Fair Sharing: The Shapley Value for Ride-Sharing and Routing Games

Chaya Levinger , Noam Hazon , Amos Azaria
Department of Computer Science
Data Science and Artificial Intelligence Center
Ariel University, Israel

Abstract

Ride-sharing services are gaining popularity and are crucial for a sustainable environment. A special case in which such services are most applicable, is the last mile variant. In this variant it is assumed that all the passengers are positioned at the same origin location (e.g. an airport), and each have a destination. One of the major issues in a shared ride is fairly splitting of the ride cost among the passengers.

In this paper we use the Shapley value, which is one of the most significant solution concepts in cooperative game theory, for fairly splitting the cost of a shared ride. We consider two scenarios. In the first scenario there exists a fixed priority order in which the passengers are dropped-off (e.g. elderly, injured etc.), and we show a method for efficient computation of the Shapley value in this setting. Our results are also applicable for efficient computation of the Shapley value in routing games.

In the second scenario there is no predetermined priority order. We show that the Shapley value cannot be efficiently computed in this setting. However, extensive simulations reveal that our approach for the first scenario can serve as an excellent proxy for the second scenario, outperforming other known proxies.

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 [Santos *et al.*, 2011] 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. 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 [Psaraftis *et al.*, 2016; Alonso-Mora *et al.*, 2017; Molenbruch *et al.*, 2017]. In this paper we study a fair allocation of the cost of the shared ride in the last mile variant [Cheng *et al.*, 2014]. That is, we analyze the cost allocation when all passengers are positioned at the same origin location. We concentrate on the Shapley value [Shapley, 1953] 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. The Shapley value has been even termed the most important normative division scheme in cooperative game theory [Winter, 2002]. However, the Shapley value depends on the travel cost of a ride of each subset of the passengers. Therefore, as stated by Özener and Ergun [Özener and Ergun, 2008], "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".

There are two possible general structures of the last-mile ride-sharing problem. In some cases 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). Clearly, in such cases, the prioritization is preserved when determining the travel cost of a ride with a subset of the passengers. We denote this problem as the *prioritized ride-sharing problem*. Indeed, in some scenarios there is no predetermined prioritization order. In such cases it is assumed that a ride with a subset of the passengers is performed using the shortest (or cheapest) path that traverses their destinations. We denote this problem as the *non-prioritized ride-sharing problem*.

The prioritized and the non-prioritized ride-sharing problems are closely related to traveling salesman games [Potters *et al.*, 1992]. In these games, a service provider makes a round-trip along the locations of several sponsors, where the total cost of the trip should be distributed among the sponsors. Specifically, the prioritized ride-sharing problem is similar to the fixed-route traveling salesman game, also known as routing game [Yengin, 2012], while the non-prioritized ride-sharing problem is similar to the traveling salesman game. Most of the works on traveling salesman games concentrated on finding an element of the core, a solution game concept which is different from the Shapley value. One exception is the work of Yengin [Yengin, 2012], who has studied the Shapley value of routing games and has conjectured that there is no efficient way for computing the Shapley value in routing games.

In this paper, we show an efficient computation of the Shapley value for the prioritized ride-sharing problem. Our method is based on smart enumeration of the components that are used in the computation of the Shapley value. Furthermore, our approach can be generalized to routing games, and we thus also provide an efficient way for computing the Shapley value in routing game. We then move to analyze the non-prioritized ride-sharing problem and show that, unless P=NP, there is no polynomial time algorithm for computing the Shapley value. Fortunately, we show through extensive simulations that when the given travel path is the shortest path the Shapley value of the prioritized ride-sharing problem can be used as an excellent proxy for the Shapley value of the non-prioritized ride-sharing problem.

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. 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:

1. We show an efficient method for computing the Shapley value of each passenger 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.
2. We show that, while there exist no polynomial algorithm for computing the Shapley value of the non-prioritized ride-sharing problem (unless P=NP), the Shapley value of the prioritized ride-sharing problem can be used as an excellent proxy for the Shapley value of the non-prioritized ride-sharing problem (under the assumption that the provided travel path is the shortest path).

2 Related Work

The ride-sharing cost allocation problems that we study are closely related to traveling salesman games [Potters *et al.*, 1992]. Specifically, the prioritized ride-sharing problem is similar to the fixed-route traveling salesman game [Fishburn and Pollak, 1983; Potters *et al.*, 1992; Besozzi *et al.*, 2014], also known as routing game [Yengin, 2012].

One variant of routing 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 Yengin [Yengin, 2012], 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.

The prioritized ride-sharing problem can also be interpreted as a generalization of the airport problem [Littlechild and Owen, 1973] to a two dimensional plane. 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 Shapley value for the traveling salesman game, which is related to our non-prioritized ride-sharing problem, has rarely received serious attention in the literature, due to its computational complexity. Notably, Aziz et al. [Aziz *et al.*, 2016] suggested a number of direct and sampling-based procedures for calculating the Shapley value for the traveling salesman game. They further surveyed several proxies for the Shapley value that are relatively easy to compute, and experimentally evaluate their performance. We develop a proxy for the Shapley value for the non-prioritized ride-sharing problem which is based on the Shapley-value for the prioritized ride-sharing problem, and compare its performance with proxies that are based on the work of Aziz et al.

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 [Guajardo and Rönnqvist, 2016]. The Shapley value was also investigated in this domain of collaborative transportation [Frisk *et al.*, 2010; Sun *et al.*, 2015]. In particular, Özener and Ergun [Özener and Ergun, 2008] stated that "we do not know of an efficient technique for calculating the Shapley value for the shippers' collaboration game". Indeed, Fiestras-Janeiro et al. [Fiestras-Janeiro *et al.*, 2012] developed the line rule, which is inspired by the Shapley value, but requires less computational effort and relates better with the core. However, the line rule is suitable for a specific inventory transportation problem. Özener [Özener, 2014] describes 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, Bistaffa et al. [Bistaffa *et al.*, 2015] 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).

3 Preliminaries

We are given a weighted graph $G(V, E)$ that represents a road network; V is the set of possible locations, and E is a set of weighted edges that represents the set of roads. We are given an ordered set $U = \{u_1, u_2, ..., u_n\}$ of passengers (users) that depart from the same origin, $d_0 \in V$, where passenger u_1 is dropped-off first, passenger u_2 is dropped off second, etc. Each passenger u_i has a corresponding destination $d_i \in V$. We denote by $\delta(u_i, u_j)$ the shortest travel distance between the destinations of u_i and u_j in G and $\delta(u_i, u_i) = 0$. To simplify the notation we define a dummy passenger, u_0, associated with d_0 and a dummy passenger, u_{n+1}, such that for every $i \in \{0, 1, ..., n\}$, $\delta(u_i, u_{n+1}) = 0$. Given a set $S \subseteq U$, let $c(S)$ be the cost associated with the subset S. That is, $c(U)$ is the total travel cost of the shared ride. We note that $c(S)$, where $S \subsetneq U$, depends on the order in which the passengers are dropped off. Therefore, $c(S)$ is defined differently in the prioritized ride-sharing problem (in which the original order in U is preserved) and in the non-prioritized ride-sharing problem (in which the shortest path is used to determine the order). The Shapley value for a passenger u_i is formally defined as:

$$\phi(u_i) = \sum_{S \subseteq U \setminus \{i\}} \frac{|S|!(|U| - |S| - 1)!}{|U|!} \big(c(S \cup \{i\}) - c(S)\big).$$

That is, the Shapley value is an average over the marginal costs of each passenger.

4 The Prioritized Ride-sharing Problem

In this section we assume that the passengers are ordered according to some predetermined priority order, and efficiently compute the payment for every passenger using the Shapley value. Unlike other related work [Potters *et al.*, 1992], we do not require that the priority order will be the optimal order that minimizes the total cost.

4.1 Notation

Given the ordered set of passengers U we assume that passenger u_1 has the highest priority, passenger u_2 has the second highest priority, etc. Given a set $S \subseteq U$, let $\tilde{S}$ be the set S ordered in an ascending order (according to the priority order), and let $S[i]$ be the passenger that is in the i-th position in $\tilde{S}$. For ease of notation we use $S[0]$ to denote u_0 and $S[|S| + 1]$ to denote u_{n+1}.

Given a set $S \subseteq U$, let $v(S)$ be the shortest travel distance of the path that starts at the origin d_0 and traverses all of the destinations of the passengers in S according to an ascending order. That is, $v(S) = \sum_{i=0}^{|S|-1} \delta(S[i], S[i + 1])$. This value ($v(S)$) serves as the cost associated with a subset of passengers, $c(S)$, in the computation of the Shapley value.

Let R be a permutation on U and let P_i^R be the set of the previous passengers to u_i in permutation R.

4.2 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, and we use the following formula to derive an efficient computation in the prioritized ride-sharing problem:

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

Given a permutation R and a passenger u_i, let $u_l \in P_i^R$ be a passenger such that $l < i$ and $\forall u_j \in P_i^R, j \le l$ or $i < j$. If no such passenger exists, then u_l is defined as u_0. Similarly, let $u_r \in P_i^R$ be a passenger such that $i < r$ and $\forall u_j \in P_i^R, j < i$ or $r \le j$. If no such passenger exists, then u_r is defined as u_{n+1}. We use ℓ (and *r*) to denote the position of u_l (and u_r) in the ordered $\tilde{P_i^R}$, respectively. If $u_l = u_0$ then $\ell = 0$, and if $u_r = u_{n+1}$ then *r*$= |P_i^R| + 1$. We note that $P_i^R[\ell] = u_l$, $P_i^R[r] = u_r$ and *r* $= \ell + 1$.

For example, assume $U = \{u_1, u_2, u_3, u_4, u_5, u_6\}$ and $R = \{u_6, u_2, u_5, u_4, u_3, u_1\}$, we get $P_4^R = \{u_6, u_2, u_5\}$ and thus $\tilde{P_4^R} = \{u_2, u_5, u_6\}$, $u_l = u_2$ (i.e., $\ell = 1$), $u_r = u_5$ (i.e., *r* $= 2$), and $P_4^R[\ell] = u_2$.

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.

Observation 1. $\phi(u_i) = \frac{1}{n!} \sum_{p=0}^{n-1} \sum_{q=p+1}^{n} \alpha_{p,q}^i \delta(u_p, u_q)$*, 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 $S \subseteq U$. By definition, $v(S) = \sum_{j=0}^{|S|-1} \delta(S[j], S[j + 1])$, such that $S[j] = u_p$ and $S[j + 1] = u_q$ where $p < q$. □

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

Lemma 1.

$$\phi(u_i) = \frac{1}{n!} \sum_R \Big(\delta(u_l, u_i) + \delta(u_i, u_r) - \delta(u_l, u_r) \Big)$$

Proof.

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

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

In addition,

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

$$\delta(u_l, u_i) + \delta(u_i, u_r) + \sum_{j=r}^{|P_i^R|-1} \delta(P_i^R[j], P_i^R[j+1]).$$

By definition,

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

$$\tfrac{1}{n!} \sum_R \Bigg(\sum_{j=0}^{\ell-1} \delta(P_i^R[j], P_i^R[j+1]) + \delta(u_l, u_i) + \delta(u_i, u_r) + \sum_{j=r}^{|P_i^R|-1} \delta(P_i^R[j], P_i^R[j+1]) - \Big(\sum_{j=0}^{\ell-1} \delta(P_i^R[j], P_i^R[j+1]) + \delta(u_l, u_r) + \sum_{j=r}^{|P_i^R|-1} \delta(P_i^R[j], P_i^R[j+1]) \Big) \Bigg) =$$

$$\tfrac{1}{n!} \sum_R \Big(\delta(u_l, u_i) + \delta(u_i, u_r) - \delta(u_l, u_r) \Big)$$ □

Following Observation 1 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, $\phi(u_i) = \sum_{p=0}^{i} \sum_{q=i}^{n} \beta_{p,q}^i \delta(u_p, u_q)$, 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 $\delta(u_p, u_q)$, where $p \leq i \leq q$. There are several terms in this sum:

$\beta_{0,i}^i$ multiplies $\delta(u_0, u_i)$. Now, $\delta(u_0, u_i)$ appears in $\phi(u_i)$ in every permutation R when $u_l = u_0$. That is, in all of the permutations where passenger u_i appears before any other passenger u_x such that $x < i$. We now count the number of such permutations. There are $\binom{n}{i}$ options to place the passengers $u_1, u_2, ..., u_i$ among the n available passengers. For each such option there are $(i-1)!$ options to order the passengers $u_1, u_2, ..., u_i$ such that u_i is the first passenger among them. Finally, there are $(n-i)!$ options to order the passengers $u_{i+1}, u_{i+2}, ..., u_n$. Therefore, $\delta(u_0, u_i)$ 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 p, q such that $p < i < q$, $\beta_{p,q}^i$ multiplies $\delta(u_p, u_q)$. Now, $\delta(u_p, u_q)$ appears negatively in $\phi(u_i)$ in every permutation R when $u_l = u_p$ and $u_r = u_q$. That is, in all of the permutations where passengers u_p, u_q appear before u_i (i.e., $u_p, u_q \in P_i^R$), but any other passenger u_x such that $p < x < q, x \neq i$, appears after u_i. We now count the number of such permutations. There are $\binom{n}{q-p+1}$ options to place the passengers $u_p, u_{p+1}, ..., u_i, ..., u_q$ among the n available passengers. For each such option there are $(q-p+1-3)!$ options to order the passengers $u_p, u_{p+1}, ..., u_i, ..., u_q$ such that u_p is the first passenger, u_q is the second and u_i is the third passenger among them. Similarly, there are $(q-p+1-3)!$ options to order these passengers such that u_q is the first passenger, u_p is the second and u_i is the third. Finally, there are $(n-(q-p+1))!$ options to order the passengers $u_1, u_2, ..., u_{p-1}, u_{q+1}, u_{q+2}, ..., u_n$. Therefore, $\delta(u_p, u_q)$ 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 $\beta_{p,q}^i = -\frac{2}{(q-p-1)\cdot(q-p)\cdot(q-p+1)}$.

Similarly, we get that $\beta_{0,q}^i = -\frac{1}{q\cdot(q-1)}$, $\beta_{p,i}^i = \frac{1}{(i-p)\cdot(i-p+1)}$ and $\beta_{i,q}^i = \frac{1}{(q-i)\cdot(q-i+1)}$. □

We note that the prioritized ride-sharing problem is very similar to the setting of routing games [Potters *et al.*, 1992]. 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 problem 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 (sketch). We use our previous definitions and results with the following slight modifications. The dummy passenger u_{n+1} becomes associated with d_0. Thus, $\delta(u_i, u_{n+1}) = \delta(u_i, u_0)$. In Observation 1 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 $\sum_{p=0}^{i} \beta_{p,n+1}^i \delta(u_p, u_{n+1})$ to the calculation of $\phi(u_i)$, where for $p < i$, $\beta_{p,n+1}^i = -\frac{1}{(n-p)\cdot(n-p+1)}$ and $\beta_{i,n+1}^i = \frac{1}{n-i+1}$. □

Note that this is an unexpected result, since it refutes the conjecture in [Yengin, 2012] that there is no efficient way for computing the Shapley value in routing games.

5 Non-prioritized Ride-sharing Problem

Similar to the prioritized ride-sharing problem we are given an initial priority order, which determines the drop-off order of the passengers. However, in the non-prioritized variant we do not enforce the fixed order for every subset of passengers. Instead, given a strict subset of passengers S, the cost associated with it, $c(S)$, is the length of the shortest path that traverses all of the destinations of the passengers in S.

5.1 The Hardness of the Non-prioritized Ride-sharing Problem

In Section 4 we showed that we can efficiently compute the Shapley value for the prioritized ride-sharing problem. In essence, the computation could be done efficiently since most of the travel distances cancel out, and only a polynomial number of terms remain in the computation. Unfortunately, this is not the case with the non-prioritized ride-sharing problem, where the Shapley value cannot be computed efficiently unless $P = NP$.

Clearly, finding the length of the shortest path (not necessarily a simple path) that starts at a specific node, v_0, and traverses all nodes in a graph (without returning to the origin) cannot be performed in polynomial time, unless $P = NP$. We denote this problem as *path-TSP*. We use the path-TSP to show that computing the Shapley value for the prioritized ride-sharing cannot be done efficiently, unless $P = NP$ (we note that theorem 1 in [Aziz *et al.*, 2016] has a flaw, and therefore cannot be used).

Theorem 3. *There is no polynomial time algorithm that computes the Shapley value for a given passenger in the non-prioritized ride-sharing problem unless $P = NP$.*

Proof. Given an instance of the path-TSP problem on a graph $G(V, E)$ we denote the solution by x. We construct an instance of the non-prioritized ride-sharing problem as follows. We build a graph $G'(V', E')$, where we add a node v', i.e., $V' = V \cup \{v'\}$. If $e \in E$ then $e \in E'$, and for all $v \in V$, $(v, v') \in E'$ with a weight of M, where M is the sum of weights of all the edges in E. Finally, we set $U = D = V' \setminus \{v_0\}$, $d_0 = v_0$, and the drop-off order is arbitrarily chosen. Recall that $c(U)$ is the total travel cost associated with the chosen drop-off order. We ask to compute the Shapley value of the passenger u' that is associated with the destination v'.

Clearly, the marginal contribution of u' to any strict subset of $U \setminus \{u'\}$ is exactly M. However, the marginal contribution of u' to the complete set $U \setminus \{u'\}$ is exactly $c(U)$ minus x (the length of the shortest path starting at v_0 and traversing all nodes in V). That is,

$$\phi(u') = \frac{(|U| - 1)!}{|U|!}(c(U) - x) + \frac{|U|! - (|U| - 1)!}{|U|!}M$$

After some simple mathematical manipulations we get that $x = (|U| - 1)M - |U|\phi(u') + c(U)$. Therefore, if we can compute $\phi(u')$ in polynomial time then we can solve the path-TSP problem in polynomial time, which is not possible unless $P = NP$. □

5.2 Shapley Approximation based on a Prioritized Order

In Section 4 we presented a method for efficiently computing the Shapley value when a prioritization exists. In this section we show that our solution may be also applicable to the non-prioritized ride-sharing problem as an efficient proxy for the Shapley value. We term our proxy SHAPO: SHapley Approximation based on a Prioritized Order.

We compare SHAPO with the following three proxies for computing the Shapley value in traveling salesman games, that are in use in real-world applications [Aziz *et al.*, 2016].

Depot Distance This method divides the total ride cost proportionally to the distance from the depot, i.e.

$$Depot(u_i) = \frac{\delta(u_0, u_i)}{\sum_{j=1}^{n} \delta(u_0, u_j)} c(U).$$

For example, a passenger traveling to a destination that is twice as distant from the origin as another passenger has to pay twice the cost, regardless of the actual travel path. We note that this method has outperformed all other methods in [Aziz *et al.*, 2016] on real data. However, since [Aziz *et al.*, 2016] consider also a form of measure that is not applicable to our domain (the ranking over the actors), they recommend using other proxies.

Shortcut Distance This method divides the total cost proportional to the change realized by skipping a destination when traversing the given path. Formally, let $Cut_i = \delta(u_{i-1}, u_i) + \delta(u_i, u_{i+1}) - \delta(u_{i-1}, u_{i+1})$. Then,

$$Shortcut(u_i) = \frac{Cut_i}{\sum_{j=1}^{n} Cut_j} c(U).$$

Re-routed Margin This method is a more sophisticated realization of the shortcut distance method. That is, instead of using the given path when skipping a destination, we compute the optimal path. Formally,

$$Reroute(u_i) = \frac{c(U) - c(U \setminus \{u_i\})}{\sum_{j=1}^{n} c(U) - c(U \setminus \{u_j\})} c(D).$$

Note that when evaluating this proxy we need to solve n TSPs, one for leaving out each destination. This is the only proxy we consider that requires a non-negligible time to compute.

Experimental Settings

In order to evaluate the performance of SHAPO, we evaluated each of the methods for $3, 4, 5, 6, 7, 8$ and 9 passengers. For the road network we used the graph of the city of Toulouse, France[1]. This graph includes the actual distances between the different vertices. To convert the distances to travel costs we assumed a cost of \$1 per kilometer. The graph also includes the Toulouse-Blagnac airport, which was set as the origin (d_0). We cropped the graph to $40,000$ vertices, by running Dijkstra algorithm [Dijkstra, 1959] starting at the airport, sorting all vertices by their distance from the airport, and removing all farther away vertices (including those that are unreachable). The destination vertices were randomly sampled for every passenger using a uniform distribution over all vertices, and each of the methods was evaluated 100 times against the true Shapley value of all passengers.

For running the simulations we assume that the given order of the passengers is according to the shortest path. This is a reasonable assumption, since if there is no prioritization, it is very likely that, in order to reduce the overall cost, the vehicle would travel using the shortest path (computed once). We conjecture that the results presented in this paper will carry-out also to situations in which the given passenger order is very close to being optimal (but not necessarily the exact optimal order), but we leave it for future investigation.

Results

Figure 1 presents the running time, in seconds, required to compute the Shapley value and its proxies for all passengers on a single instance (in logarithmic scale). As expected, we can compute the proxies, except for the Re-routed margin proxy, almost instantaneously. However, due to the extensive time required to compute the Shapley value, and since we evaluate each method 100 times, we only evaluate the performance of all methods with up-to 9 passengers.

[1] obtained from `https://www.geofabrik.de/data/shapefiles_toulouse.zip`

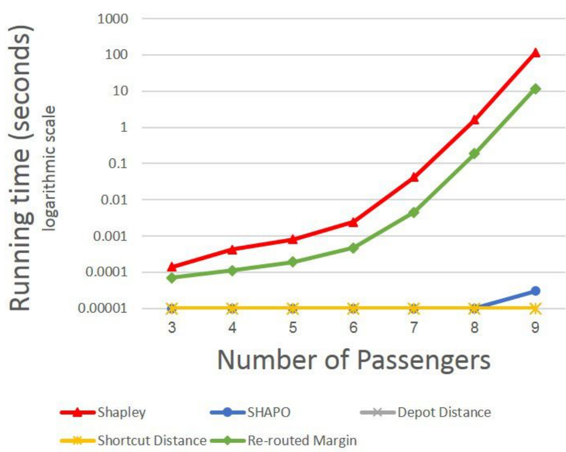

Figure 1: Running time, in seconds, required to compute a single instance of the Shapley value (in logarithmic-scale).

We evaluate the performance of SHAPO against the three other proxies using 5 different statistical measures (averaged on all 100 iterations). We use $X(u_i)$ to denote the estimated Shapley value by the evaluated proxy.

1. **Percent**: The average percentage of the deviation from the Shapley value. Formally, $Percent = \frac{1}{n}\sum_{i=1}^{n}\frac{|X(u_i)-\phi(u_i)|}{\phi(u_i)}$.
2. **MAE**: The mean absolute error, $MAE = \frac{1}{n}\sum_{i=1}^{n}|X(u_i)-\phi(u_i)|$.
3. **MSE**: The mean squared error, $MSE = \frac{1}{n}\sum_{i=1}^{n}(X(u_i) - \phi(u_i))^2$. This measure gives higher weight to larger deviations.
4. **RMSE**: The root mean squared error, $RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(X(u_i)-\phi(u_i))^2}$.
5. **Max-Error**: The maximum deviation among all passengers between the real and estimated Shapley value, $Max = \max_{i=1}^{n}(|X(u_i)-\phi(u_i)|)$.

The results are depicted in Tables 1, 2, 3, 4 and 5. SHAPO significantly outperforms the other proxies in all measures, with any number of passengers evaluated. Despite the depot distance method outperforming the other two methods, SHAPO is between 5.5 to 42.3 times better than the depot distance in all measures. Note that the units of MAE and Max-Error are dollars. That is, as depicted in Table 2, SHAPO deviated by only 19 cents, on average, from the actual Shapley value. The depot distance deviated by \$1.33, while the averaged shared-ride cost per passenger was approximately \$5. Similarly, the maximal deviation of SHAPO was less than 44 cents (on average), while the maximal deviation of the depot distance was more than \$2.9.

6 Conclusions

The Shapley value is considered one of the most important division scheme of revenues or costs, but its direct computation is often not practical for a reasonable size game. Therefore, Mann and Shapely [Mann and Shapley, 1962] suggest to consider restrictions and constraints in order to find games where the Shapley value can be efficiently computed. We showed that the prioritized ride-sharing problem is an example of such a game by showing that the Shapley value can be efficiently computed. However, we show that the non-prioritized ride-sharing problem, which is possibly the next level of generalization, cannot be efficiently computed (unless $P = NP$). Interestingly, the prioritized ride-sharing can still serve as an efficient proxy for the Shapley value of the non-prioritized ride-sharing problem where the provided travel path is the shortest path.

There are several interesting directions for future work. One possible direction is to compare our proxy for computing the Shapley value in the non-prioritized ride-sharing problem to a sampling based approach [Castro *et al.*, 2009]. It is expected that a sampling based approach will be more accurate if there is a sufficient number of samples, but it will certainly require a lot more computation time. It is thus interesting to analyze when our proxy is still better than a sampling-based approach, and when it is the point in which a sampling-based approach becomes better than our proxy. From a theoretical perspective, we showed that computing the Shapley value for the non-prioritized ride-sharing problem is a hard problem. However, the hardness may be derived also from the hardness of path-TSP. There are several polynomial time approximation and heuristics for TSP that can be adjusted for path-TSP. It is thus interesting to analyze the computational complexity of finding the Shapley value, where $c(S)$ is computed using one of these approximations or heuristics.

7 Acknowledgments

This research was supported in part by the Ministry of Science, Technology & Space, Israel. It was also supported in part by Volkswagen Foundation (Volkswagenstiftung) under grant "EC-RIDER: Explainable AI Methods for Human-Centric Ridesharing".

References

[Alonso-Mora *et al.*, 2017] Javier Alonso-Mora, Samitha Samaranayake, Alex Wallar, Emilio Frazzoli, and Daniela Rus. On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences*, 114(3):462–467, 2017.

[Aziz *et al.*, 2016] Haris Aziz, Casey Cahan, Charles Gretton, Philip Kilby, Nicholas Mattei, and Toby Walsh. A study of proxies for shapley allocations of transport costs. *Journal of Artificial Intelligence Research*, 56:573–611, 2016.

[Besozzi *et al.*, 2014] Nicola Besozzi, Luca Ruschetti, Chiara Rossignoli, and Fernanda Strozzi. The traveling salesman game for cost allocation: The case study of the bus service in castellanza. *Game theory*, 2014, 2014.

[Bistaffa *et al.*, 2015] Filippo Bistaffa, Alessandro Farinelli, Georgios Chalkiadakis, and Sarvapali D Ramchurn. Recommending fair payments for large-scale social ridesharing. In *Proceedings of the 9th ACM Conference on Recommender Systems*, pages 139–146, 2015.

	3	4	5	6	7	8	9	AVG
SHAPO	**1.84%**	**3.44%**	**4.49%**	**4.68%**	**5.27%**	**6.53%**	**5.94%**	**4.60%**
Depot Distance	20.02%	26.36%	29.52%	35.28%	35.78%	35.92%	35.91%	31.26%
Shortcut Distance	34.79%	41.55%	45.44%	53.70%	54.23%	53.35%	55.89%	48.42%
Re-routed Margin	84.59%	76.13%	79.65%	119.90%	80.57%	85.97%	72.83%	85.66%

Table 1: Average percentage of the deviation from the Shapley value (Percent). Averaged over 100 iterations. Lower is better.

	3	4	5	6	7	8	9	AVG
SHAPO	**$0.11**	**$0.18**	**$0.20**	**$0.20**	**$0.20**	**$0.22**	**$0.21**	**$0.19**
Depot Distance	$1.11	$1.29	$1.34	$1.53	$1.38	$1.35	$1.28	$1.33
Shortcut Distance	$1.87	$2.19	$2.23	$2.47	$2.27	$2.07	$2.09	$2.17
Re-routed Margin	$3.21	$2.81	$2.45	$2.43	$2.10	$2.03	$1.85	$2.41

Table 2: The mean absolute error of the deviation from the Shapley value (MAE). Averaged over 100 iterations. Lower is better.

	3	4	5	6	7	8	9	AVG
SHAPO	**0.068**	**0.115**	**0.124**	**0.098**	**0.102**	**0.159**	**0.117**	**0.112**
Depot Distance	1.988	2.634	3.054	4.130	3.327	3.366	3.066	3.081
Shortcut Distance	5.805	8.431	8.937	10.840	9.749	8.097	7.731	8.513
Re-routed Margin	15.951	12.404	9.587	10.174	7.399	7.168	5.986	9.810

Table 3: The mean squared error of the deviation from the Shapley value (MSE). Averaged over 100 iterations. Lower is better.

	3	4	5	6	7	8	9	AVG
SHAPO	**0.121**	**0.204**	**0.239**	**0.246**	**0.259**	**0.299**	**0.277**	**0.235**
Depot Distance	1.205	1.474	1.594	1.859	1.699	1.703	1.617	1.593
Shortcut Distance	2.112	2.556	2.698	3.068	2.904	2.675	2.662	2.668
Re-routed Margin	3.540	3.200	2.866	2.951	2.517	2.508	2.292	2.839

Table 4: The root mean squared error of the deviation from the Shapley value (RMSE). Averaged over 100 iterations. Lower is better.

	3	4	5	6	7	8	9	AVG
SHAPO	**$0.17**	**$0.30**	**$0.39**	**$0.44**	**$0.49**	**$0.62**	**$0.58**	**$0.43**
Depot Distance	$1.66	$2.24	$2.73	$3.44	$3.32	$3.51	$3.47	$2.91
Shortcut Distance	$2.81	$3.92	$4.69	$5.66	$5.77	$5.49	$5.56	$4.84
Re-routed Margin	$4.81	$4.74	$4.60	$5.32	$4.67	$4.95	$4.71	$4.83

Table 5: The maximum deviation among all passengers between the real and estimated Shapley value (Max-Error). Averaged over 100 iterations. Lower is better.

[Castro *et al.*, 2009] Javier Castro, Daniel Gómez, and Juan Tejada. Polynomial calculation of the shapley value based on sampling. *Computers & Operations Research*, 36(5):1726–1730, 2009.

[Cheng *et al.*, 2014] Shih-Fen Cheng, Duc Thien Nguyen, and Hoong Chuin Lau. Mechanisms for arranging ride sharing and fare splitting for last-mile travel demands. In *Proceedings of the 13th International Conference on Autonomous Agents and Multi-agent Systems*, pages 1505–1506, 2014.

[Dijkstra, 1959] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271, 1959.

[Fiestras-Janeiro *et al.*, 2012] M. G. Fiestras-Janeiro, I. García-Jurado, A. Meca, and M. A. Mosquera. Cost allocation in inventory transportation systems. *TOP*, 20(2):397–410, 2012.

[Fishburn and Pollak, 1983] PC Fishburn and HO Pollak. Fixed-route cost allocation. *The American Mathematical Monthly*, 90(6):366–378, 1983.

[Frisk *et al.*, 2010] Mikael Frisk, Maud Göthe-Lundgren, Kurt Jörnsten, and Mikael Rönnqvist. Cost allocation in collaborative forest transportation. *European Journal of Operational Research*, 205(2):448–458, 2010.

[Guajardo and Rönnqvist, 2016] Mario Guajardo and Mikael Rönnqvist. A review on cost allocation methods in collaborative transportation. *International transactions in operational research*, 23(3):371–392, 2016.

[Littlechild and Owen, 1973] Stephen C Littlechild and Guillermo Owen. A simple expression for the shapley value in a special case. *Management Science*, 20(3):370–372, 1973.

[Mann and Shapley, 1962] Irwin Mann and Lloyd S Shapley. Values of large games. 6: Evaluating the electoral college exactly. Technical report, RAND CORP SANTA MONICA CA, 1962.

[Molenbruch *et al.*, 2017] Yves Molenbruch, Kris Braekers, and An Caris. Typology and literature review for dial-a-ride problems. *Annals of Operations Research*, 2017.

[Özener and Ergun, 2008] Okan Örsan Özener and Özlem Ergun. Allocating costs in a collaborative transportation procurement network. *Transportation Science*, 42(2):146–165, 2008.

[Özener, 2014] Okan Örsan Özener. Developing a collaborative planning framework for sustainable transportation. *Mathematical Problems in Engineering*, 2014.

[Potters *et al.*, 1992] Jos AM Potters, Imma J Curiel, and Stef H Tijs. Traveling salesman games. *Mathematical Programming*, 53(1-3):199–211, 1992.

[Psaraftis *et al.*, 2016] Harilaos N Psaraftis, Min Wen, and Christos A Kontovas. Dynamic vehicle routing problems: Three decades and counting. *Networks*, 67(1):3–31, 2016.

[Santos *et al.*, 2011] Adella Santos, Nancy McGuckin, Hikari Yukiko Nakamoto, Danielle Gray, and Susan Liss. Summary of travel trends: 2009 national household travel survey. Technical report, U.S Department of Transportation, Federal Highway Administration, 2011.

[Shapley, 1953] Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317, 1953.

[Sun *et al.*, 2015] Lei Sun, Atul Rangarajan, Mark H Karwan, and Jose M Pinto. Transportation cost allocation on a fixed route. *Computers & Industrial Engineering*, 83:61–73, 2015.

[Winter, 2002] Eyal Winter. The shapley value. *Handbook of game theory with economic applications*, 3:2025–2054, 2002.

[Yengin, 2012] Duygu Yengin. Characterizing the shapley value in fixed-route traveling salesman problems with appointments. *International Journal of Game Theory*, 41(2):271–299, 2012.