Cost Allocation for Prioritized Ride-Sharing

Chaya Levinger, Noam Hazon, and Amos Azaria

Department of Computer Science, Ariel University, Israel
chaya.levinger@msmail.ariel.ac.il
{noamh,amos.azaria}@ariel.ac.il

Abstract. Ride-sharing services are gaining popularity and are crucial for a sustainable environment. In the last mile variant it is assumed that all the passengers are positioned at the same origin location. Indeed, the priority order in which the passengers are dropped-off plays an important role in passenger preferences.
In this paper we analyze fair cost allocation with a given priority order and propose a mechanism for determining this priority order. Namely, we show a method for efficient computation of the Shapley value, when there exist a priority order on the passengers. Our results are also applicable for efficient computation of the Shapley value in routing games. In addition, we present a VCG based mechanism for prioritization that obtains the value of time from each of the passengers and outputs a priority order. The mechanism is both efficient and truthful. We provide simulations showing that the overhead cost paid to the mechanism is reasonable, and is significantly lower than the commission charged by companies providing ride-sourcing services.

Keywords: Cost Allocation · Shapley Value · Mechanism Design · Ride-Sharing.

1 Introduction

On-demand ride-sharing services, which group together passengers with similar itineraries, can be of significant social and environmental benefit, by reducing travel costs, road congestion and CO_2 emissions. Indeed, the National Household Travel Survey performed in the U.S. in 2009 [22] revealed that approximately 83.4% of all trips in the U.S. were in a private vehicle (other options being public transportation, walking, etc.). The average vehicle occupancy was only 1.67 when compensating for the number of passengers (i.e., when several passengers travel together, the travel distance counts for every passenger in the vehicle). The deployment of autonomous cars in the near future is likely to increase the spread for ride-sharing services, since it will be easier and cheaper for a company to handle a fleet of autonomous cars that can serve the demands of different passengers.

Most works in the domain of ride-sharing are dedicated to the assignment of passengers to vehicles, or to planning optimal drop-off routes [21, 1, 15]. In this paper we introduce the concept of passenger prioritization to the domain of ride-sharing in the last mile variant [4]. That is, we analyze the cost allocation when all passengers are positioned at the same origin location and there is a priority order in which the passengers are dropped-off. Such prioritization may be attributed to the order in which the passengers arrived at the origin location, or the frequency of passenger usage of the service; the latter is similar to the different boarding groups on an aircraft. Other rationales for prioritization may include urgency of arrival or priority groups in need (e.g. elderly, disabled, pregnant women, and the injured). Indeed, in some scenarios there is no predetermined prioritization order. Moreover, different passengers may have different preferences regarding the possible prioritization. For example, young people may not mind being dropped-off last if they save a few dollars, but people in their 40's may be more concerned about their time and thus would like to pay more to be dropped-off first. Therefore, we also present a mechanism for determining the passenger prioritization that is based on the passengers' preferences.

The problem of cost allocation when there exists a predetermined priority order is closely related to the fixed route traveling salesman game [20], also known as routing game [26]. In this game a service provider makes a round-trip along the locations of several sponsors in a fixed order, where the total cost of the trip should be distributed among the sponsors. The fair cost allocation was achieved by finding an element of the core. We concentrate on the Shapley value [23] as our notion of fair cost allocation. The Shapley value is widely used in cooperative games, and is the only cost allocation satisfying efficiency, symmetry, null player property and additivity. However, as stated by [18], "In general, explicitly calculating the Shapley value requires exponential time. Hence, it is an impractical cost-allocation method unless an implicit technique given the particular structure of the game can be found". Furthermore, [26] has conjectured that there is no efficient way for computing the Shapley value in routing games. We show how to efficiently compute the Shapley value in a ride-sharing domain with a predetermined priority order, as well as in routing games.

In addition, we present a passenger prioritization mechanism based on the Vickrey–Clarke–Groves (VCG) mechanism [25, 5, 10]. Our mechanism allows every passenger, who was assigned to a specific vehicle, to provide her "value of time", and the mechanism then picks the priority order which maximizes the social value. The mechanism is truthful, that is, it is a dominant strategy for the passengers to provide their true "value of time". We run simulations to estimate the average commission charged by our mechanism, which resulted in only 8.5% of the ride cost. This value is significantly lower than the commission charged by ride-sourcing services such as Uber and Lyft which was reported to be at least 25% [19]. Clearly, our mechanism may charge an additional fee on the ride cost, without losing its beneficial properties.

We note that the term ride-sharing is used in the literature with different meanings. We consider only the setting where the vehicle operator does not have any preferences or predefined destination. Instead, the vehicle's route is determined solely by the passengers' requests. In addition, the context of our work is that the assignment of the passengers to the vehicle has already been determined, either by a ride-sharing system or by the passengers themselves, and we only need to decide on the cost allocation and the passenger prioritization. Therefore, we may assume that the passengers' destinations are located in a relatively close proximity to each other, and thus any permutation on the destinations is likely to be a reasonable prioritization. Moreover, since we focus on the case where the assignment has already been determined we do not consider the ability of passengers to deviate from the given assignment and join a different vehicle, which is acceptable since either they want to travel together or no other alternative exists.

To summarize, the contributions of this paper are two-fold:

(i) We show an efficient method for computing the Shapley value of each user in a shared-ride when the priority order is predetermined. Our solution entails that the Shapley value can be computed in polynomial time in routing games as well, which is in contrast to a previous conjecture made.
(ii) We present a truthful and efficient VCG based mechanism for prioritization in which each passenger provides only her "value of time". We provide simulations that show that the overhead cost paid to the mechanism is significantly lower than the commission charged by companies providing ride-sourcing services.

2 Related Work

The problem of cost allocation when there exists a predetermined priority order is closely related to the fixed route traveling salesman game [8, 20, 2]. One variant of this game is the fixed-route traveling salesman problems with appointments. In this variant the service provider is assumed to travel back home (to the origin) when she skips a sponsor. This variant was introduced by [26], who also showed how to efficiently compute the Shapley value for this problem but stated that his technique does not carry over to routing games.

Our setting can also be interpreted as a generalization of the airport problem [14] to a two dimensional plain. In the airport problem we need to decide how to distribute the cost of an airport runway among different airlines who need runways of different lengths. In our case we distribute the cost among passengers who need rides of different lengths and destinations. It was shown that the Shapley value can be efficiently computed for the airport problem, however achieving efficient computation of the Shapley value in our setting requires a different technique.

The problem of fair cost allocation was also studied in the context of logistic operation. In this domain, shippers collaborate and bundle their shipment requests together to achieve better rates from a carrier [11]. The Shapley value was also investigated in this domain of collaborative transportation [9, 24]. In particular, [18] stated that "we do not know of an efficient technique for calculating the Shapley value for the shippers' collaboration game". Indeed, [7] developed the line rule, which inspired in Shapley value, but requires less computational effort and relates better with the core, but it is suitable for a specific inventory transportation problem. [17] describe an approximation of the Shapley Value when trying to simultaneously allocate both the transportation costs and the emissions among the customers. Overall, we note that the main requirements from a cost allocation in the context of logistic operation is stability, and an equal distribution of the profit, since the collaboration is assumed to be long-termed. The type of interaction is our setting is inherently different, as it is an ad-hoc short term collaboration.

In another domain, [3] introduce a fair payment scheme, which is based on the game theoretic concept of the kernel, for the social ride-sharing problem (where the set of commuters are connected through a social network).

In the second part of our paper we use an auction based mechanism for determining the priority order. [16] also investigate an auction mechanism when sharing submodular costs. In their setting each agent may either receive a service or not, and the marginal cost of serving additional agents decreases as the group of agents who are already served increases. [16] investigate an auction mechanism to decide which agents are served, and then how to share the cost among them. In their work there is no priority order and each agent has a fixed value for being served. In our work the passengers are assumed to submit their value for each priority order, and are assumed to have a different value depending on the priority order. The service provider, on the other hand, only selects the priority order and must serve all the passengers.

Several works have considered the use of auctions in the domain of car-pooling. [12] use VCG-based payments in the car-pooling domain, which are based on the assignments of passengers to vehicles and the chosen routes. We use the VCG-based payment in the domain of ride-sharing, and it depends only on the priority order of the passenger in a given vehicle. [13] provide a solution for the assignment problem in the domain of car-pooling, based on auctions. Passengers are bidding for increasing their ranking, and thus visibility to drivers, whereas drivers can select passengers according to their preferences.

3 Fair Payments with a Predetermined Priority Order

In this section we assume that the passengers are ordered according to some predetermined priority order, and attempt to fairly compute the payment for every passenger. Unlike other related work [20], we do not require that the priority order will be the optimal order that minimizes the total cost. We use the Shapley value and show an efficient method to compute it.

3.1 Notation

We are given a set $U = \{u_1, u_2, ..., u_n\}$ of passengers (users) that depart from the same origin, d_0. Without loss of generality, passenger u_1 has the highest priority and therefore will be dropped-off first, passenger u_2 will be dropped off second, etc. Each passenger u_i has a corresponding destination d_i. Let D be the set of destinations, $D = \{d_1, d_2, ..., d_n\}$. We also add a dummy destination, d_{n+1}, to simplify the notation. We denote by $\delta(d_i, d_j)$ the shortest travel distance between d_i and d_j. In addition, for every $i \in \{0, 1, ..., n\}$, $\delta(d_i, d_{n+1}) = 0$ and $\delta(d_i, d_i) = 0$.

Given a set $S \subseteq D$, let $\tilde{S}$ be the set S ordered in an ascending order, and let $S[i]$ be the destination that is in the i-th position in $\tilde{S}$. For ease of notation we use $S[0]$ to denote d_0 and $S[|S|+1]$ to denote d_{n+1}.

Given a set $S \subseteq D$, let $v(S)$ be the shortest travel distance of the path that starts at the origin d_0 and traverses all destinations $d_i \in S$ according to an ascending order. That is, $v(S) = \sum_{i=0}^{|S|-1} \delta(S[i], S[i+1])$.

Let R be a permutation on D and let P_i^R be the set of the previous destinations to d_i in permutation R.

3.2 Efficient Computation of the Shapley Value

We are interested in determining the payment for each passenger, u_i, according to the Shapely value, $\phi(u_i)$. The Shapley value has several equivalent formulas, we use the following formula to derive an efficient computation:

$$\phi(u_i) = \frac{1}{n!}\sum_R \Big(v(P_i^R \cup \{d_i\}) - v(P_i^R)\Big)$$

Given a permutation R and a passenger u_i, let $d_l \in P_i^R$ be a destination such that $l < i$ and $\forall d_j \in P_i^R, j \leq l$ or $i < j$. If no such destination exists, then d_l is defined as d_0. Similarly, let $d_r \in P_i^R$ be a destination such that $i < r$ and $\forall d_j \in P_i^R, j < i$ or $r \leq j$. If no such destination exists, then d_r is defined as d_{n+1}. We use ℓ (and r) to denote the position of d_l (and d_r) in the ordered $\tilde{P_i^R}$, respectively. If $d_l = d_0$ then $\ell = 0$, and if $d_r = d_{n+1}$ then r= $|P_i^R|$+1. We note that $P_i^R[\ell] = d_l$, $P_i^R[r] = d_r$ and $r = \ell + 1$.

For example, assume $D = \{d_1, d_2, d_3, d_4, d_5, d_6\}$ and $R = \{d_6, d_2, d_5, d_4, d_3, d_1\}$, we get $P_4^R = \{d_6, d_2, d_5\}$ and thus $\tilde{P_4^R} = \{d_2, d_5, d_6\}$, $d_l = d_2$ (i.e., $\ell = 1$), $d_r = d_5$ (i.e., $r = 2$), and $P_4^R[\ell] = d_2$.

Our first observation is that the computation of the Shapley value in our setting, $\phi(u_i)$, may be written as the sum over the distances between pairs of destinations. $\phi(u_i) = \frac{1}{n!}\sum_{p=0}^{n-1}\sum_{q=p+1}^{n}\alpha_{p,q}^i\delta(d_p, d_q)$, for some $\alpha_{p,q}^i \in \mathbb{Z}$.

Proof. We note that $\phi(u_i)\cdot n!$ is a sum over $v(S)$ for multiple $S \subseteq D$. By definition, $v(S) = \sum_{j=0}^{|S|-1}\delta(S[j], S[j+1])$, such that $S[j] = d_p$ and $S[j+1] = d_q$ where $p < q$.

We now show that we can rewrite the computation of the Shapely value in our setting as follows.

Lemma 1.

$$\phi(u_i) = \frac{1}{n!}\sum_R \Big(\delta(d_l, d_i) + \delta(d_i, d_r) - \delta(d_l, d_r)\Big)$$

Proof.

$$v(P_i^R) = \sum_{j=0}^{|P_i^R|-1}\delta(P_i^R[j], P_i^R[j+1]) =$$

$$\sum_{j=0}^{\ell-1}\delta(P_i^R[j], P_i^R[j+1]) + \delta(d_l, d_r) + \sum_{j=r}^{|P_i^R|-1}\delta(P_i^R[j], P_i^R[j+1])$$

In addition,

$$v(P_i^R \cup \{d_i\}) = \sum_{j=0}^{\ell-1}\delta(P_i^R[j], P_i^R[j+1]) +$$

$$\delta(d_l,d_i)+\delta(d_i,d_r)+\sum_{j=r}^{|P_i^R|-1}\delta(P_i^R[j],P_i^R[j+1]).$$

By definition,

$$\phi(u_i)=\frac{1}{n!}\sum_R\big[v(P_i^R\cup\{d_i\})-v(P_i^R)\big]=$$

$\frac{1}{n!}\sum\limits_R\Big(\sum\limits_{j=0}^{\ell-1}\delta(P_i^R[j],P_i^R[j+1])+\delta(d_l,d_i)+\delta(d_i,d_r)+\sum\limits_{j=r}^{|P_i^R|-1}\delta(P_i^R[j],P_i^R[j+$ $1])-\big(\sum\limits_{j=0}^{\ell-1}\delta(P_i^R[j],P_i^R[j+1])+\delta(d_l,d_r)+\sum\limits_{j=r}^{|P_i^R|-1}\delta(P_i^R[j],P_i^R[j+1])\big)\Big)=$
$\frac{1}{n!}\sum\limits_R\big(\delta(d_l,d_i)+\delta(d_i,d_r)-\delta(d_l,d_r)\big)$

Following Observation 3.2 and Lemma 1 we now show that we can rewrite the computation of the Shapely value as a sum over distances, that can be computed in polynomial time.

Theorem 1. *For each i, $\phi(u_i)=\sum\limits_{p=0}^{i}\sum\limits_{q=i}^{n}\beta_{p,q}^i\delta(d_p,d_q)$, where $q\neq p$, and $\beta_{p,q}^i\in\mathbb{Q}$ are computed in polynomial time.*

Proof. By definition, $l<i<r$. According to Lemma 1 $\phi(u_i)\cdot n!$ is a sum over $\delta(d_p,d_q)$, where $p\le i\le q$. There are several terms in this sum:

- $\beta_{0,i}^i$ multiplies $\delta(d_0,d_i)$. Now, $\delta(d_0,d_i)$ appears in $\phi(u_i)$ in every permutation R when $d_l=d_0$. That is, in all of the permutations where destination d_i appears before any other destination d_x such that $x<i$. We now count the number of such permutations. There are $\binom{n}{i}$ options to place the destinations $d_1,d_2,...,d_i$ among the n available destinations. For each such option there are $(i-1)!$ options to order the destinations $d_1,d_2,...,d_i$ such that d_i is the first destination among them. Finally, there are $(n-i)!$ options to order the destinations $d_{i+1},d_{i+2},...,d_n$. Therefore, $\delta(d_0,d_i)$ appears in $\binom{n}{i}\cdot(i-1)!\cdot(n-i)!=\frac{n!}{i}$ permutations, and by inserting $\frac{1}{n!}$ into the sum we get that $\beta_{0,i}^i=\frac{1}{i}$.
- For each $q>i$, $\beta_{0,q}^i$ multiplies $\delta(d_0,d_q)$. Now, $\delta(d_0,d_q)$ appears negatively in $\phi(u_i)$ in every permutation R when $d_l=d_0$ and $d_r=d_q$. That is, in all of the permutations where destination d_q appears before d_i (i.e., $d_q\in P_i^R$), but any other destination d_x such that $x<q$, appears after d_i. We now count the number of such permutations. There are $\binom{n}{q}$ options to place the destinations $d_1,d_2,...,d_i,...,d_q$ among the n available destinations. For each such option there are $(q-2)!$ options to order the destinations $d_1,d_2,...,d_i$ such that d_q is the first destination and d_i is the second destination among them. Finally, there are $(n-q)!$ options to order the destinations $d_{q+1},d_{q+2},...,d_n$. Therefore, $\delta(d_0,d_q)$ appears negatively in $\binom{n}{q}\cdot(q-2)!\cdot(n-q)!=\frac{n!}{q\cdot(q-1)}$ permutations, and by inserting $\frac{1}{n!}$ into the sum we get that $\beta_{0,q}^i=-\frac{1}{q\cdot(q-1)}$.

- For each $0 < p < i$, $\beta^i_{p,i}$ multiplies $\delta(d_p, d_i)$. Now, $\delta(d_p, d_i)$ appears in $\phi(u_i)$ in every permutation R when $d_l = d_p$. That is, in all of the permutations where destination d_p appears before d_i (i.e., $d_p \in P_i^R$), but any other destination d_x such that $p < x < i$, appears after d_i. We now count the number of such permutations. There are $\binom{n}{i-p+1}$ options to place the destinations $d_p, d_{p+1}, ..., d_i$ among the n available destinations. For each such option there are $(i-p+1-2)!$ options to order the destinations $d_p, d_{p+1}, ..., d_i$ such that d_p is the first destination and d_i is the second destination among them. Finally, there are $(n-(i-p+1))!$ options to order the destinations $d_1, d_2, ..., d_{p-1}, d_{i+1}, d_{i+2}, ..., d_n$. Therefore, $\delta(d_p, d_i)$ appears in $\binom{n}{i-p+1} \cdot (i-p-1)! \cdot (n-(i-p+1))! = \frac{n!}{(i-p)\cdot(i-p+1)}$ permutations, and by inserting $\frac{1}{n!}$ into the sum we get that $\beta^i_{p,i} = \frac{1}{(i-p)\cdot(i-p+1)}$.
- For each $q > i$, $\beta^i_{i,q}$ multiplies $\delta(d_i, d_q)$. Now, $\delta(d_i, d_q)$ appears in $\phi(u_i)$ in every permutation R when $d_r = d_q$. That is, in all of the permutations where destination d_q appears before d_i (i.e., $d_q \in P_i^R$), but any other destination d_x such that $i < x < q$, appears after d_i. We now count the number of such permutations. There are $\binom{n}{q-i+1}$ options to place the destinations $d_i, d_{i+1}, ..., d_q$ among the n available destinations. For each such option there are $(q-i+1-2)!$ options to order the destinations $d_i, d_{i+1}, ..., d_q$ such that d_q is the first destination and d_i is the second destination among them. Finally, there are $(n-(q-i+1))!$ options to order the destinations $d_1, d_2, ..., d_{i-1}, d_{q+1}, d_{q+2}, ..., d_n$. Therefore, $\delta(d_p, d_i)$ appears in $\binom{n}{q-i+1} \cdot (q-i-1)! \cdot (n-(q-i+1))! = \frac{n!}{(q-i)\cdot(q-i+1)}$ permutations, and by inserting $\frac{1}{n!}$ into the sum we get that $\beta^i_{i,q} = \frac{1}{(q-i)\cdot(q-i+1)}$.
- For each p, q such that $p < i < q$, $\beta^i_{p,q}$ multiplies $\delta(d_p, d_q)$. Now, $\delta(d_p, d_q)$ appears negatively in $\phi(u_i)$ in every permutation R when $d_l = d_p$ and $d_r = d_q$. That is, in all of the permutations where destinations d_p, d_q appear before d_i (i.e., $d_p, d_q \in P_i^R$), but any other destination d_x such that $p < x < q, x \neq i$, appears after d_i. We now count the number of such permutations. There are $\binom{n}{q-p+1}$ options to place the destinations $d_p, d_{p+1}, ..., d_i, ..., d_q$ among the n available destinations. For each such option there are $(q-p+1-3)!$ options to order the destinations $d_p, d_{p+1}, ..., d_i, ..., d_q$ such that d_p is the first destination, d_q is the second and d_i is the third destination among them. Similarly, there are $(q-p+1-3)!$ options to order these destinations such that d_q is the first destination, d_p is the second and d_i is the third. Finally, there are $(n-(q-p+1))!$ options to order the destinations $d_1, d_2, ..., d_{p-1}, d_{q+1}, d_{q+2}, ..., d_n$. Therefore, $\delta(d_p, d_q)$ appears in $\binom{n}{q-p+1} \cdot 2 \cdot (q-p-2)! \cdot (n-(q-p+1))! = \frac{2\cdot n!}{(q-p-1)\cdot(q-p)\cdot(q-p+1)}$ permutations, and by inserting $\frac{1}{n!}$ into the sum we get that $\beta^i_{p,q} = -\frac{2}{(q-p-1)\cdot(q-p)\cdot(q-p+1)}$.

We note that our setting is very similar to the setting of routing games [20]. The model of routing games is of one service provider that makes a round-trip along the locations of several sponsors in a fixed order, where the total cost of the trip should be distributed among the sponsors. Clearly, our model of ride-sharing with predetermined priority order is almost identical: the service provider corresponds to the vehicle and the sponsors correspond to the passengers. The only difference is that in a routing game the sponsors also pay the cost of the trip back to the origin. Indeed, the results presented in this section carry over to routing games.

Theorem 2. *The Shapley value in routing games can be computed in polynomial time.*

Proof (Proof (sketch)). We use our previous definitions and results with the following slight modifications. The dummy destination d_{n+1} becomes d_0. Thus, $\delta(d_i, d_{n+1}) = \delta(d_i, d_0)$. In Observation 3.2 we need to modify the bound in the outer sum (with the index p) to n and the bound in the inner sum (with the index q) to $n+1$. In addition, we use the proof of Theorem 1, but we add $\sum_{p=0}^{i} \beta^i_{p,n+1}\delta(d_p, d_{n+1})$ to the calculation of $\phi(u_i)$, where for $p < i$, $\beta^i_{p,n+1} = -\frac{1}{(n-p)\cdot(n-p+1)}$ and $\beta^i_{i,n+1} = \frac{1}{n-i+1}$.

Note that this is an unexpected result, since it refutes the conjecture of [26] that there is no efficient way for computing the Shapley value in routing games.

4 Mechanism Design for Prioritization

In Section 3 we assumed that the passengers are ordered according to some priority order. In this section we assume that there is no such predetermined priority order. Instead, the group of passenger should decide on the priority order. Clearly, each order has a different total and personal ride cost, and the prioritization also affects the travel time of each passenger (and possibly additional factors). Since each passenger may have a different value of time, we build a mechanism that elicits the preferences from the passengers and outputs a prioritization order as well as the mechanism fee for each passenger (in addition to the ride cost). We would like the mechanism to maximize the social welfare and to be strategy-proof. That is, the mechanism chooses an order that maximizes the sum of passengers' values, and truth-telling is a dominant strategy for each passenger.

We begin with some definitions. Let U be the set of passengers, and let R be a permutation of U. Passenger u_i has a true value v_i^R, for the shared-ride that drops-off all passengers according to the order in the permutation R. Given a set of passengers U and a drop-off order, R, a passenger u_i is associated with some ride cost c_i^R for this specific ride. This cost may depend on the entire set of destinations, as well as the order in which all passengers are dropped-off, and is known in advance to all passengers. Each passenger u_i reports her value for all possible drop-off orders; we use $\bar{v}_i^R$ to denote each of these reported values. The mechanism selects a priory order, $\hat{R}$, and determines the fee, $f_i^{\hat{R}}$, for passenger u_i. We note that, in addition to the fee paid by the passenger, the passenger must pay the ride cost, $c_i^{\hat{R}}$. That is, unlike the standard setting when VCG is applied, a user may have different values for the different options as well as different costs that are associated with each of the options. We use $g_i^{\hat{R}}$ to denote the utility (gain) of passenger u_i, $g_i^{\hat{R}} = v_i^{\hat{R}} - c_i^{\hat{R}} - f_i^{\hat{R}}$.

Before we present a truthful mechanism for this problem, we show that, in our case, the VCG mechanism that maximizes the reported values and ignores the predetermined ride cost is not strategy-proof. Recall that the priority order ultimately selected by the VCG mechanism is $\hat{R} = \operatorname*{argmax}_R \sum_i (\bar{v}_i^R)$, and the fee of VCG is $f_i^{\hat{R}} = \max_R \sum_{j \neq i} (\bar{v}_j^R) - \sum_{j \neq i} (\bar{v}_j^{\hat{R}})$. Consider the example described in Table 4, where $U = u_1, u_2$.

Passengers \ Drop-off Order	$u_1 \rightarrow u_2$	$u_2 \rightarrow u_1$
	value (cost)	value (cost)
u_1	6 (4)	3 (2)
u_2	2 (1)	4 (4)

Table 1. An example in which the VCG mechanism that ignores the predetermined ride cost does not result in a truthful mechanism.

If u_1 and u_2 bid truthfully (i.e. for all $R, \bar{v}_i^R = v_i^R$), the selected order is $u_1 \rightarrow u_2$ and the utilities are
$g_1^{u_1 \rightarrow u_2} = 6 - 4 - f_1^{u_1 \rightarrow u_2} = 6 - 4 - (4 - 2) = 0$,
$g_2^{u_1 \rightarrow u_2} = 2 - 1 - f_1^{u_1 \rightarrow u_2} = 2 - 1 - (6 - 6) = 1$. Now, suppose that u_1 does not bid truthfully and $\bar{v}_1^{u_2 \rightarrow u_1} = 5$. The selected order in this case is $u_2 \rightarrow u_1$ and the utility of u_1 is
$g_1^{u_2 \rightarrow u_1} = 3 - 2 - f_1^{u_2 \rightarrow u_1} = 3 - 2 - (4 - 4) = 1$,
That is, truth-telling is not a dominant strategy for u_1 and VCG is not strategy-proof in this case.

Our proposed VCG-based mechanism takes into account also the ride cost associated with every ride. It therefore selects the following drop-off order $\hat{R} = \underset{R}{\operatorname{argmax}} \sum_i (\bar{v}_i^R - c_i^R)$. In addition, given the selected order $\hat{R}$, we define $f_i^{\hat{R}} = \max_R \sum_{j \neq i} (\bar{v}_j^R - c_j^R) - \sum_{j \neq i} (\bar{v}_j^{\hat{R}} - c_j^{\hat{R}})$.

We now show that the proposed mechanism is truthful for every passenger u_i. Intuitively, given the selected order $\hat{R}$, $g_i^{\hat{R}} = v_i^{\hat{R}} - c_i^{\hat{R}} - f_i^{\hat{R}} = v_i^{\hat{R}} - c_i^{\hat{R}} - \Big(\max_R \sum_{j \neq i} (\bar{v}_j^R - c_j^R) - \sum_{j \neq i} (\bar{v}_j^{\hat{R}} - c_j^{\hat{R}}) \Big)$. Therefore, given the selected order, the utility of a passenger does not depend on her reported values.

Theorem 3. *For any passenger u_i, reporting the true values v_i^R is a dominant strategy.*

Proof. We need to show that for any reported values of the other passengers, the utility of u_i when she reports her true values, v_i^R, is greater than or equals her utility when she reports any other values. We fix $\bar{v}_j^R$ for all R and $j \neq i$. Let R^* be $\underset{R}{\operatorname{argmax}} \Big(v_i^R - c_i^R + \sum_{j \neq i} (\bar{v}_j^R - c_j^R) \Big)$. That is, the selected order when u_i bids truthfully. Let $\bar{R}$ be the selected order when u_i bids $\bar{v}_i^R$. We show that

$$g_i^{R^*} \geq g_i^{\bar{R}}$$

By definition,

$$g_i^{R^*} = v_i^{R^*} - c_i^{R^*} - \max_R \sum_{j \neq i} (\bar{v}_j^R - c_j^R) + \sum_{j \neq i} (\bar{v}_j^{R^*} - c_j^{R^*}).$$

In addition,

$$v_i^{R^*} - c_i^{R^*} + \sum_{j \neq i} (\bar{v}_j^{R^*} - c_j^{R^*}) = \max_R \Big(v_i^R - c_i^R + \sum_{j \neq i} (\bar{v}_j^R - c_j^R) \Big).$$

Therefore,

$$\begin{aligned} g_i^{R^*} &= \max_R \Big(v_i^R - c_i^R + \sum_{j \neq i} (\bar{v}_j^R - c_j^R) \Big) - \max_R \sum_{j \neq i} (\bar{v}_j^R - c_j^R) \\ &\geq v_i^{\bar{R}} - c_i^{\bar{R}} + \sum_{j \neq i} (\bar{v}_j^{\bar{R}} - c_j^{\bar{R}}) - \max_R \sum_{j \neq i} (\bar{v}_j^R - c_j^R) \\ &= v_i^{\bar{R}} - c_i^{\bar{R}} - (\max_R \sum_{j \neq i} (\bar{v}_j^R - c_j^R) - \sum_{j \neq i} (\bar{v}_j^{\bar{R}} - c_j^{\bar{R}})) \\ &= g_i^{\bar{R}}. \end{aligned}$$

A possible limitation of our approach is that each passenger needs to specify her value for each priority order. This can be a tedious task for a human to specify her exact value for each order. Furthermore, even specifying the value of a single priority order may be challenging for a human passenger, as this value requires an estimation of the monetary value of arriving at the destination. Instead, our mechanism can request a passenger to bid only on her value of time. That is, the amount (per minute) that a passenger would be willing to pay in order to save travel time. The mechanism can then automatically compute the values for each priority order based on the value of time, since the value of arriving to the destination is the same for each order. This results in a desirable property: a mechanism that encourages passengers to report their true value of time, and chooses the route accordingly.

4.1 Simulations

We now turn to evaluate the performance of our mechanism in a simulated environment. Our mechanism requires the definition of the ride cost (for each passenger and every priority order), as well as the valuation of each ride. Since our mechanism is truthful, we assume that $\bar{v}_i^R = v_i^R$. For every passenger, u_i, and priority order, R, we set the ride cost, c_i^R, to the Shapley value of that ride $\phi(u_i)$ (according to Section 3). Let t_i^R be the travel time from d_0 to d_i by the order R. Let c_i^p be the cost of a private and direct ride from d_0 (the airport) to d_i, and t_i^p be the time of this ride. Each passenger is assumed to have a "value or time" Vot_i. In order to evaluate the value of a shared ride, we assume that the passenger's utility from a private ride equals 0. That is, the value of a passenger arriving at her destination, v_i^{dest} is given by her "value of time" multiplied by the travel time of a private ride, added to the cost of a private ride, i.e., $v_i^{dest} = t_i^p \cdot Vot_i + c_i^p$.

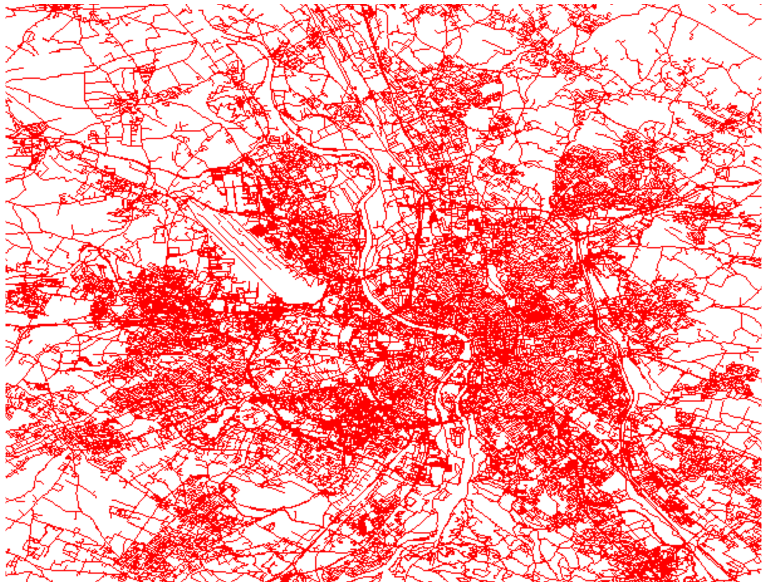

Fig. 1. A graph created from a map of the city of Toulouse, France.

Therefore, given a passenger u_i, and a priority order R, the value of the shared ride $v_i^R = v_i^{dest} - t_i^R \cdot Vot_i$.

We use the graph of the city of Toulouse, France[1] as presented in Figure 1. This graph includes the actual distances between the different vertices. The graph also includes the Toulouse-Blagnac airport. We cropped the graph to $40,000$ vertices, by running Dijkstra algorithm [6] starting at the airport, sorting all vertices by their distance from the airport, and removing all farther away vertices (including those that are unreachable).

Being a last mile problem, we set the origin vertex (d_0) to be the same for all passengers, the Toulouse-Blagnac airport. To convert the distances to travel times we set the average speed to 30 kph. We also set the cost per minute of travel to $\$1$. The capacity of each vehicle was set to 4 passengers. We repeat the following process 1000 times. We sample 12 destination vertices using a uniform distribution over all vertices. We assign the passengers associated with the destinations to vehicles such that the total cost of the rides would be minimized. From this assignment we choose the vehicles which have 4 passengers assigned to them, but disregard the priority order determined by the assignment algorithm. The value of time was randomly sampled from an average income per minute, computed using US data of income and hours of work for each decile[2]. This process resulted in a total of 7424 passengers assigned to 1856 vehicles.

---

[1] The graph of Toulouse was obtained from `https://www.geofabrik.de/data/shapefiles_toulouse.zip`.

[2] Data was obtained from: `https://dqydj.com/income-percentile-calculator`.

The main factor that we wanted to evaluate is the overhead of our mechanism. That is, we would like to ensure that the fees from the mechanism are not too high so that the total cost of the shared-ride would become inexpedient for the passengers. Indeed, the average fee in our simulations was only 8.48% from the overall cost ($c_i^{\hat{R}} - f_i^{\hat{R}}$). This overhead is significantly less than the commission charged by ride-sourcing services such as Uber and Lyft, which was reported to be at least 25% [19]. In addition, the average utility, $g_i^{\hat{R}}$ (which considers the travel time, the cost and the service fee) was 4.92; recall that we assume that the utility of a private ride is 0, this entails that our proposed ride-sharing mechanism seems quite beneficial for the passengers.

5 Conclusions

In this paper we focus on the domain of ride-sharing with passenger prioritization, that is, a group of passengers who travel using a shared-vehicle and are ordered according to some priority order. We show a method for efficient computation of the Shapley value, when there exist a priority order on the passengers. This order could be obtained by several means including the order of arrival to the service-station. For the case in which no such order is predetermined, we consider a VCG based mechanism for prioritization. The proposed mechanism obtains the value of time from each of the passengers and outputs a priority order. The mechanism is both efficient and truthful, and can easily be modified in order to take into account additional properties such as the travel distance, the number of additional passengers in the vehicle, and other properties that may affect the passengers' value from each priority order. We provide simulations showing that the cost paid to the mechanism service is reasonable.

References

1. Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., Rus, D.: On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. Proceedings of the National Academy of Sciences **114**(3), 462–467 (2017)
2. Besozzi, N., Ruschetti, L., Rossignoli, C., Strozzi, F.: The traveling salesman game for cost allocation: The case study of the bus service in castellanza. Game theory **2014** (2014)
3. Bistaffa, F., Farinelli, A., Chalkiadakis, G., Ramchurn, S.D.: Recommending fair payments for large-scale social ridesharing. In: Proceedings of the 9th ACM Conference on Recommender Systems. pp. 139–146 (2015)
4. Cheng, S.F., Nguyen, D.T., Lau, H.C.: Mechanisms for arranging ride sharing and fare splitting for last-mile travel demands. In: AAMAS. pp. 1505–1506 (2014)
5. Clarke, E.H.: Multipart pricing of public goods. Public choice **11**(1), 17–33 (1971)
6. Dijkstra, E.W.: A note on two problems in connexion with graphs. Numerische Mathematik **1**(1), 269–271 (1959)
7. Fiestras-Janeiro, M.G., García-Jurado, I., Meca, A., Mosquera, M.A.: Cost allocation in inventory transportation systems. TOP **20**(2), 397–410 (2012)
8. Fishburn, P., Pollak, H.: Fixed-route cost allocation. The American Mathematical Monthly **90**(6), 366–378 (1983)
9. Frisk, M., Göthe-Lundgren, M., Jörnsten, K., Rönnqvist, M.: Cost allocation in collaborative forest transportation. European Journal of Operational Research **205**(2), 448–458 (2010)

10. Groves, T.: Incentives in teams. Econometrica **41**(4), 617–631 (1973)
11. Guajardo, M., Rönnqvist, M.: A review on cost allocation methods in collaborative transportation. International transactions in operational research **23**(3), 371–392 (2016)
12. Kamar, E., Horvitz, E.: Collaboration and shared plans in the open world: Studies of ridesharing. In: Proceedings of the 21st International Jont Conference on Artifical Intelligence. pp. 187–194 (2009)
13. Kleiner, A., Nebel, B., Ziparo, V.A.: A mechanism for dynamic ride sharing based on parallel auctions. In: Proceedings of the 22nd International Jont Conference on Artifical Intelligence. pp. 266–272 (2011)
14. Littlechild, S.C., Owen, G.: A simple expression for the shapley value in a special case. Management Science **20**(3), 370–372 (1973)
15. Molenbruch, Y., Braekers, K., Caris, A.: Typology and literature review for dial-a-ride problems. Annals of Operations Research (2017)
16. Moulin, H., Shenker, S.: Strategyproof sharing of submodular costs: budget balance versus efficiency. Economic Theory **18**(3), 511–533 (2001)
17. Özener, O.Ö.: Developing a collaborative planning framework for sustainable transportation. Mathematical Problems in Engineering (2014)
18. Özener, O.Ö., Ergun, Ö.: Allocating costs in a collaborative transportation procurement network. Transportation Science **42**(2), 146–165 (2008)
19. Perea, C.: What's the real commission that uber takes from its drivers? https://therideshareguy.com/whats-the-real-commission -that-uber-takes-from-its-drivers-infographic/ (2016)
20. Potters, J.A., Curiel, I.J., Tijs, S.H.: Traveling salesman games. Mathematical Programming **53**(1-3), 199–211 (1992)
21. Psaraftis, H.N., Wen, M., Kontovas, C.A.: Dynamic vehicle routing problems: Three decades and counting. Networks **67**(1), 3–31 (2016)
22. Santos, A., McGuckin, N., Nakamoto, H.Y., Gray, D., Liss, S.: Summary of travel trends: 2009 national household travel survey. Tech. rep., U.S Departemnt of Transportation, Federal Highway Adminstration (2011)
23. Shapley, L.S.: A value for n-person games. Contributions to the Theory of Games **2**(28), 307–317 (1953)
24. Sun, L., Rangarajan, A., Karwan, M.H., Pinto, J.M.: Transportation cost allocation on a fixed route. Computers & Industrial Engineering **83**, 61–73 (2015)
25. Vickrey, W.: Counterspeculation, auctions, and competitive sealed tenders. The Journal of finance **16**(1), 8–37 (1961)
26. Yengin, D.: Characterizing the shapley value in fixed-route traveling salesman problems with appointments. International Journal of Game Theory **41**(2), 271–299 (2012)