Social Aware Assignment of Passengers in Ridesharing

Paper Id: 1862

Abstract

Ride-sharing services are gaining popularity and are crucial for a sustainable environment. We analyze the assignment of passengers in a shared ride, which considers the social relationship among the passengers. Namely, there is a maximum fixed number of passengers in each vehicle, denoted by k, and the goal is to assign passengers such that the number of friendship relations is maximized. We show that the problem is computationally hard when $k > 2$, but we provide an efficient algorithm with an approximation ratio of $\frac{1}{k-1}$. We further show that this bound is tight. In addition, we analyze the distributed case of this problem, in which the passengers split-up arbitrary but maximally, and show that this procedure achieves an approximation ratio of, at most, $\frac{1}{k}$.

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 (i.e., when several passengers travel together, the travel distance counts for every passenger in the vehicle). This extremely low average vehicle occupancy entails a very large number of vehicles on the road that collectively contribute to carbon dioxide emissions, fuel consumption, air pollution and an increase in traffic load, which in turn requires additional investment in enlarging the road infrastructure. 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.

In this paper we focus on the following ridesharing scenario. Consider a group of travelers (passengers) who are located at some origin, would like to reach the same destination, and later return. Each of the users has her own vehicle; but each passenger has a preference related to who will be with her in the vehicle. Namely, each passenger would rather share a vehicle with as many of her friends during the ride, and thus the utility of each passenger is the number of friends traveling with her. However, the vehicles have a limited capacity; this capacity can either be a physical constraint of the vehicles, or the maximal number of passengers willing to travel together. The goal is to assign the passengers to vehicles while maximizing the social welfare (the sum of all passengers' utilities).

We formulate the described problem as the social aware assignment problem, which assumes that the agents' utilities depend on a social network that represents the social relationships among the agents. The social network is modeled as an unweighted graph where the vertices are agents and the edges indicate friendship among the agents. The utility function of an agent is the number of friends she has within the coalition to which she is assigned. In addition, there is a hard constraint on the maximal size, k, of each coalition, which is bounded by the capacity of the vehicles. In this paper, we show that the social aware assignment problem is computationally hard for any $k \geq 3$, i.e., the decision variant of the social aware assignment is in NP-Complete. Therefore, we provide an approximation algorithm and prove that it has an approximation ratio of $\frac{1}{k-1}$. Furthermore, we show that this approximation ratio is tight for our algorithm, i.e., in some cases, our algorithm will provide a social welfare of exactly $\frac{1}{k-1}$ of an optimal assignment. In addition, we analyze the scenario in which there is no centralized mechanism to assign passengers to vehicles, and we assume that the passengers join vehicles in an arbitrary, but maximally manner. We show that, unfortunately, the approximation ratio drops to at most $\frac{1}{k}$. Since, in most cases, the vehicle capacity, k, is relatively small (3,4 or 5), this difference is significant.

The social aware assignment problem belongs to the field of coalition formation, which is an important research branch within multiagent systems [Chalkiadakis *et al.*, 2011]. It analyses the outcome that results when a set of agents is partitioned into coalitions. Actually, our model is a special case of simple Additively Separable Hedonic Games (ASHGs) [Bogomolnaia *et al.*, 2002].

2 Related Work

We begin by providing a brief review of the current literature on the broad class of Vehicle Routing and scheduling Problems (VRPs), in order to place our ridesharing problem in an appropriate context. The VRP was first introduced by [Dantzig and Ramser, 1959]. The growing body of research on routing problem over the past 50 years has led to the development of several research communities, which sometimes denote the same problem types by various names. In particular, the traditional VRP and some of its extensions deal with finding an optimal set of routes for a fleet of vehicles to traverse in order deliver or pickup some goods to a given set of costumers. We refer to the comprehensive survey of [Parragh *et al.*, 2008a] on this class of problems, which they denote by Vehicle Routing Problems with Backhauls (VRPB). A more recent survey, that also defines a taxonomy to classify the various variants of VRP by 11 criteria, is given by [Psaraftis *et al.*, 2016]. A second class of problems, that is denoted by Parragh et al. as Vehicle Routing Problems with Pickups and Deliveries (VRPPD), deal with all those problems where goods are transported between pickup and delivery customers. We refer to the survey of [Parragh *et al.*, 2008b] on this class of problems. One subclass of VRPPD compromises the dial-a-ride problem (DARP), where the goods that are transported are passengers with associated pickup and delivery points. It was noted by [Cordeau and Laporte, 2003] that the DARP is distinguished from other problems in vehicle routing since transportation cost and user inconvenience must be weighed against each other in order to provide an appropriate solution. Therefore, the DARP typically includes more quality constraints that aim at capturing the user's inconvenience. We refer to a recent survey on DARP by [Molenbruch *et al.*, 2017], which also makes this distinction.

Another domain closely related to our problem is sometimes referred to as car-pooling. In this domain, ordinary drivers, may opt to take an additional passenger on their way to a shared destination. The common setting of car-pooling is within a long-term commitment between people to travel together for a particular purpose, where ridesharing is focused on single, non-recurring trips. Indeed, several works investigated car-pooling that can be established on a short-notice, and they refer to this problem as ridesharing [Agatz *et al.*, 2012; Montazery and Wilson, 2016]. Schleibaum et al. [Schleibaum *et al.*, 2020] conduct a survey with an attempt to analyze which measures affect user satisfaction from a ride. However, they do not consider a scenario similar to ours, in which some passengers are friends of others, but rather seem to assume that all passengers are strangers. Guidotti et al. [Guidotti *et al.*, 2015] measure the similarity between different users, and assume some users prefer traveling with similar users while some rather traveling with passengers who are more diverse. They term this preference enjoyability. They assume a given threshold on the distance each passenger is willing to walk and another threshold on the time each passenger is willing wait. They compose two heuristic based models, one for minimizing the number of drivers, and the second model which also tries to maximize the enjoyability of the passengers. To the best of our knowledge, no previous work has considered the assignment of passengers to vehicles with an attempt to maximize the total number of friendships inside all vehicles.

From the field of coalition formation, Sless et al. [2018] tackle a problem similar to ours. Similar to our work, they assume a friendship graph and attempt to maximize the number of friends in each group. However, in their setting the agents must be partitioned into exactly k groups without any restriction on each group's size.

Wright and Vorobeychik [2015] also study a model of ASHG where there is a restriction on the size of each coalition. Within their model, they propose a strategyproof mechanism that achieves good and fair experimental performance, despite not having a theoretical guarantee.

Flammini et al. [Flammini *et al.*, 2021] study the problem of the online coalition structure generation problem. Similar to our work, they also consider the scenario that the coalitions are bounded by some number. They consider two cases for the value of a coalition, the sum of the weights of its edges, which is similar to our work, and the sum of the weights of its edges divided by its size. However, in both cases they only consider the online version, i.e., the agents arrive sequentially and must be assigned to a coalition as they arrive. This assignment cannot be adapted later on, and must remain. They show that a simple greedy algorithm achieves an approximation ratio of $\frac{1}{k}$ when the value of the coalition is the sum of the weights.

3 Preliminaries

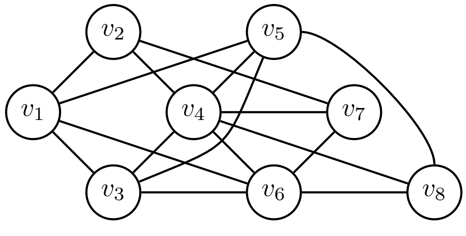
In this paper we analyse the problem of the assignment in ride-sharing problem, while maintaining the human-centric approach. Specifically, our goal is to assign the users to vehicles such that each user will be matched with as many friends as possible in the same vehicle, while each vehicle is limited to a number of passengers, k. Formally,

Definition 3.1 (Social aware assignment)**.** We are given a number k and an undirected friendship graph $G = (V, E)$ where $(v_i, v_j) \in E$ if v_i and v_j are friends of each other. The goal is to find an assignment P, which is a partition of the set V, such that $\forall S \in P, |S| \leq k$, and the value of P, $V_P = |\{(v_i, v_j) \in E\text{: } \exists S \in P \text{ where } v_i \in S \text{ and } v_j \in S\}|$ is maximized.

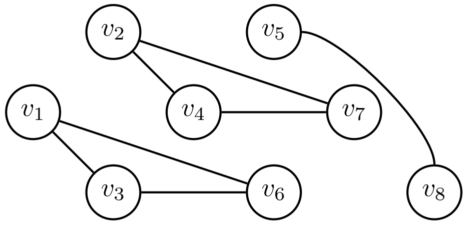
For example, given the graph in Figure 1a and a vehicle size limit $k = 3$, the value of the partition $P = \{\{v_1, v_3, v_6\}, \{v_2, v_4, v_7\}, \{v_5, v_8\}\}$, shown in Figure 1b, equals 7. Indeed, this is the optimal partition since there is no other partition with higher value. Clearly, the decision variant of the social aware assignment problem is to decide whether there exists a partition with a value of at least v.

4 The Hardness of the Social Aware Assignment Problem

The social aware assignment problem when $k = 2$ is equivalent to the maximum matching problem, and thus it can be computed in polynomial time [Edmons, 1965]. However, our



(a) The graph G



(b) An optimal partition in G

Figure 1: An example for the social aware assignment problem where $k = 3$.

problem becomes intractable when $k \geq 3$. For the hardness proof, we define for each $k \in \mathbb{N}$ the $Cliques_k$ problem, which is as follows.

Definition 4.1 ($Cliques_k$)**.** Given an undirected graph $G = (V, E)$, decide whether V can be partitioned into disjoint cliques, such that each clique is composed of exactly k vertices.

Clearly, $Cliques_2$ can be decided in polynomial time by computing a maximum matching of the graph G, M, and testing whether $|M| = \frac{|V|}{2}$. However, $Cliques_k$ becomes hard when $k \geq 3$.

Lemma 1. *$Cliques_k$ is in NP-Complete for every $k \geq 3$.*

Proof. Clearly, $Cliques_k$ is in NP for every k. We use induction to show that any $Cliques_k$ is in NP-Hard for every $k \geq 3$. $Cliques_3$ is known as the 'partition into triangles' problem, which was shown to be in NP-Complete [di Complessita and Rizzi, 2004]. Given that $Cliques_k$ is in NP-Hard we show that $Cliques_{k+1}$ is also in NP-Hard. Given an instance of the $Cliques_k$ on a graph $G(V, E)$, we construct the following instance. We build a graph $G'(V', E')$, in which we add a set of nodes $\hat{V} = \hat{v}_1, ..., \hat{v}_{\frac{|V|}{k}}$, i.e., $V' = V \cup \hat{V}$. If $e \in E$ then $e \in E'$, and for every $v \in V, \hat{v} \in \hat{V}$ we add $(v, \hat{v})$ to E'. Clearly, V can be partitioned into disjoint cliques with exactly k vertices if and only if V' can be partitioned into disjoint cliques with exactly $k + 1$ vertices. $\square$

Theorem 1. *The decision variant of the social aware assignment problem is in NP-Complete.*

Proof. Clearly the problem is in NP, since if we are given a partition P, a limit k, and the value υ, we can easily check that $\forall U \in P$, $|U| \leq k$ and the value of P, $V_P \geq \upsilon$ in polynomial time. For the hardness proof we use the $Cliques_k$ problem. Given an instance of $Cliques_k$ on a graph $G(V, E)$, we use the same graph with the same k and $\upsilon = \frac{|V|(k-1)}{2}$ as an instance to the social aware assignment problem. Clearly, V can be partitioned into disjoint cliques with exactly k vertices if and only if there exist a partition P such that $\forall S \in P, |S| \leq k$, and $V_P = \upsilon$. $\square$

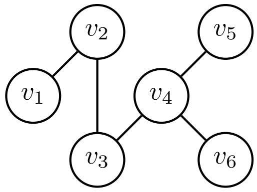
5 Approximation of the Social Aware Assignment Problem

Since we showed that the social aware assignment problem is in NP-Complete, we now provide the Match and Merge (MnM) algorithm (Algorithm 1), which is an approximation algorithm for any $k \geq 3$. The algorithm consists of $k - 1$ rounds. Each round is composed of a matching phase followed by a merging phase. Specifically, in round l MnM computes a maximum matching, $M_l \subseteq E_l$, for G_l (where $G_1 = G$). In the merging phase, MnM creates a graph G_{l+1} that includes a unified node for each pair of matched nodes. G_{l+1} also includes all unmatched nodes, along with their edges to the unified nodes (lines 10-13). Clearly, each node in V_l is composed of up-to l nodes from V_1. Finally, MnM returns the partition, P, of all the matched sets. For example, given the graph G_1 in Figure 2a and $k = 4$, the algorithm finds a maximum matching $M_1 = \{(v_1, v_2), (v_3, v_4)\}$ shown in Figure 2b. It then creates the graph G_2, as shown in Figure 2c, and finds a maximum matching for it, $M_2 = \{(v_{3,4}, v_5)\}$ shown in Figure 2d. It then creates the graph G_3, as shown in Figure 2e, and finds a maximum matching for it, $M_3 = \{(v_{3,4,5}, v_6)\}$. Finally, MnM created the graph G_4, as shown in Figure 2f, and returns the partition $P = \{v_1, v_2\}, \{v_3, v_4, v_5, v_6\}$. We note that by the algorithm construction, a unified node $v_{i_1,...,i_l}$, is created by merging nodes v_{i_1} and v_{i_2}, and then by merging v_{i_1,i_2} and v_{i_3}, and so on.

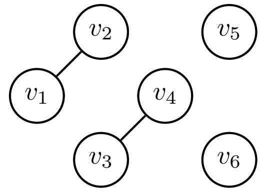
Algorithm 1: Match and Merge (MnM)

1 **Input**: A graph $G(V, E)$
  Result: A partition P of G.
2 $G_1(V_1, E_1) \leftarrow G(V, E)$
3 **for** $l \leftarrow 1$ *to* $k - 1$ **do**
4   $M_l \leftarrow$ maximum matching in G_l
5   $G_{l+1} = (V_{l+1}, E_{l+1}) \leftarrow$ an empty graph
6   $V_{l+1} \leftarrow V_l$
7   **for** every $(v_{i_1,...,i_l}, v_j) \in M_l$ **do**
8   Add vertex $v_{i_1,...,i_l,j}$ to V_{l+1}
9   remove $v_{i_1,...,i_l}, v_j$ from V_{l+1}
10   **for** every $v_{i_1,...,i_{l+1}} \in V_{l+1}$ **do**
11   **for** every $v_q \in V_{l+1}$ **do**
12   **if** $(v_{i_1,...,i_{l+1}}, v_q) \in E_l$ **then**
13   Add $(v_{i_1,...,i_{l+1}}, v_q)$ to E_{l+1}
14 $P \leftarrow$ an empty partition
15 **for** every $v_{i_1,...,i_j} \in G_k$ **do**
16   add the set $\{v_{i_1}, ..., v_{i_j}\}$ to P
17 **return** P

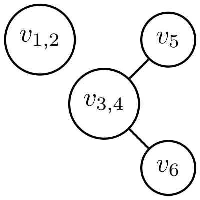
We now show that MnM provides an approximation ratio of $\frac{1}{k-1}$ for the social aware assignment problem. For that



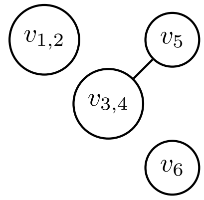
(a) The graph G_1



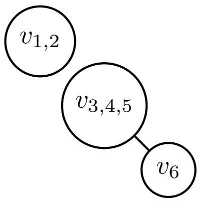
(b) The Matching M_1



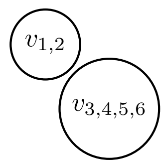
(c) The graph G_2



(d) The Matching M_2



(e) The graph G_3



(f) The graph G_4

Figure 2: An example for Algorithm 1 where $k = 4$.

end, we first prove the following lemma related to the possible edges in every G_l, $l > 1$.

Lemma 2. *Given $\hat{v} = v_{i_1,...,i_l} \in V_l$, if there exist $v_i, v_j \in V_l$, $v_i \neq v_j$ such that $(v_i, v_{i_n}), (v_j, v_{i_m}) \in E$ then $n = m$.*

Proof. Observe that for every $v_i, v_j \in V_l$ where $l > 1$, $(v_i, v_j) \notin E$, since M_1 is a maximum matching in G_1. Assume by contradiction and without loss of generality that $n < m$. If $n = 1$ and $m = 2$, then the path $v_i \to v_{i_n} \to v_{i_m} \to v_j$ is an M_1-augmenting path in G_1 ([Edmons, 1965]), contrary to the fact that M_1 is a maximum matching in G_1. Therefore, $m \geq 3$. Consider the graph G_2, since $v_{i_m}, v_j \in V_2$, there exists an edge $(v_j, v_{i_m}) \in E$, contrary to our observation. □

We now present a theoretical procedure (Procedure 2) that is provided with an optimal partition *Opt*, a graph G_l, and a corresponding round index l. Without loss of generality, we assume that every set $S \in Opt$ is a connected component. Let $O = \{v_o | \{v_o\} \in Opt$ and $v_o \in V_2\}$. That is, $|O|$ is the number of singletons in the partition *Opt* that are also not matched in M_1. In addition, let $V_l' = \{v_i | v_i \in V_l\}$, i.e., the set of all the single nodes in G_l. We show that Procedure 2 finds a matching with a size of at least $\frac{|V_l'|-|O|}{k-1}$. We further show that MnM is guaranteed to perform at least as well as this procedure, which, as we show, results in an approximation ratio of $\frac{1}{k-1}$ for every $k \geq 3$.

Lemma 3. *Procedure 2 finds a matching, R_l, in the graph G_l.*

Proof. At each iteration of the loop in line 7, we add an edge between a single node, v_q, and a unified node, $v_{i_1,...,i_l}$. We

---

Procedure 2: Find matching

1 **Input**:
2 The optimal partition Opt
3 A graph $G_l = (V_l, E_l)$
Result: A matching in G
4 $R_l \leftarrow$ an empty matching
5 **for** each $v_i \in V_l$ such that $\{v_i\} \in Opt$ **do**
6   remove v_i from V_l
7 **for** each $v_q \in V_l$ **do**
8   let $\hat{v}$ be a vertex $v_{i_1,...,i_l}$ such that $(v_q, \hat{v}) \in E_l$ and for some $1 \leq j \leq l$, v_q and v_{i_j} belong to the same set in Opt
9   **for** each $v_n \neq v_q$ **do**
10   **if** $(v_n, \hat{v}) \in E_l$ and exists $1 \leq m \leq l$, s.t. v_{i_m} and v_n belong to the same set in Opt **then**
11   remove v_n from V_l
12   add $(v_q, \hat{v})$ to R_l
13 **return** R_l

---

consider each single node only once. Therefore, it is not possible to add a single node twice to R_l. Similarly, each time a unified node is added to R_l, every single node $v_n \neq v_q$ such that v_{i_m} and v_n belong to the same set in *Opt*, for some $1 \leq m \leq l$, is removed from V_l. Therefore, a unified node is not added more than once. That is, R_l is a matching in G_l. □

Lemma 4. *Given an optimal partition Opt, procedure 2 finds a matching, R_l, in the graph G_l such that $|R_l| \geq (|V| - 2|M_1| - \sum_{i=2}^{l-1} |M_i| - |O|)/(k-1)$, where $l > 1$.*

Proof. In line 11 we remove nodes only when $m = j$ (according to Lemma 2). Given $\hat{v} = v_{i_1,...,i_l}$, there are at most $k - 1$ different nodes, $v_{j_1}, ..., v_{j_{k-1}}$ that are in the same set with $\hat{v}$ in *Opt*. Therefore, in each iteration of the loop in line 7, we remove at most $k - 2$ single nodes in line 11 while adding one edge to R_l in line 12. Thus, at least $\frac{1}{k-1}$ of the single nodes in v_l (who are not in O) are matched to a unified node. Therefore, $|R_l| \geq \frac{|V_l'|-|O|}{k-1}$. Now, $|V_2'| = |V_1| - 2|M_1|$. In addition, at each iteration $l > i > 1$, $|M_i|$ single nodes are each added to a unified node. Therefore, $|V_l'| = |V_1| - 2|M_1| - \sum_{i=2}^{l-1} |M_i|$. In addition, $V = V_1$. Overall, $|R_l| \geq (|V| - 2|M_1| - \sum_{i=2}^{l-1} |M_i| - |O|)/(k-1)$. □

Theorem 2. *Algorithm 1 provides a solution for the social aware assignment problem with an approximation ratio of $\frac{1}{k-1}$ for every $k \geq 3$.*

Proof. Since in *Opt* there are at least $|O|$ singletons, the value of *Opt* is at most $\frac{(|V|-|O|)\cdot(k-1)}{2}$, which occurs when all nodes are partitioned into cliques of size k (except those in

O). Clearly, $|P|\geq \sum_{i=1}^{k-1}|M_i|$. For every $l \geq 1$, $|M_l|$ is a maximum matching and thus $M_l \geq R_l$. In addition, according to Lemma 4, $R_l \geq \frac{|V|-|O|-2|M_1|-\sum_{i=2}^{l-1}|M_i|}{k-1}$.

Therefore,

$$|P|\geq \sum_{i=1}^{k-1}|M_i|= |M_1|+\sum_{i=2}^{k-1}|M_i|.$$

$$\sum_{i=2}^{k-1}|M_i|= |M_2|+|M_3|+... + |M_{k-1}|\geq$$

$$|M_2|+|M_3|+...+|M_{k-2}|+|R_{k-1}|\geq |M_2|+|M_3|+...+|M_{k-2}|+ \frac{|V|-|O|-2|M_1|-|M_2|-... - |M_{k-2}|}{k-1} =$$

$$\frac{|V|-|O|-2|M_1|}{k-1} + \frac{k-2}{k-1}\sum_{i=2}^{k-2}|M_i|\geq$$

$$(1 + \frac{k-2}{k-1}) \cdot \frac{|V|-|O|-2|M_1|}{k-1} + (\frac{k-2}{k-1})^2 \sum_{i=2}^{k-3}|M_i|\geq ... \geq$$

$$(1+\frac{k-2}{k-1}+(\frac{k-2}{k-1})^2+...+(\frac{k-2}{k-1})^{k-3})\cdot \frac{|V|-|O|-2|M_1|}{k-1}+$$

$$(\frac{k-2}{k-1})^{k-2} \sum_{i=2}^{k-1-(k-2)} |M_i|=$$

$$\sum_{i=0}^{k-3}\Big((\frac{k-2}{k-1})^i \cdot \frac{|V|-|O|-2|M_1|}{k-1}\Big)$$

That is,

$$|P|\geq |M_1|+\sum_{i=0}^{k-3}\Big((\frac{k-2}{k-1})^i \cdot \frac{|V|-|O|-2|M_1|}{k-1}\Big) =$$

$$|M_1|+\frac{|V|-|O|-2|M_1|}{k-1}\cdot \frac{(\frac{k-2}{k-1})^{(k-2)} - 1}{\frac{k-2}{k-1} - 1} =$$

$$|M_1|+(|V|-|O|-2|M_1|)\cdot \frac{(\frac{k-2}{k-1})^{(k-2)} - 1}{(k-1)(\frac{k-2}{k-1} - 1)} =$$

$$|M_1|-(|V|-|O|-2|M_1|)((\frac{k-2}{k-1})^{(k-2)} - 1) =$$

$$(|V|-|O|)(1 - (\frac{k-2}{k-1})^{(k-2)}) - |M_1|(1 - 2 \cdot (\frac{k-2}{k-1})^{(k-2)})$$

Next, we show that

$$(1 - 2 \cdot (\frac{k-2}{k-1})^{(k-2)}) \geq 0$$

Let

$$f(k) = (\frac{k-2}{k-1})^{k-2}, \text{ for } k \geq 3$$

Thus

$$f'(k) = \frac{(k-2)^{k-2}(ln(\frac{k-2}{k-1})(k-1) + 1)}{(k-1)^{k-1}}$$

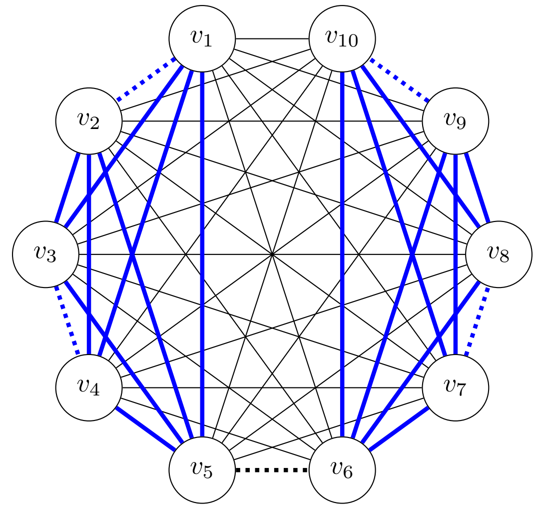


Figure 3: An example of a graph in which $k = 5$, and MnM achieves an approximation ratio of exactly $\frac{1}{4}$.

Now, $\frac{(k-2)^{k-2}}{(k-1)^{k-1}} > 0$.
In addition, it is known that $ln(x) \leq x - 1$ [Love, 1980], and thus

$$ln(\frac{k-2}{k-1})(k-1) + 1 \leq -\frac{1}{k-1}(k-1) + 1 = 0$$

Therefore, for all $k \geq 3$, $f'(k) \leq 0$ and $f(k) \leq f(3) = \frac{1}{2}$.

Overall, since $|M_1|\leq \frac{|V|-|O|}{2}$,

$$|P|\geq (|V|-|O|)(1-(\frac{k-2}{k-1})^{(k-2)})-\frac{|V|-|O|}{2}(1-2\cdot(\frac{k-2}{k-1})^{(k-2)})$$

$$= \frac{|V|-|O|}{2} \geq \frac{|Opt|}{k-1}$$

. □

Next, we show that our approximation ratio is tight.

Theorem 3. *The approximation ratio of MnM for the social aware assignment problem is tight.*

Proof. Given $k > 2$, consider a complete graph of size $2k$. In this case, MnM finds a perfect matching in M_1, and thus the partition P returned by MnM contains k groups of 2 nodes. That is, $V_P = k$. On the other hand, an optimal partition Opt consists of 2 Cliques of size k, and thus $V_{Opt} = 2\frac{k(k-1)}{2} = k(k-1)$. That is, MnM provides an approximation of exactly $\frac{1}{k-1}$. □

Figure 3 presents a case where $k = 5$, and G is a complete graph with 10 nodes. Here, $P = \{\{v_1, v_2\}, \{v_3, v_4\}, \{v_5, v_6\}, \{v_7, v_8\}, \{v_9, v_10\}\}$, as shown in the dotted lines, and thus $V_P = 5$. However, $Opt = \{\{v_1, v_2, v_3, v_4, v_5\}, \{v_6, v_7, v_8, v_9, v_10\}\}$, as shown in the blue lines, and $V_{Opt} = 20$.

However, we note that MnM can be improved such that the partition is maximal, that is, no two groups can join together and add additional edges. This can be achieved by first running MnM, and then adding back all the edges between any two unified nodes, as long as they may be merged together without violating the size capacity k. Multiple edges between two unified nodes, may either be simplified to a single edge, or may be associated with a weight proportionate to the number of edges. Now, MnM can continue the process of matching and merging for up to $\frac{k}{2}$ additional rounds, but in the merging process it also adds the edges between two previously unified nodes, as long as they may be merged together without violating the size capacity k.

6 Distributed Social Aware Assignment

In this section, we provide a procedure that attempts to model the behavior of the passengers in the social aware assignment problem when there is no central mechanism that determines the assignment. Assume that the users are split up arbitrarily but maximally, i.e., in a way that there are no additional connections that can be added. We call this procedure $Arbmax$. Without loss of generality, we assume that every set $S \in Arbmax$ is a connected component. We show that $Arbmax$ may provide an approximation ratio of $\frac{1}{k}$.

Theorem 4. *For any k, $Arbmax$ provides an approximation ratio of at most $\frac{1}{k}$*

Proof. Given k, consider the following graph G. There are k distinguished nodes, $v_1, \ldots, v_k$, with the edges $(v_i, v_{i+1}) \in E$ for $i = 1, \ldots, k-1$. Each distinguished node v_i has $k-1$ additional neighbors that are connected only to v_i, i.e., v_i is the internal node of a star graph with $k-1$ leaves. Clearly, Opt consists of k sets, where each set consists of a star graph. Thus, $V_{Opt} = k(k-1)$. On the other hand, $Arbmax$ may partition the graph such that the distinguished nodes $v_1, \ldots, v_k$ are in the same set. Since there are no edges between two undistinguished nodes, the value of the resulting partition is $k-1$. Therefore, $Arbmax$ provides an approximation ratio of at most $\frac{1}{k}$. □

Figure 4 presents a case where $k = 5$, and $Arbmax$ may provide an approximation ratio of $\frac{1}{5}$. Here, $Arbmax$ may return the partition $P' = \{\{v_1, v_2, v_3, v_4, v_5\}, \{v_6\}, \{v_7\}, \{v_8\}, \{v_9\}, \{v_{10}\}, \{v_{11}\}, \{v_{12}\}, \{v_{13}\}, \{v_{14}\}, \{v_{15}\}, \{v_{16}\}, \{v_{17}\}, \{v_{18}\}, \{v_{19}\}, \{v_{20}\}, \{v_{21}\}, \{v_{22}\}, \{v_{23}\}, \{v_{24}\}, \{v_{25}\}\}$ and thus $V_{P'} = 4$, while $Opt = \{\{v_1, v_6, v_7, v_8, v_9\}, \{v_2, v_{10}, v_{11}, v_{12}, v_{13}\}, \{v_3, v_{14}, v_{15}, v_{16}, v_{17}\}, \{v_4, v_{18}, v_{19}, v_{20}, v_{21}\}, \{v_5, v_{22}, v_{23}, v_{24}, v_{25}\}\}$ and thus $V_{Opt} = 20$.

7 Conclusions and Future Work

In this paper we argued that one way to promote ridesharing is by attempting to maximize the number of friendships in vehicles. For the that end we introduced the social aware assignment problem. In the social aware assignment problem there is a maximum fixed number of passengers in each vehicle, denoted by k, and the goal is to assign passengers such that the number of friendship relations is maximized. We showed that the problem is trivial for the case of $k = 2$, as it can be solved by a matching algorithm. However, we showed that the problem becomes computationally hard for any $k > 2$.

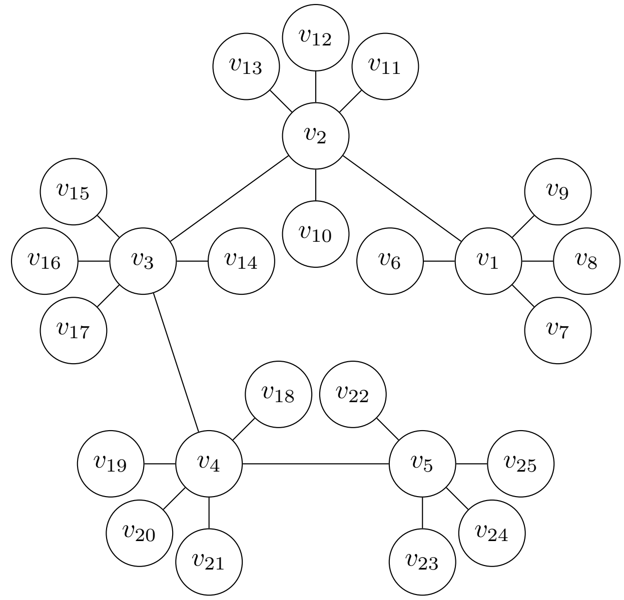


Figure 4

We therefore provided an efficient algorithm with an approximation ratio of $\frac{1}{k-1}$ and showed that this bound is tight. In addition, we analyzed the distributed case of this problem, in which the passengers split-up arbitrary but maximally, and showed that this procedure achieves an approximation ratio of, at most, $\frac{1}{k}$.

There are several interesting directions for future work. Because social aware assignment cannot be computed in polynomial time (unless $P = NP$), it will be interesting to investigate other variants of the problem. For example, we will consider assignments of users to vehicles such that each user will be matched with at least one friend in the same vehicle, while each vehicle is limited to a number of passengers, k. In this paper we have discussed a case where the capacity of each vehicle is limited and identical. However, in practice, there are vehicles of various types and sizes. Therefore, in future work we will examine a case where there are vehicles with different capacities. It will be interesting to see how this will affect the behavior and approximation quality of MnM, and whether we can develop an algorithm more suitable to this problem.

Another interesting research direction is to investigate the strategic aspects of the problem. That is, treating the passengers as strategic agents who don't necessarily accept their assigned vehicle and may attempt to join a different vehicle, if it is more valuable for them. We will investigate the existence and the complexity of calculation of partitions that are swap stable, envy-free, find which partitions are in the core, which are pareto-optimal, etc. [Aziz *et al.*, 2013] Furthermore, we intend to analyze the setting also with respect to the differences between drivers and passengers, i.e., some users may only be passengers, and it is not possible to group together only passengers (without a driver).

References

[Agatz *et al.*, 2012] Niels Agatz, Alan Erera, Martin Savelsbergh, and Xing Wang. Optimization for dynamic ride-sharing: A review. *European Journal of Operational Research*, 223(2):295–303, 2012.

[Aziz *et al.*, 2013] Haris Aziz, Felix Brandt, and Hans Georg Seedig. Computing desirable partitions in additively separable hedonic games. *Artificial Intelligence*, 195:316–334, 2013.

[Bogomolnaia *et al.*, 2002] Anna Bogomolnaia, Matthew O Jackson, et al. The stability of hedonic coalition structures. *Games and Economic Behavior*, 38(2):201–230, 2002.

[Chalkiadakis *et al.*, 2011] Georgios Chalkiadakis, Edith Elkind, and Michael Wooldridge. Computational aspects of cooperative game theory. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 5(6):1–168, 2011.

[Cordeau and Laporte, 2003] Jean-François Cordeau and Gilbert Laporte. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B: Methodological*, 37(6):579–594, 2003.

[Dantzig and Ramser, 1959] G. B. Dantzig and J. H. Ramser. The truck dispatching problem. *Management Science*, 6(1):80–91, 1959.

[di Complessita and Rizzi, 2004] Corso di Complessita and Romeo Rizzi. Np-complete problem: Partition into triangles. Technical report, Technical Report, 2004.

[Edmons, 1965] E Edmons. Paths, trees, and flowers. *Canad. J. Math*, 17:449–467, 1965.

[Flammini *et al.*, 2021] Michele Flammini, Gianpiero Monaco, Luca Moscardelli, Mordechai Shalom, and Shmuel Zaks. On the online coalition structure generation problem. *Journal of Artificial Intelligence Research*, 72:1215–1250, 2021.

[Guidotti *et al.*, 2015] Riccardo Guidotti, Andrea Sassi, Michele Berlingerio, Alessandra Pascale, and Bissan Ghaddar. Social or green? a data-driven approach for more enjoyable carpooling. In *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*, pages 842–847. IEEE, 2015.

[Love, 1980] E. R. Love. 64.4 some logarithm inequalities. *The Mathematical Gazette*, 64(427):55–57, 1980.

[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.

[Montazery and Wilson, 2016] Mojtaba Montazery and Nic Wilson. Learning user preferences in matching for ridesharing. In *ICAART (2)*, pages 63–73, 2016.

[Parragh *et al.*, 2008a] Sophie N. Parragh, Karl F. Doerner, and Richard F. Hartl. A survey on pickup and delivery problems. part I: Transportation between customers and depot. *Journal für Betriebswirtschaft*, 58(1):21–51, 2008.

[Parragh *et al.*, 2008b] Sophie N. Parragh, Karl F. Doerner, and Richard F. Hartl. A survey on pickup and delivery problems. part II: Transportation between pickup and delivery locations. *Journal für Betriebswirtschaft*, 58(1):81–117, 2008.

[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.

[Schleibaum *et al.*, 2020] Sören Schleibaum, Maike Greve, Tim-Benjamin Lembcke, Amos Azaria, Jelena Fiosina, Noam Hazon, Lutz M Kolbe, Sarit Kraus, Jörg P Müller, and Mark Vollrath. How did you like this ride? an analysis of user preferences in ridesharing assignments. In *VEHITS*, pages 157–168, 2020.

[Sless *et al.*, 2018] Liat Sless, Noam Hazon, Sarit Kraus, and Michael Wooldridge. Forming k coalitions and facilitating relationships in social networks. *Artificial Intelligence*, 259:217–245, 2018.

[Wright and Vorobeychik, 2015] Mason Wright and Yevgeniy Vorobeychik. Mechanism design for team formation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(1):1050–1056, Feb. 2015.