trip i
τ_i^0	Scheduled departing time of trip i
u_i^0	Scheduled arriving time of trip i
p_i	Penalty coefficient for trip i
$f_i(\tau)$	Actual shifted time of trip i departing at time τ
τ_i	Decision variable, actual departing time of trip i
u_i	Decision variable, actual arriving time of trip i
x_{ij}	Decision variable, related to the sequence of trip i and trip j
y_{ij}	Decision variable, related to the sequence of trip i and trip j
v_{gap}	Decision variable, maximum time penalty gap between users in the current matching group

one person will be served by the autonomous vehicle at the same time. The notation for AVSP is presented in Table 2.

We define the following variable:

$$x_{ij} = \begin{cases} 1 & \text{if trip } i \text{ is served before trip } j\text{, and the end of trip } i \text{ precedes the start of trip } j \\ 0 & \text{else trip } i \text{ is served after trip } j\text{, and the end of trip } j \text{ precedes the start of trip } i \end{cases} \tag{11}$$

Thus, if trip i is finished earlier and the trip j is finished later, we will have $x_{ij} = 1$. Note that only one trip can be served at a time, thus the two cases here in the definition are mutually exclusive and collectively exhaustive. However, this does not mean trip i and trip j are two trips served by one vehicle adjacently. Consider the first trip i and the last trip j of all trips, we will still have $x_{ij} = 1$. With the variable x_{ij}, we can easily define a new variable y_{ij} as follows:

$$y_{ij} = \begin{cases} 1 & \text{if trip } i \text{ and trip } j \text{ are adjacent, and trip } i \text{ is served immediately before trip } j \\ 0 & \text{else} \end{cases} \tag{12}$$

Consider the matrix of x_{ij}, the number of trips later than trip i can be expressed as $\text{rank}(i) = \sum_k x_{ik}$. Obviously, trips with higher rank value will be finished earlier and trips will be adjacent if the rank difference is 1. Thus, we can have the following relationship:

$$y_{ij} = 1 \ \text{ if and only if } \ \sum_{k \in V} x_{ik} - \sum_{k \in V} x_{jk} = 1 \tag{13}$$

Based on the notation for AVSP of the current group k, we know t_{ij} is the empty vehicle travel time to connect adjacent trips i and j when we have $y_{ij} = 1$. Given a match with fixed y_{ij} values, we can obtain the empty vehicle travel time as follows:

$$D^k_{\text{empty}} = \sum_{i\in V}\sum_{j\in V} y_{ij}t_{ij} \tag{14}$$

The time penalty for a person m in group k is as follows:

$$P^k_m = \sum_{i\in V_m} p_i f_i(\tau_i) \tag{15}$$

With given values of s_i, we have $\sigma^k = \sum_{i\in V} s_i$ and $\sigma^k_m = \sum_{i\in V_m} s_i$ as known values. According to equation (10), we have the disutility of person m in group k under the trip assignment as:

$$\delta^k_m = \frac{\sigma_m}{\sigma^k}\left(C_{\text{lease}} + r_1\sum_{i\in V}\sum_{j\in V} y_{ij}t_{ij}\right) + r_2\sum_{i\in V_m} p_i f_i(\tau_i) \tag{16}$$

We want to satisfy all users in a group as much as possible, so we want to minimize the total disutility of all users in the current matching group. Also, we consider the fairness between different users in one group by limiting the maximum time penalty gap between all users to avoid the condition that some people suffer much higher delay than others. Thus, the objective function of AVSP is formulated as follows:

$$\min \alpha_1 \sum_{m=1}^{|k|} \delta^k_m + \alpha_2 v_{\text{gap}} \tag{17}$$

Since we have $\displaystyle\sum_{m=1}^{|k|} \frac{\sigma_m}{\sigma^k} = 1$, the first part of the objective function can be expressed as follows:

$$\sum_{m=1}^{M} \delta^k_m = C_{\text{lease}} + r_1\sum_{i\in V}\sum_{j\in V} y_{ij}t_{ij} + r_2\sum_{i\in V} p_i f_i(\tau_i) \tag{18}$$

We can ignore C_{lease} in the objective function of AVSP since it is a constant.

Suppose we have M persons in the current group k, we have the following formulation for AVSP:

$$\min \alpha_1\left(r_1\sum_{i\in V}\sum_{j\in V} y_{ij}t_{ij} + r_2\sum_{i\in V} p_i f_i(\tau_i)\right) + \alpha_2 v_{\text{gap}} \tag{19}$$

subject to:

$$u_i - \tau_i = s_i \qquad \forall i \in V \tag{20}$$

$$\tau_j - u_i \geq \tau^0_j - u^0_i \qquad \forall (i,j) \in W \tag{21}$$

$$\tau_j - u_i + M(1 - x_{ij}) \geq t_{ij} \qquad \forall i,j \in V, i \neq j \tag{22}$$

$$\tau_i - u_j + Mx_{ij} \geq t_{ji} \qquad \forall i,j \in V, i \neq j \tag{23}$$

$$\sum_l x_{il} - \sum_l x_{jl} - 1 \leq N(1 - y_{ij}) \qquad \forall i,j \in V, i \neq j \tag{24}$$

$$-\Big(\sum_l x_{il} - \sum_l x_{jl} - 1\Big) \leq N(1 - y_{ij}) \qquad \forall i,j \in V, i \neq j \tag{25}$$

$$\sum_i \sum_j y_{ij} = N - 1 \tag{26}$$

$$\sum_{i \in V_{m_1}} p_i f_i(\tau_i) - \sum_{i \in V_{m_2}} p_i f_i(\tau_i) \leq v_{\text{gap}} \qquad \forall m_1, m_2 \in M \tag{27}$$

$$-\Big(\sum_{i \in V_{m_1}} p_i f_i(\tau_i) - \sum_{i \in V_{m_2}} p_i f_i(\tau_i)\Big) \leq v_{\text{gap}} \qquad \forall m_1, m_2 \in M \tag{28}$$

$$y_{ii} = 0,\ x_{ii} = 0 \qquad \forall i \in V \tag{29}$$

$$x_{ij}, y_{ij} \in \{0, 1\} \qquad \forall i,j \in V \tag{30}$$

$$\tau_i, u_i \geq 0 \qquad \forall i,j \in V \tag{31}$$

We can define the shifted time $f_i(\tau_i)$ as well as the corresponding penalty coefficient p_i as follows:

$$f_i(\tau_i) = \begin{cases} \tau_i - \tau_i^0 & \text{when } \tau_i - \tau_i^0 \geq 0\text{, and the trip is delayed} \\ \tau_i^0 - \tau_i & \text{when } \tau_i - \tau_i^0 \leq 0\text{, and the trip is earlier} \end{cases} \tag{32}$$

$$p_i = \begin{cases} 1 & \text{when } \tau_i - \tau_i^0 \geq 0\text{, and the trip is delayed} \\ 0.6 & \text{when } \tau_i - \tau_i^0 \leq 0\text{, and the trip is earlier} \end{cases} \tag{33}$$

Here, we assume that late arrival penalty is greater than late penalty and the coefficient p_i is different for delayed or early trips (Small, 1982). We can linearize $p_i f_i(\tau_i)$ with linear variables f_i^l, f_i^e and a binary variable f_i' as $p_i f_i(\tau_i) = f_i^l + 0.6 f_i^e$ with following constraints:

$$f_i^l \geq \tau_i - \tau_i^0 \qquad \forall i \in V \tag{34}$$

$$f_i^e \geq \tau_i^0 - \tau_i \qquad \forall i \in V \tag{35}$$

$$f_i^l \leq M f_i' \qquad \forall i \in V \tag{36}$$

$$f_i^e \leq M(1 - f_i') \qquad \forall i \in V \tag{37}$$

$$f_i^l - (\tau_i - \tau_i^0) \leq M(1 - f_i') \qquad \forall i \in V \tag{38}$$

$$f_i^e - (\tau_i^0 - \tau_i) \leq M f_i' \qquad \forall i \in V \tag{39}$$

$$\tau_i - \tau_i^0 \leq M f_i' \qquad \forall i \in V \tag{40}$$

$$\tau_i^0 - \tau_i \leq M(1 - f_i') \qquad \forall i \in V \tag{41}$$

Obviously, when $\tau_i - \tau_i^0 \geq 0$, we have $f_i' = 1$ and $f_i^l = \tau_i - \tau_i^0,\ f_i^e = 0$; when $\tau_i - \tau_i^0 \leq 0$, we have $f_i' = 0$ and $f_i^l = 0,\ f_i^e = \tau_i^0 - \tau_i$. Thus, we can linearize the nonlinear function $f_i(\tau_i)$ in the objective function.

Constraint (20) ensures that the travel duration time of trip i remains the same even if the starting time is shifted. Constraint (21) ensures that the activity duration time between two adjacent trips of the same person are not shortened. Constraints (22) and (23) ensure that x_{ij} will follow our variable definition. Suppose we have $x_{ij} = 1$, we know $\tau_j - u_i \geq t_{ij}$ based on constraint (22), and constraint (23) is redundant. Thus, we have enough time to finish the empty trip from trip i's destination to trip j's origin so that trip j can be served after trip i. On the other hand, if we have $x_{ij} = 0$ and redundant constraint (22), we have $\tau_i - u_j \geq t_{ji}$ indicating trip j happens before trip i with enough empty trip traveling time, following the definition of x_{ij}. Constraints (24), (25) and (26) ensure that the value of y_{ij} follows the definition (13). Constraints (24) and (25) ensure that $y_{ij} = 0$ if $\sum_l x_{il} - \sum_l x_{jl} - 1 \neq 0$, which means trip j does not follow trip i. Constraint (26) ensures that y_{ij} will always take value of 1 if possible and the summation will be $N - 1$ since there are N trips. Constraints (27) and (28) ensure that v_{gap} is the maximum gap between all users in the current group.

We can easily obtain the personal disutility δ_m^k of each person m in group k by solving this model, and the acceptable groups of each person can be obtained given personal disutility of each matching group. The empty vehicle travel cost and time penalty cost of each person can also be obtained easily, which could help us estimate the quality of the matching result.

2.3 Characteristics of stable user matches

We know that a stable match may not exist for stable roommate matching, and similarly the conclusion can be extended in our generalized stable user matching problem.

Consider 4 people (A, B, C, D), whose rankings are: $A : (B, C, D, A)$, $B : (C, A, D, B)$, $C : (A, B, D, C)$, $D : (A, B, C, D)$. Here, we assume that each person prefers to be matched than staying alone. In classic stable roommate matching, we assign people into pairs of two, and we have potential matches of (AB, CD), (AC, BD) and (AD, BC). For the matching pair (AB, CD), person C and person B prefer to stay together instead of staying in the current group, making BC a blocking pair. Similarly, we can see all potential matches are blocked from being stable and a stable match for the problem does not exist.

Suppose we apply our generalized model with the same preference list for the 4 people, we list all potential groups with size less than three as well as their unfavored group sets as follows:

According to our generalized formulation, if we choose more than two groups from a collection of unfavored group sets (same row in Table 3) in a match, there exist a blocking group that prevent the current match from being stable. For example, we can see the match (AC, BD) is blocked by AB since AC and BD both appear in the unfavored group sets of AB. The blocking groups for all potential matches (group size ≤ 2) are listed as follows: Obviously, we can see that it is still possible that a stable match does not exist for our generalized matching problem.

In our generalized formulation of user matching problem, we have constraint (5) to ensure each person will be matched and the problem becomes infeasible when such a stable match does not exist.

Table 3: Unfavored group sets for all groups

Potential groups	Unfavored group sets
A	$\{\phi\}$
B	$\{\phi\}$
C	$\{\phi\}$
D	$\{\phi\}$
AB	{AC, AD, A, BD, B}
AC	{AD, A, BC, CD, C}
AD	{A, BD, CD, D}
BC	{AB, BD, B, CD, C}
BD	{B, CD, D }
CD	{C, D }

Table 4: Blocking groups for all matches

Potential matches	Blocking groups
(AB, CD)	{BC}
(AC, BD)	{AB}
(AD, BC)	{AC}
(AB, C, D)	{CD, BC}
(AC, B, D)	{AB, BD}
(AD, B, C)	{AB, AC, BC}
(BC, A, D)	{AC, AD}
(BD, A, C)	{AB, BC, AC}
(CD, A, B)	{AC, BC, AB}
(A, B, C, D)	{AB, AC, AD, BC, BD, CD}

Table 5: Unfavored group sets for all groups

Potential groups	**Unfavored group sets**
AB	{AE, AD, A, BD, BE, BC, B}
AC	{AB, AE, AD, A, C}
AD	{A, D}
AE	{AD, A, BE, CE, DE, E}
BC	{B, CE, AC, C}
BD	{BE, BC, B, CD, DE, AD, D}
BE	{BC, B, CE, DE, E}
CD	{BC, CE, AC, C, DE, AD, D}
CE	{AC, C, DE, E}
DE	{AD, D, E}

We make the following observations:

Proposition 4. Consider a stable match $\overline{\Omega}$ for a set of people M and a subset $M_1 \subseteq M$, and let G_1 denote the set of all groups for people in M_1. We say Ω_1 is a *sub-match* of $\overline{\Omega}$, if $k \in \Omega_1$ implies $k \in \overline{\Omega}$, denoted by $\Omega_1 \subseteq \overline{\Omega}$. Every sub-match Ω_1 for M_1 is a stable match among groups in G_1.

Proof. We prove by contradiction. Suppose we have a sub-match Ω_1 of a stable match $\overline{\Omega}$, and Ω_1 is not stable. Then, there must exist a blocking group $k \in G_1$ for Ω_1. In other words, each person m in k prefers to switch from their current group in Ω_1. Since the members of group k are from M_1 and blocking group k is not chosen in match Ω_1, we know group k is also not chosen in stable match $\overline{\Omega}$. However, each person in group k will also prefer to switch from their current group since they have better choice among $M_1 \subseteq M$. Thus, k will also be a blocking group for $\overline{\Omega}$. This contradicts with the assumption that $\overline{\Omega}$ is stable. Thus, we can conclude that any sub-match of a stable match is also stable. □

Proposition 5. A sub-match of an optimal stable match may not be optimal.

We can solve our generalized matching problem to find an optimal stable match to satisfy the objective of the problem. Since the sub-match of a stable match is equivalent to removing the variables and constraints related to all removed people, the optimal stable match can change given such a new problem.

Consider 5 people (A, B, C, D, E), whose rankings are: $A : (C, B, E, D, A)$, $B : (A, D, E, C, B)$, $C : (D, B, E, A, C)$, $D : (B, C, E, A, D)$. Here, we assume that each person prefers to be matched than staying alone. We list all potential groups with size two (unfavored group sets are empty for single-person groups) as well as their unfavored group sets as follows: According to Table 5, we can see that (AB, CD, E) is a stable match (no two groups appear in an unfavored group set of any potential group). Suppose we remove E from the population, we can obtain all potential groups with size two (unfavored group sets are empty for single-person groups) as well as their unfavored group sets as follows: Obviously, (AB, CD) is still a stable match, and we can see that (AC, BD) is

Table 6: Unfavored group sets for all groups

Potential groups	**Unfavored group sets**
AB	{AD, A, BD, BC, B}
AC	{AB, AD, A, C}
AD	{A, D}
BC	{B, AC, C}
BD	{BC, B, CD, AD}
CD	{BC, AC, C, AD, D}

also a stable match. However, (AC, BD, E) is not a stable match since the match is blocked by the group CE. It is possible that the total disutility of match (AC, BD) is lower than that of (AB, CD), and thus we show that a sub-match of an optimal stable match is not optimal in the current case.

Easily, we can use the constraint $\sum_{k \in G_m} x_k = 0$ to denote that person m is pre-selected to be a single-person group, and we can add the disutility of those people after we solve the stable matching problem for the rest of the people.

2.4 Alternative objectives

We believe a good user matching model for autonomous vehicle co-leasing programs should satisfy the need of both sellers and users. According to the current vehicle co-ownership market, most sellers are car manufacturers. Thus, we want to consider the need for either car manufacturers or users based on our detailed definition of user disutility. In addition to the original objective function that minimizes the total disutility of all users, we propose two alternative objectives including maximizing the revenue of vehicle-leasing companies as well as maximizing the total number of groups matched. The following are the different alternative objective functions that can be applied in our generalized matching model to replace z in the objective function (4).

Alternative 1: Maximize the revenue of vehicle-leasing companies.

$$\max z_1 = \sum_{k \in G} \Big(C_{\text{lease}} + r_1 D^k_{\text{empty}} + r_1 \sigma^k \Big) x_k \tag{42}$$

Alternative 2: Maximize the total number of groups matched.

$$\min z_2 = \sum_{k \in G} x_k \tag{43}$$

In alternative objective 1, we maximize the revenue of car manufacturers providing vehicle co-ownership programs, and the revenue is related to the fixed vehicle leasing cost for each group as well as the total group vehicle usage. Note that mileage limitations are usually applied on conventional vehicle leases to avoid the vehicle being abnormally overused by the driver, we assume that manufacturers will include total vehicle travel distance while estimating their utility, so empty

Algorithm 1 Enumerate all potential groups and run AVSP for each group

for Each person $m \in M$ (we use m to denote the index of a person) **do**
    Generate initial group g containing only person m
    Add group g to the set of potential groups Υ_1
for $n < |M|$ **do**
    for Each group $k \in \Upsilon_n$ **do**
        for Each person $m \in M$ **do**
            if $m > m'$ for any person m' in group k **then**
                Add person m to group k to form a new group k'
                Add group k' to the set of potential groups Υ_{n+1}
Generate the set of all potential groups $\Upsilon = \bigcup_{i=1}^{|M|} \Upsilon_i$
for Each group k in Υ **do**
    Solve AVSP for group k
    Record the corresponding personal disutility function values
Calculate unfavored group sets based on personal disutility function values
Solve the stable user matching problem to find optimal match for all users

travel distance of autonomous vehicles should be included while we estimate the revenue of vehicle leasing companies. In alternative objective 2, we maximize the total number of groups matched, which is the same as the classic stable matching problem. We always allow people to stay alone (as 1-person groups) in our problem with different objective functions.

3 Solution method

In this section, we discuss the solving process for our stable user matching problem as well as the autonomous vehicle scheduling problem. In order to find the optimal solution of the stable user matching problem, we need to obtain the personal disutility in each possible group in the objective function as well as the unfavored group sets in the stability constraint. We can solve AVSP for each possible group to obtain the personal disutility, and the unfavored group sets can be chosen based on the known disutility. We summarize the process to solve a stable user matching problem for $|M|$ persons as in Algorithm 1, allowing any number of people to form a group.

In Algorithm 1, we enumerate all possible groups of different sizes and solve AVSP for each group to obtain the required parameters in stable user matching problem. Obviously, the total number of possible groups for $|M|$ people is $2^{|M|} - 1$, since the number of people in a group is not limited. Thus, we can face an enormously large number of potential groups for a small population due to the property of combination. Then, it becomes difficult to solve the whole problem based on our former enumeration Algorithm 1.

3.1 A heuristic method based on valid groups

We introduce the concept of *valid group* to improve the computational time of our enumeration algorithm. We develop a heuristic method to find a solution that is close to optimal for the whole user matching problem based on the valid group.

As we have mentioned earlier in Proposition 1, all groups in optimal stable user matching should be acceptable groups. In other words, we can reduce the total number of possible groups by creating only acceptable groups. If we have fewer possible groups, we can improve the computational time of the whole problem by reducing the number of AVSPs to be solved as we need to run the AVSP for a group to obtain the disutility of each group member to verify whether the current group is an acceptable group. Thus, we define the concept of *valid group* to describe the potential groups that are *likely* to be acceptable groups. If a group is not a valid group, we assume that it is less likely to be an acceptable group. We can abandon those groups in the stable user matching problem since they are unlikely to be chosen in the optimal stable match solution. Naturally, we can imagine the case where a new person m is added to a group k. If person m is not satisfied with joining group k, person m will also be unsatisfied with joining any group containing all members from group k. Based on this idea, we present the following definitions:

Definition 7 (Black list of person m)**.** For a group k, we can form a new group k' by adding a new person m. If any member in group k' prefers to leave group k', we say group k is on the black list of person m.

Here, if any member from group k suffers higher disutility or person m prefers to stay alone, we know that there is at least one member prefers to leave group k'.

Definition 8 (Valid group)**.** Suppose we have a group k containing person m, we use group k' to denote the subgroup containing all group members from group k except person m. If *no subset* of group k' is on the black list of any person m in group k, we say group k is a *valid group.* Furthermore, if group k is not a valid group, we call it an *invalid group.*

Based on these given definitions, we can have the following proposition:

Proposition 6. Any subset of a valid group is also a valid group.

Proof. Suppose we have an invalid group k' as a subset group of a valid group k. According to the definition of invalid group, there exist a subset group $\hat{k}$ of group k' and group $\hat{k}$ is on the black list of some person m in group k'. Then, we find a subset group $\hat{k}$ of group k and group $\hat{k}$ is on the black list of the member m in group k. This contradicts our assumption that group k is a valid group. Thus, we can conclude that any subset of a valid group is also a valid group. $\square$

Here, we assume that valid groups are more likely to be chosen in stable matching results since new members joining the group always improve the disutility of old group members in valid groups. In other words, these groups are the better options and are more likely to be accepted by users.

Algorithm 2 Valid group algorithm

for Each person pair (m, m') in M **do**
  Generate initial group g with person pair (m, m')
  Run AVSP for group g
  if Group g is an acceptable group **then**
    Add g to the set of acceptable groups Υ_2
    Record the corresponding personal disutility function values
  else
    Add m' to the black list of person m
    Add m to the black list of person m'
for $n < |M|$ **do**
  for Each group $k \in \Upsilon_n$ **do**
    for Each person $m \in M$ **do**
      if $m > m'$ for any person m' in group k and m passes valid group check **then**
        Add person m to group k as a new group k'
        Run AVSP for group k'
        if Disutility of all members in group k are higher than that in group k' **then**
          Add group k' to the set of potential groups Υ_{n+1}
          Record the corresponding personal disutility function values
        else
          Add group k to the black list of person m
Calculate unfavored group sets based on personal disutility function values
Solve the stable user matching problem to find optimal match for all users

Suppose we find an invalid group k of two people and let $G(k)$ be the set of all groups that include group k; that is, $G(k) = \{p \in G : k \subseteq p\}$. We know that all groups in $G(k)$ are not valid groups according to Proposition 6, and we assume these groups are less likely to be chosen in the optimal stable match. Then, we decide to abandon these groups and skip the process of evaluating those potential groups to improve computational efficiency.

In this way, we can greatly reduce the total number potential groups to evaluate so that the computational time of the whole solving process can be shortened. Easily we can know that all 1-person groups are valid and acceptable groups, so the minimum number of people in invalid groups becomes two. In the two-person group case, if it is an invalid group, it means one of the person has higher disutility than staying alone according to the definition of valid group. Thus, all valid groups of two people are acceptable groups. We can add 1-person groups to the black lists of each person during the enumeration process of evaluating all 2-person groups. We can use the black list of 1-person groups while evaluating potential 3-person groups. Similarly, 2-person groups are added to the black lists during the process of evaluating 3-person groups. Thus, we could use the black list of each person containing 1-person groups and 2-person groups while evaluating 4-person groups. We call this process *valid group check*. In the enumeration process, the total number of potential groups need to be evaluated is greatly reduced. We present this valid group algorithm as in Algorithm 2.

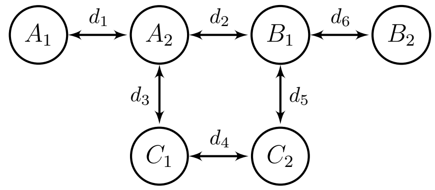


Figure 3: Simple network

3.2 Limitations in our valid group algorithm

Although we can eliminate most of the potential groups that are unacceptable with the valid group algorithm, it is possible that we also eliminate some of the acceptable groups. For example, we have 4 nodes and 3 edges in the following simple network. Suppose we have three persons A, B and C, person A travels from A_1 to A_2; person B travels from B_1 to B_2; and person C travels from C_1 to C_2. Here, we assume that there is no overlap in the trip schedule of all three people, and the only additional cost is the empty vehicle trip. We assume the number on the edge is the travel distance cost and fixed vehicle leasing cost is C_{lease}. Consider the potential group containing person A and B, they are both responsible for part of the vehicle leasing cost and the empty vehicle trip from A_2 to B_1 is split depending on the travel distance of each person. Also, consider the group containing A, B and C, the empty vehicle trip is now from A_2 to C_1 and from C_2 to B_1 with the total cost of $d_3 + d_5$. Thus, we have the following personal cost given the former personal disutility definition:

Two-person match costs: For A: $\frac{d_1(C_{\text{lease}}+d_2)}{d_1+d_6}$; for B: $\frac{d_6(C_{\text{lease}}+d_2)}{d_1+d_6}$

Three-person match costs: For A: $\frac{d_1(C_{\text{lease}}+d_3+d_5)}{d_1+d_4+d_6}$; for B: $\frac{d_6(C_{\text{lease}}+d_3+d_5)}{d_1+d_4+d_6}$; for C: $\frac{d_4(C_{\text{lease}}+d_3+d_5)}{d_1+d_4+d_6}$

Suppose the cost of sharing a vehicle for a person is higher than staying alone (with only vehicle leasing cost C_{lease}), the group containing A and B is an unacceptable group or invalid group. However, suppose all three persons are satisfied (with total cost lower than vehicle leasing cost C_{lease}), the new three-person group is actually an acceptable group. The relationships of the parameters are as follows:

$$\left.\begin{array}{r}\dfrac{d_1(C_{\text{lease}}+d_2)}{d_1+d_6} > C_{\text{lease}}\\ \text{or}\\ \dfrac{d_6(C_{\text{lease}}+d_2)}{d_1+d_6} > C_{\text{lease}}\end{array}\right\} \Rightarrow UB(C_{\text{lease}}) = \max\left\{\frac{d_6 d_2}{d_1}, \frac{d_1 d_2}{d_6}\right\}$$

$$\left.\begin{array}{r}\dfrac{d_1(C_{\text{lease}}+d_3+d_5)}{d_1+d_4+d_6} < C_{\text{lease}}\\ \dfrac{d_6(C_{\text{lease}}+d_3+d_5)}{d_1+d_4+d_6} < C_{\text{lease}}\\ \dfrac{d_4(C_{\text{lease}}+d_3+d_5)}{d_1+d_4+d_6} < C_{\text{lease}}\end{array}\right\} \Rightarrow LB(C_{\text{lease}}) = \max\left\{\frac{d_1(d_3+d_5)}{d_4+d_6}, \frac{d_6(d_3+d_5)}{d_1+d_4}, \frac{d_4(d_3+d_5)}{d_1+d_6}\right\}$$

Suppose we have the vehicle leasing cost C_{lease} falling in the range of $[LB(C_{\text{lease}}), UB(C_{\text{lease}})]$, we will observe an unacceptable two-person group becoming acceptable after a new member joins the group. As a simple example, we can take $C_{\text{lease}} = 100, d_1 = d_6 = 100, d_2 = d_4 = 200, d_3 = d_5 = 1$. Then, we have $LB(C_{\text{lease}}) = 2$ and $UB(C_{\text{lease}}) = 200$, and C_{lease} falls in the range of $[2, 200]$. In our valid group algorithm, if a subgroup is considered invalid, all groups containing the subgroup are eliminated. Thus, the acceptable group of ABC will not be considered as a potential group in the matching step since group AB is invalid. Our valid group algorithm actually eliminates some potential groups that is likely to be chosen in the final matching result. In other cases with different visiting orders between group members, similar relationships between parameters exist and our algorithm will eliminate acceptable groups if those relationships are satisfied. However, the case shown here is very rare in reality and we believe our valid group algorithm performs well considering the computational time saved.

4 Case study results

We apply our models based on the household travel data from the 2000−2001 California Statewide Household Travel Survey provided by the California Department of Transportation (Casas, 2002). The survey included much socio-economic information including trip travel time and activity duration time from 17,040 households from various counties and regions. Since the activity location, as well as home location, are given as coordinates, we can easily calculate the travel time between any two locations. In our case study, we choose 200 persons and 805 trips from the same local area so that the empty vehicle trip time is of acceptable length. All trips from these people are vehicle trips in one single day so that these persons could be potential users of autonomous vehicle co-ownership programs. We assume that all vehicles have an average speed of 35 mph. We assume that vehicle operating cost r_1 is \$1.32 per hour or \$2.64 per mile for an autonomous vehicle with a 24 kWh battery (70 miles) and the electricity price is \$0.11 per kWh (Alternative Fuels Data Center, 2017). We also assume the monthly leasing cost of an electric vehicle is \$200 (Bernstein, 2017), and the daily leasing cost C_{leasing} is \$6.667 given 30 days in a month. Litman (2003) mentioned that the value of travel saving time is around 25% of the average wage and the value of waiting time is usually half of the value of travel saving time when the travel condition is not very uncomfortable for travelers. Thus, we take the value of time penalty estimation r_2 as \$2.745 per hour given the average 2017 US hourly wage is \$21.96 (Trading Economics, 2017). We take all penalty coefficients as 1 to simplify the problem, considering the penalty as the shifted time.

4.1 Scenario analysis

We apply our model on the dataset, hoping to derive insights facing the new co-ownership market. In addition to stable matching, we also used *maximum weight matching* to consider the case where stability is not required in the match. In maximum weight matching problem, we remove the stability constraint and transfer our user matching problem to a set packing problem which can also be solved based on our proposed method. We define the following 6 scenarios given our 3 different objectives (original and 2 alternative ones) and 2 different matching models:

ST Max Revenue: All people will be matched by stable matching while maximizing car leasing company's revenue.

ST Max Matches: All people will be matched by stable matching while maximizing the number of matching groups.

ST Min Utilities: All people will be matched by stable matching while minimizing total personal disutility (original objective).

MW Max Revenue: All people will be matched by maximum weight matching while maximizing the car leasing company's revenue.

MW Max Matches: All people will be matched by maximum weight matching while maximizing the number of matching groups.

MW Min Utilities: All people will be matched by maximum weight matching while minimizing total personal disutility (original objective).

We show the total revenue, total empty vehicle travel time, total personal disutility and total time shifted for all people under the matching result of each model and alternative objective in the following Table 7. According to our former definitions, objective Max Revenue and Max Matches benefits service providers (car manufacturers) while objective Min Utilities benefit users (travelers). We can see the actual personal disutility function values for users in stable matching scenarios are less affected by the objective function. In maximum weight matching scenarios, all users are assigned alone to improve supplier's revenue in objective Max Revenue and Max Matches, resulting in significantly higher personal disutility function values for all users. Thus, we can conclude that stability should be considered in matching problems to ensure a better experience for co-leasing program users.

We also show the number of people in each matched valid group for all scenarios in the following Table 8. As we can see, the number of groups of different members are similar in stable matching scenarios while the group assignments are greatly affected by objective in maximum weight matching scenarios. Also, the number of groups with many members is relatively small and it is reasonable to conclude that there is a practical upper bound for the group size. Computational cost is reduced with less potential groups limited by such constraints. We also add scenarios ST' to denote the stable matching results based on the original three stable matching scenarios after removing single person in stable matches. Since the revenue, empty vehicle travel time, disutility and shifted time can be easily calculated for each single person, we can easily obtain the total values in these attributes for scenarios ST' to compare with scenarios ST in Table 7. We can see that actually the objective is increased in ST' Max Revenue and Max Matches after we remove some of the single-person groups in stable matching results of ST Max Revenue and Max Matches. In addition, the number of people

Table 7: Numerical results for various model and objectives

Case	**Revenue**	**Empty vehicle travel time (h)**	**Personal disutility**	**Shifted time (h)**
MW Max Revenue	1767.28	0	1333.33	0
MW Max Matches	1767.28	0	1333.33	0
MW Min Utilities	877.084	66.08	761.17	12.38
ST Max Revenue	920.28	70.21	814.14	10.27
ST Max Matches	918.48	68.16	811.74	10.38
ST Min Utilities	905.45	68.92	803.31	11.80
ST' Max Revenue	923.07	68.30	815.90	9.39
ST' Max Matches	923.21	68.64	818.52	10.19
ST' Min Utilities	905.45	68.92	803.31	11.80

Table 8: Number of people in the matched group in different cases

Case	$N = 1$	$N = 2$	$N = 3$	$N > 3$	**Total groups**
MW Max Revenue	200	0	0	0	200
MW Max Matches	200	0	0	0	200
MW Min Utilities	35	31	21	9	200
ST Max Revenue	39	40	19	6	104
ST Max Matches	39	40	19	6	104
ST Min Utilities	37	37	23	5	102
ST' Max Revenue	2(+39)	38	21	5	66(+39)
ST' Max Matches	2(+39)	38	21	5	66(+39)
ST' Min Utilities	0(+37)	37	23	5	65(+37)

Table 9: Numerical results for maximizing revenue with different fleet sizes

Maximize Revenue	**Fleet size**	**Revenue**	**Empty vehicle travel time (h)**	**Personal disutility**	**Shifted time (h)**
MW	20	258.69	76.04	289.42	20.29
MW	40	493.47	130.64	524.95	31.26
MW	60	704.82	170.11	704.77	29.20
MW	80	907.28	197.19	882.38	32.31
MW	Unlimited	1767.28	0	1333.33	0
ST	20	240.68	47.01	197.37	0.71
ST	40	444.10	84.02	395.26	6.44
ST	60	619.32	97.70	555.17	9.53
ST	80	783.01	100.51	694.96	10.56
ST	Unlimited	920.28	70.21	814.14	10.27

in the matched group are also different as shown in Table 8. We have 2 more single people in ST' Max Revenue in addition to the 39 single people removed from ST Max Revenue.

In a stable match Ω, people are assigned to various groups of different sizes. For people in a group of multiple members, they have lower travel costs comparing to traveling alone. For people in a group of single member, they have the same travel costs comparing to traveling alone. According to Proposition 4, the sub-match Ω_1 containing only multi-person groups is also a stable match. It seems to be reasonable to follow Ω_1 as the optimal match even if some people may decide to leave the co-leasing program when they are assigned to a single-person group. However, we have shown here Ω_1 may not be the optimal stable match if there are changes in the participants of the co-leasing program according to Proposition 5 as well as the numerical results shown in Table 7. Suppose some people in single-person groups decide to quit the program, the stable match needs to be recalculated to guarantee the optimality of the match. Thus, we recommend keeping the single-person groups on the waiting list, and better group members for them may appear later when new users participate in the co-leasing program.

In Table 8, the total number of groups in optimal matchings can be larger than 100 for 200 people. Thus, we add scenario analysis with additional vehicle fleet size constraints comparing maximum weight matching and stable matching results. For purposes of minimizing user disutility, not all people are included in the final match with limited fleet size available, and it will lead to the fact that mostly one person is assigned to each vehicle to avoid higher user disutility in the whole group. Also, the results from maximizing the number of matches will be trivial since it will be the same as the number of vehicles available in the current scenario. We only include matching results with maximizing company revenue. We assume that the number of vehicles available to the 200 people group for sharing is 20, 40, 60, 80 and unlimited. Note that we have shown the results of unlimited vehicle fleet size in the upper Tables 7 and 8.

In Table 9, we show the vehicle leasing company's revenue as well as users' disutility for different

Table 10: Number of people in the matching group for maximizing revenue with different fleet sizes

Max Revenue	**Fleet size**	$N = 1$	$N = 2$	$N = 3$	$N > 3$
MW	20	2	3	5	10
MW	40	5	9	12	14
MW	60	9	21	21	9
MW	80	15	39	22	4
MW	Unlimited	200	0	0	0
ST	20	4	7	2	7
ST	40	7	15	9	9
ST	60	12	27	13	8
ST	80	22	33	20	5
ST	Unlimited	39	40	19	6

fleet sizes, and the number of groups with different sizes are provided in Table 10. The conclusion is consistent with our previous observations that the vehicle leasing company tends to increase its revenue while sacrificing users' travel experience since both revenue and disutilities are high. In terms of the number of people in each group, we have observed the extreme case that all users will be assigned to a vehicle when the fleet size is not limited. When vehicle fleet size is limited, we can see the company is trying to assign more people into a group to share one vehicle so that more people are included in the vehicle sharing program to improve revenue. However, the company may assign people into smaller groups to improve their revenue as we can see the number of groups with $N > 3$ is decreasing as the fleet size increases in Table 10. Eventually, it will lead to the extreme case that all people are assigned to a vehicle with unlimited fleet size, which totally unfair for the users. In stable matching results, the empty vehicle travel time, as well as shifted time, is much lower than that in maximum weight matching results, indicating that users are more fairly matched regardless of the company's revenue. At the same time, the distribution of the number of groups with different sizes is quite consistent as fleet size increases, indicating that the stable matching results always ensure the fairness among users in different conditions.

4.2 Algorithm analysis

In this section, we randomly chose a sample of 200 people to analyze the computational performance as well as solution quality differences between the optimal enumeration algorithm and our heuristics algorithm. We ran 12-people and 16-people sample cases for 50 times to obtain a more reliable conclusion on our algorithm performance. We assume that only a maximum of 3 people is allowed in user groups for computational considerations in these small sample cases. Later, the results of a larger toy problem containing 50 people are also shown.

4.2.1 Algorithm analysis based on small samples

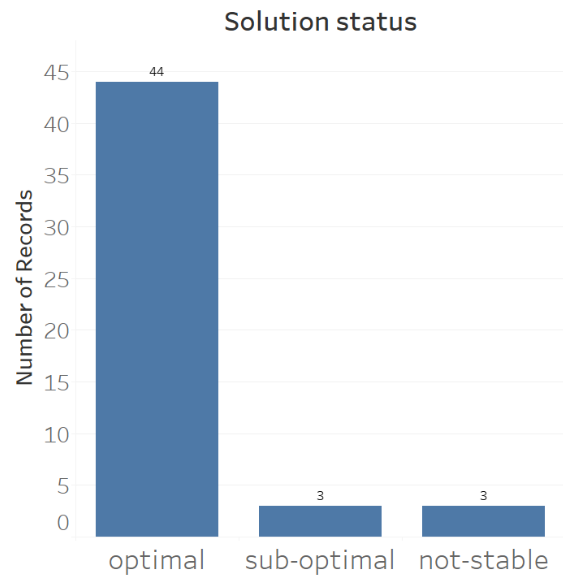
Here, we compare the pre-processing time for generating user groups in both algorithms, and we show the differences in the total user disutility in the final matching results. Note that we use minimizing user disutility in our stable matching problem in all cases. In terms of computational performance, the pre-processing time used to generate user disutility for all groups based on AVSP is much longer for both algorithms, and the results are shown in the following Figure 4. We can see the average enumeration solving time for 12-people cases is around 160.3 seconds where a total number of 286 AVSP problems including all groups with no more than 3 people are solved in Figure 4b. Similarly, the average enumeration solving time for 16-people cases is around 1125 seconds where a total number of 680 AVSP problems are solved in Figure 4d. In both small sample cases, we can see that our heuristic is much for efficient than the enumeration algorithm.

In Figure 4a and 4c, we compare the total user disutility of the enumeration algorithm and our heuristic results. There are three different cases for heuristics matching results, including 'optimal', 'sub-optimal' and 'not-stable'. In *optimal* cases, we have the same matching results for both algorithms. In *sub-optimal* cases, the heuristic algorithm eliminates some groups that may be in the optimal solution so that the solution is sub-optimal. In *not-stable* cases, the heuristic algorithm eliminates some groups that may be blocking groups for the optimal solution so that the solution is infeasible in the actual stable matching problem with better objective values. The average gap between heuristic disutility and optimal disutility is around 3.4% for 12-people sample cases and 6.5% for 16-people sample cases. According to the results, our algorithm seems to be robust given most sample cases can be solved to optimal with our heuristic algorithm.

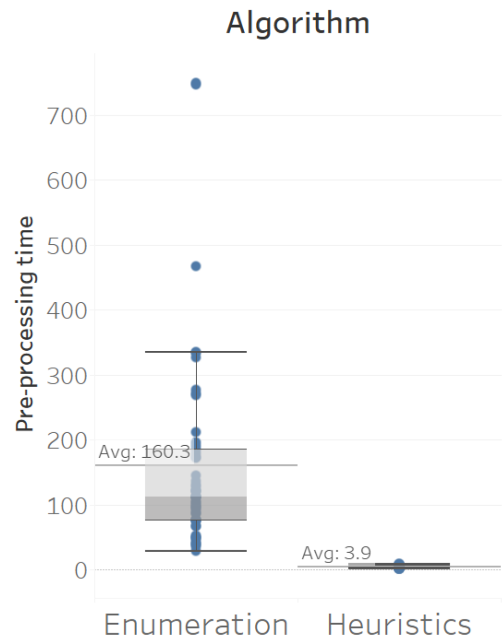
We have also visualized a 12-people sample case in Figure 5 showing the trip routes of different users to obtain insights on why the heuristic result has around 9% gap (which is the largest gap among all cases) comparing to optimal matching result. We use different colors to denote different users, and the person no is labeled along with their trip routes. The optimal stable matching result from enumeration algorithm contains 1-person groups of $\{\{1\}, \{8\}, \{9\}, \{10\}, \{11\}\}$, 2-person groups of $\{\{4, 12\}, \{6, 7\}\}$ and a 3-person group of $\{\{2, 3, 5\}\}$ while our heuristic algorithm suggests to separate person 2 from person 3 and person 5 with all other groups stay the same. The problem here is that the group of person 2 and person 5 is a bad match, and person 5 would prefer to stay along. However, since we can see the trips of person 2 are on the way to connect person 3 and 5, which can help reduce the cost of all people in the group of person 2, 3 and 5. Thus, we observe the special case discussed in the earlier sections of algorithm limitation. We can also observe a similar pattern for person 9, 10 and 11, while they all prefer to stay along due to their longer trips and less flexibility in travel schedules. However, it is possible that they may form a good match if ride sharing is considered, and we may explore that direction in our future works.

4.2.2 Performance of larger sample cases

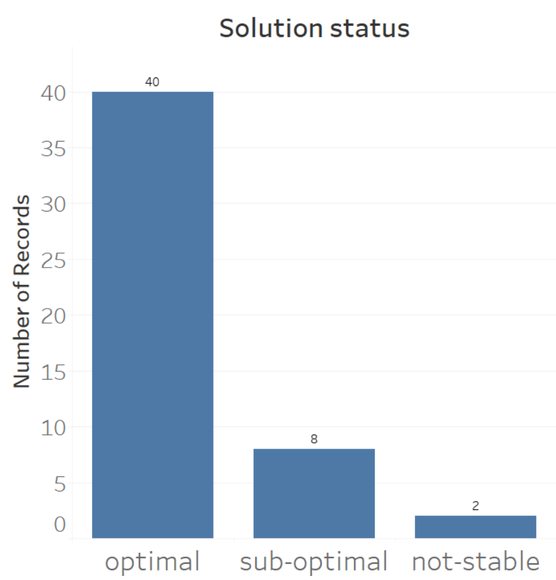
For our enumeration algorithm, it took more than 10 hours to solve the stable matching problem with 50 participates, and the problem is unsolvable if we increase the number of users to 200. On



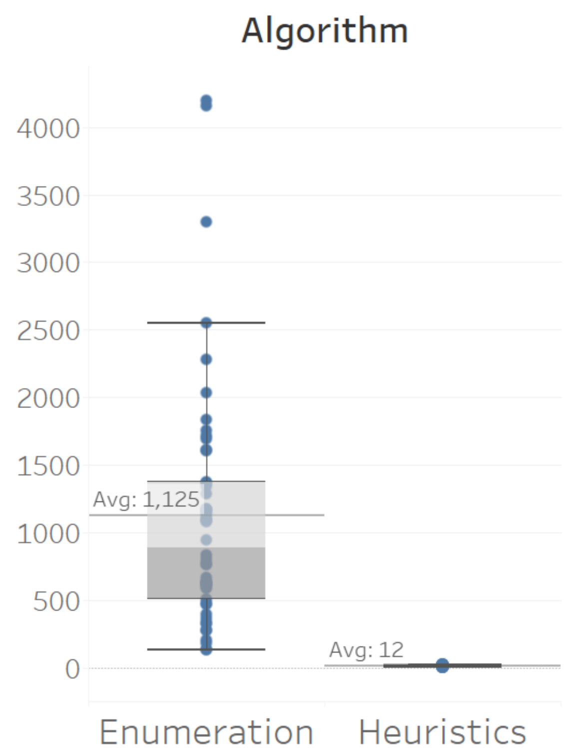
(a) Solution status summary for 12-people case



(b) Pre-processing time for 12-people case



(c) Solution status summary for 16-people case



(d) Pre-processing time for 16-people case

Figure 4: Algorithm analysis results based on small sample problems

the other hand, our valid group algorithm is much faster with acceptable suboptimal solutions. The computational time for user matching problem of different sizes based on valid group algorithm is

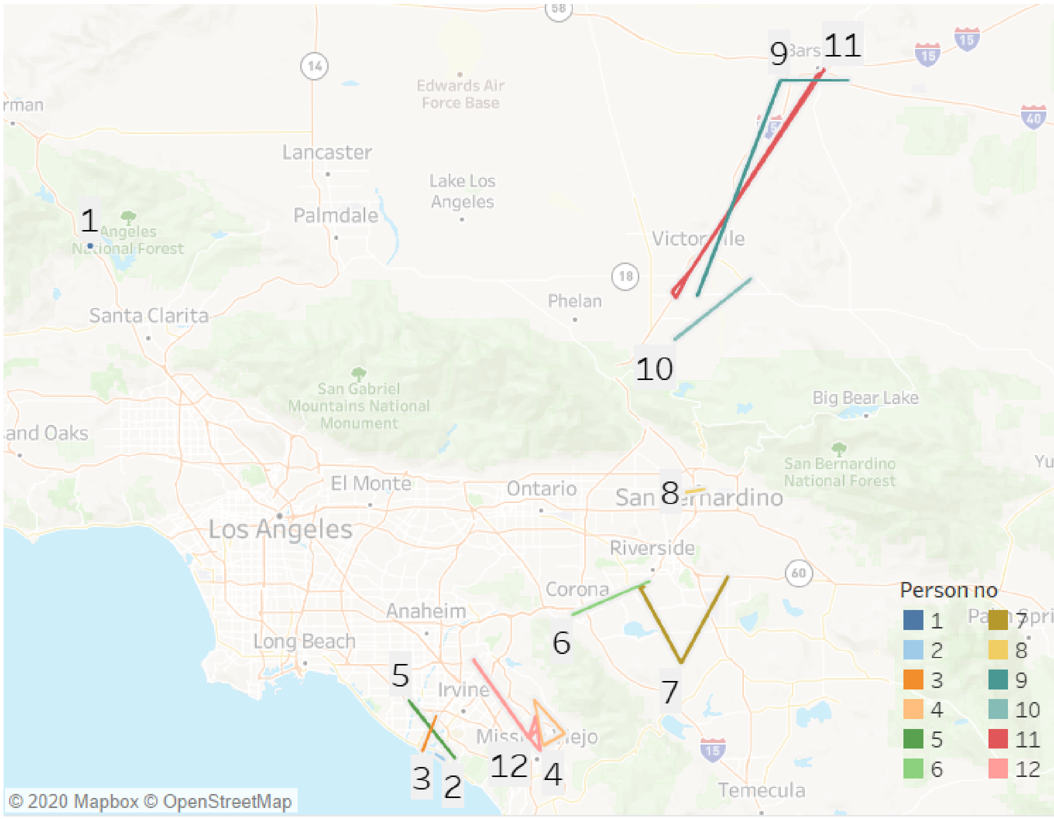


Figure 5: Trip route visualization for a sample group of 12 people in California

shown in Table 11. We use T_{sum} to denote the total solving time for the whole user stable matching problem, and we use $N_{\text{valid groups}}$ to denote the total number of valid groups obtained in the solving process. For each valid group, we need to solve AVSP for once, and we use $\tau_{\max}$ to denote the maximum solving time of AVSP and τ_{avg} to denote the average solving time of AVSP in the whole solving process. Also, we use T_i to denote the total time spent on solving AVSP for all valid groups of size i. Apparently, the maximum size of a valid group in a 200-person stable matching problem is 6, which is far less than our preset value of 200. We can also observe that our algorithm can accelerate the process of identifying multi-member potential groups since we can eliminate most of them based on black list rules.

As we have discussed earlier, we eliminate some of the potential groups that could be part of

Table 11: Valid group algorithm computational time (seconds)

Problem size	T_{sum}	$N_{\text{valid groups}}$	$\tau_{\max}$	τ_{avg}	T_2	T_3	T_4	T_5	T_6
$M = 20$	200.24	206	5.85	0.36	43.12	251.45	0.09	—	—
$M = 40$	1691.85	918	4.48	0.29	123.54	1561.94	1.01	9.7E-5	—
$M = 50$	2127.73	1199	4.49	0.27	185.30	1936.00	1.04	2.4E-4	—
$M = 100$	6642.87	4304	8.14	0.21	744.80	5855.13	38.10	0.062	—
$M = 200$	25469.44	16631	18.69	0.17	3437.35	21738.64	279.50	9.160	0.004

Table 12: Confusion matrix for $N = 50$

		Algorithm-valid test: Positive	Negative	**Total**
Stable test	**Positive**	711	74	785
	Negative	1456	17359	18815
	Total	2167	17433	19600

the optimal stable match. Here, we show the confusion matrix for stable groups and valid groups to see how often eliminated stable groups could appear in our final matching result. The solving results with both algorithms are shown in Table 12. The observations are consistent comparing to the earlier small sample cases. Although some stable groups are eliminated in our algorithm, the number is quite small comparing to the huge number of unstable groups we identified with our algorithm. We can conclude that the special cases discussed earlier are quite rare and have little effect on the final matching result.

Based on the analysis of small sample results, we believe that it is reasonable to accept the trade-off between computational time and solution quality based on our valid group algorithm even in larger problems due to the rareness of the special cases.

5 Summary and future works

In this paper, we develop a user matching model for autonomous co-leasing problems based on a generalized stable matching formulation as well as a single-vehicle scheduling problem (AVSP). We use AVSP to obtain personal disutility based on empty vehicle travel costs and time penalty cost of trip rescheduling. Several objective functions are introduced to derive insights on the relationship between suppliers and users while facing the new car-sharing market. An algorithm based on the concept of valid group and black list is developed to improve the enumeration process while solving the problem. Some rare cases are abandoned to improve the overall computational time of the whole matching problem. Based on our experiments in case study, we believe our proposed algorithm performs well in both solution quality and computational time. According to our findings, we believe that it is reasonable to ask suppliers to consider stability in the matching model so that the overall fairness for all users can be ensured. Our major contribution in this work is that we proposed a generalized formulation to analyze the potential new car-sharing market. The flexibility in our model (alternative objectives and removable constraints) allows us to derive insights on the behaviors of service providers and users. Stable matching will ensure that most people are satisfied, potentially attracting more people to participate in the co-leasing program. Also, the size of the group could be limited to improve computational efficiency since very few people choose to form larger groups.

There is much to improve for our model in the future. First, vehicle-specific considerations can be incorporated such as vehicle types, car seats and etc. This requires extensions of the multi-dimensional stable matching formulation. Second, a more sophisticated trip scheduling models can be used instead of the AVSP to derive preference metrics. In our current model, two travelers with similar routes and departure times would not be grouped into one group; instead, they would either be assigned to two different groups or left aside if no one else can help them reduce traveling costs. Given seating capacity information, we need to allow ride sharing among members in the same group to provide more reasonable utility estimations for user groups to be a stable match. In addition, trade-off between cost of sharing the vehicle and flexible trip schedule should also be considered in personal disutility, and additional assumptions to ensure the fairness in trip scheduling in a group. Or other types of mobility service options, which will be made more available to travelers with AVs, can complement the trip requirements. Since the trips of users are not always identical across days, we need to consider the variability of trips in multiple days when data is available. Our current AVSP only takes a set of deterministic trips as inputs, thus the estimation of disutility can be biased on specific days while the actual disutility in a general long-term car co-ownership group may be different. Consideration of traveler's multi-day travel pattern may result in preference distributions instead of single preference values, which may lead to extending the proposed multi-dimensional user matching problem to an uncertain or a robust problem. Lastly, rigorous efforts in understanding the multidimensional user matching problem and development of optimal solution methodologies will enable further applications.

6 Acknowledgment

This research, in part, is supported by the National Science Foundation. Authors are grateful for generous support. We would also like to thank Center for Computational Research, University at Buffalo for their support in computational resources.

References

Abeledo, H., Y. Blum. 1996. Stable matchings and linear programming. *Linear Algebra and Its Applications* **245** 321–333.

Abeledo, H. G., U. G. Rothblum. 1994. Stable matchings and linear inequalities. *Discrete Applied Mathematics* **54**(1) 1–27.

Abraham, D. J., P. Biró, D. F. Manlove. 2005. "almost stable" matchings in the roommates problem. *International Workshop on Approximation and Online Algorithms*. Springer, 1–14.

Agatz, N., A. Erera, M. Savelsbergh, X. Wang. 2012. Optimization for dynamic ride-sharing: A review. *European Journal of Operational Research* **223**(2) 295–303.

Allahviranloo, M., J. Y. Chow. 2019. A fractionally owned autonomous vehicle fleet sizing problem with time slot demand substitution effects. *Transportation Research Part C: Emerging Technologies* **98** 37–53.

Alternative Fuels Data Center, A. 2017. Charging plug-in electric vehicles at home. `http://www.afdc.energy.gov/fuels/electricity_charging_home.html`. Accessed: 2017-06-02.

Asghari, M., D. Deng, C. Shahabi, U. Demiryurek, Y. Li. 2016. Price-aware real-time ride-sharing at scale: an auction-based approach. *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 3.

Bernstein, A. 2017. Best green car deals: May 2017. `http://www.carsdirect.com/deals-articles/best-green-car-deals`. Accessed: 2017-06-02.

Boros, E., V. Gurvich, S. Jaslar, D. Krasner. 2004. Stable matchings in three-sided systems with cyclic preferences. *Discrete Mathematics* **289**(1) 1–10.

Casas, J. 2002. 2000-2001 california statewide household travel survey final report. *California Department of Transportation* URL `https://trid.trb.org/view.aspx?id=681358`.

Cordeau, J.-F., G. Laporte, et al. 2007. The dial-a-ride problem: models and algorithms. *Annals of Operations Research* **153**(1) 29–46.

Egan, M., M. Jakob. 2014. A profit-aware negotiation mechanism for on-demand transport services. *Proceedings of the Twenty-first European Conference on Artificial Intelligence*. IOS Press, 273–278.

Eriksson, K., J. Sjöstrand, P. Strimling. 2006. Three-dimensional stable matching with cyclic preferences. *Mathematical Social Sciences* **52**(1) 77–87.

Fagnant, D. J., K. M. Kockelman. 2014. The travel and environmental implications of shared autonomous vehicles, using agent-based model scenarios. *Transportation Research Part C: Emerging Technologies* **40** 1–13.

Faye, A., D. Watel. 2016. Static dial-a-ride problem with money as an incentive: Study of the cost constraint URL `https://hal.archives-ouvertes.fr/hal-01507417`.

Gale, D., L. S. Shapley. 1962. College admissions and the stability of marriage. *The American Mathematical Monthly* **69**(1) 9–15.

Haboucha, C. J., R. Ishaq, Y. Shiftan. 2017. User preferences regarding autonomous vehicles. *Transportation Research Part C: Emerging Technologies* **78** 37–49.

Huang, C.-C. 2007. Two's company, three'sa crowd: Stable family and threesome roommates problems. *European Symposium on Algorithms*. Springer, 558–569.

Irving, R. W. 1985. An efficient algorithm for the "stable roommates" problem. *Journal of Algorithms* **6**(4) 577–595.

Irving, R. W. 1994. Stable marriage and indifference. *Discrete Applied Mathematics* **48**(3) 261–272.

Iwama, K., S. Miyazaki. 2008. A survey of the stable marriage problem and its variants. *Informatics Education and Research for Knowledge-Circulating Society, 2008. ICKS 2008. International Conference on*. IEEE, 131–136.

Iwama, K., S. Miyazaki, K. Okamoto. 2007. Stable roommates problem with triple rooms. *Proc. 10th KOREA-JAPAN Joint Workshop on Algorithms and Computation (WAAC 2007)*. 105–112.

Khayati, Y., J. E. Kang, M. H. Karwan, C. C. Murray. 2019a. Household activity pattern problem with autonomous vehicles. *Under Review* .

Khayati, Y., J. E. Kang, M. H. Karwan, C. C. Murray. 2019b. Household use of autonomous vehicles with ride sourcing. *Under Review* .

Knuth, D. E. 1997. *Stable marriage and its relation to other combinatorial problems: An introduction to the mathematical analysis of algorithms*, vol. 10. American Mathematical Soc.

Lavieri, P. S., V. M. Garikapati, C. R. Bhat, R. M. Pendyala, S. Astroza, F. F. Dias. 2017. Modeling individual preferences for ownership and sharing of autonomous vehicle technologies. *Transportation Research Record: Journal of the Transportation Research Board* (2665) 1–10.

Litman, T. 2003. Transportation cost and benefit analysis: Techniques, estimates and implications URL `https://trid.trb.org/view.aspx?id=1158240`.

Litman, T. 2014. The future isn't what it used to be: Changing trends and their implications for transport planning. Tech. rep., Victoria Transport Policy Institute.

Masoud, N., R. Jayakrishnan. 2016. Formulations for optimal shared ownership and use of autonomous or driverless vehicles. *Transportation Research Board 95th Annual Meeting*. 16-4473.

Nair, G. S., S. Astroza, C. R. Bhat, S. Khoeini, R. M. Pendyala. 2017. An application of a rank ordered probit modeling approach to understanding level of interest in autonomous vehicles .

Ng, C., D. S. Hirschberg. 1991. Three-dimensional stabl matching problems. *SIAM Journal on Discrete Mathematics* **4**(2) 245–252.

Nissan. 2018. Tout savoir sur nissan intelligent get & go micra. `https://www.nissan.fr/vehicules/neufs/micra/get-and-go.html`. Accessed: 2018-02-25.

Ronn, E. 1986. On the complexity of stable matchings with and without ties AAI8728391.

Ronn, E. 1990. Np-complete stable matching problems. *Journal of Algorithms* **11**(2) 285–304.

Roth, A. E. 1984. The evolution of the labor market for medical interns and residents: a case study in game theory. *Journal of political Economy* **92**(6) 991–1016.

Roth, A. E. 1991. A natural experiment in the organization of entry-level labor markets: regional markets for new physicians and surgeons in the united kingdom. *The American economic review* 415–440.

Roth, A. E. 2008. Deferred acceptance algorithms: History, theory, practice, and open questions. *international Journal of game Theory* **36**(3) 537–569.

Roth, A. E., U. G. Rothblum, J. H. Vande Vate. 1993. Stable matchings, optimal assignments, and linear programming. *Mathematics of operations research* **18**(4) 803–828.

Santos, D. O., E. C. Xavier. 2015. Taxi and ride sharing: A dynamic dial-a-ride problem with money as an incentive. *Expert Systems with Applications* **42**(19) 6728–6737.

Sexton, T. R., Y.-M. Choi. 1986. Pickup and delivery of partial loads with "soft" time windows. *American Journal of Mathematical and Management Sciences* **6**(3-4) 369–398.

Shaheen, S., A. Cohen, M. Jaffee. 2018. Innovative mobility: Carsharing outlook .

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

Sivak, M. 2013. Has mass motorization in the us peaked? the university of michigan transportation research institute. Tech. rep., UMTRI-2013-17, June.

Small, K. A. 1982. The scheduling of consumer activities: work trips. *The American Economic Review* **72**(3) 467–479.

Subramanian, A. 1994. A new approach to stable matching problems. *SIAM Journal on Computing* **23**(4) 671–700.

Teo, C.-P., J. Sethuraman. 1998. The geometry of fractional stable matchings and its applications. *Mathematics of Operations Research* **23**(4) 874–891.

Teo, C.-P., J. Sethuraman. 2000. On a cutting plane heuristic for the stable roommates problem and its applications. *European Journal of Operational Research* **123**(1) 195–205.

Trading Economics. 2017. United states average hourly wages 1964-2017. `http://www.tradingeconomics.com/united-states/wages`. Accessed: 2017-06-02.

Vate, J. H. V. 1989. Linear programming brings marital bliss. *Operations Research Letters* **8**(3) 147–153.

Vohra, R. V. 2012. Stable matchings and linear programming. *Current Science(Bangalore)* **103**(9) 1051–1055.

Wang, X., N. Agatz, A. Erera. 2017. Stable matching for dynamic ride-sharing systems. *Transportation Science* .

Winter, K., O. Cats, K. Martens, B. van Arem. 2017. A stated-choice experiment on mode choice in an era of free-floating carsharing and shared autonomous vehicles. Tech. rep.

Appendix

A Pricing rules for the disutility of two-person match

In the simple case of matching two persons with one vehicle, we consider the major costs include fixed vehicle leasing costs, time penalty costs and vehicle usage costs for each person.

The fixed vehicle leasing costs should be divided depending on personal usage of the vehicle, which is the in-vehicle travel distance of each person. This is reasonable since people pay more if they request more usage time than other users. Suppose the in-vehicle travel distance for person m and person n are σ_m and σ_n, the vehicle leasing cost becomes $\frac{\sigma_m}{\sigma_m+\sigma_n}C_{\text{lease}}$ for person m and $\frac{\sigma_n}{\sigma_m+\sigma_n}C_{\text{lease}}$ for person n. Given the trip schedule, we can have the total empty vehicle trip distance as D_{empty}.

For time penalty costs in a two-person-match, we assume that the trip schedule will be decided under a fair negotiation that both people will accept. Considering the similar case of single driver sharing rides with multiple riders, Egan and Jakob (2014) presented a four-stage pricing negotiation mechanism with profit-maximum consideration. In our model, we assume that the trip schedule should be fair to both person considering the time penalty. The corresponding empty vehicle trip will not affect the result of the negotiation.

For total vehicle usage costs including empty vehicle trips and personal trips, we define three different pricing schemes to split the cost, including *equal splitting*, *balance splitting* and *detour based balance splitting*. Equal splitting and balance splitting are based on the similar rules that pricing split on ride-sharing based on variable travel costs are roughly proportional to vehicle-miles, presented in ride sharing systems according to previous literature (Agatz et al., 2012; Santos and Xavier, 2015; Faye and Watel, 2016). The total vehicle usage cost will be equal splitting for each user while the total vehicle usage cost will be split depending on in-vehicle travel distance in balance splitting. However, it could be unfair to divide the empty vehicle trip cost by in-vehicle travel distance since the empty vehicle trip may be due to a person's activity that locates in a remote area. In that case, the person should be responsible for the empty vehicle trip instead of spiting among two persons. This is similar to the case of detour in ride sharing and Asghari et al. (2016) assumed that riders will receive discounts proportional to his detour and drivers' compensation should increase depending on the detour in order to provide a fair pricing model. Thus, we define a detour based splitting for the empty vehicle trip that the empty vehicle travel distance as well as the group member's travel distance are considered as the detour.

The formulations of the three pricing schemes to split the total vehicle usage cost between a matching pair (m, n) are as follows:

Equal splitting $r_1\frac{\sigma_m+\sigma_n+D_{\text{empty}}}{2}$ for each person

Balance splitting $r_1(\frac{\sigma_m}{\sigma_m+\sigma_n}D_{\text{empty}} + \sigma_m)$ for person m; $r_1(\frac{\sigma_n}{\sigma_m+\sigma_n}D_{\text{empty}} + \sigma_n)$ for person n

Detour based balance splitting $r_1(\sigma_m+\nu_m(D_{\text{empty}}+\sigma_n)D_{\text{empty}})$ for person m; $r_1(\sigma_n+\nu_n(D_{\text{empty}}+$

$\sigma_m)D_{\text{empty}})$ for person n

Here, we use $\nu_m(\cdot)$ to denote the profile function of person m and r_1 to denote vehicle cost per mile. We assume that $\nu_m(\cdot)$ is a decreasing function so that a person will pay less with higher detour. Obviously we have $\nu_m(D_{\text{empty}} + \sigma_n) + \nu_n(D_{\text{empty}} + \sigma_m) = 1$ in a match (m, n) with empty vehicle travel distance D_{empty}. The detailed assumptions of original profile function can be referred to Asghari et al. (2016). Similar ideas can be extended to multiple-person group matching.