

The Merits of Sharing a Ride

Pooyan Ehsani¹

Jia Yuan Yu¹

Abstract—The culture of sharing instead of ownership is sharply increasing in individuals behaviors. Particularly in transportation, concepts of sharing a ride in either carpooling or ridesharing have been recently adopted. An efficient optimization approach to match passengers in real-time is the core of any ridesharing system. In this paper, we model ridesharing as an online matching problem on general graphs such that passengers do not drive private cars and use shared taxis. We propose an optimization algorithm to solve it. The outlined algorithm calculates the optimal waiting time when a passenger arrives. This leads to a matching with minimal overall overheads while maximizing the number of partnerships. To evaluate the behavior of our algorithm, we used NYC taxi real-life data set. Results represent a substantial reduction in overall overheads.

I. INTRODUCTION

Currently, cars are vastly underutilized. Firstly, most cars sit in parking spaces most of the time. Secondly, cars under-used due to their low occupancy rate relative to the number of seats available (1.67 persons per vehicle in the U.S. [Di Gianni, 2015]). This inefficient use of transportation and parking resources had to be tolerated to enjoy reliable, safe, and on-demand transportation. Today, however, technology allows us to move away from the car-ownership paradigm toward a shared transportation one [Gargiulo et al., 2015], [Saranow, 2006]. Individuals who have similar itineraries and time schedules use a shared vehicle in any portion of their paths in a ridesharing scheme. An efficient use of empty seats of cars in urban trips has numerous economic and societal outcomes for individuals and the general public. Increasing warnings about high consumption rates of finite sources of fossil fuels and global warming, getting stuck at traffic congestions [Xiang et al., 2008] and rising gas prices are few examples demanding wiser adoption of vehicles. Additionally, Sharing a car enables passengers to share travel expenses which could be a substantial source of motivation. It should be noted that ridesharing is different from conventional carpooling and vanpooling systems dedicated to trips on a regular basis [Levofsky and Greenberg, 2001] which are very limited with respect to scheduling.

Taxi service providers can benefit from the advantages of ridesharing as well. These companies may lose some of the passengers due to their limited fleet size compared to the requests in metropolitan areas. Relatively high costs of taking taxis in comparison with other public transportation means

is a deterrent factor of extensive usage of taxis. Having more than one passenger in a taxi could be a key factor in solving issues above since the occupancy rate of taxis increase, and every passenger pays marginally lower expenses.

In this paper, we focus on the case when a central coordinator is owning a finite fleet of cars offering an automated matching of passengers within an urban area. Optimal time windows and other constraints on accepted detours for every passenger need to be respected while overall costs minimized. Each active individual has a specific itinerary and has to make a detour and extra stops to accommodate ridesharing partner. The length of the individual optimal waiting time depends on the passenger's willingness to deviate from the shortest path to get to his destination.

In this paper, we design and implement an algorithm to maximize the matching of passengers (based on the matching algorithms on general graphs [Witzgall and Zahn, 1965]) while optimizing the costs incurred for a taxi service provider. We evaluate the effectiveness of the proposed algorithm using real data set.

We can formulate this problem in the sense of a cooperative agent based system [Xing et al., 2009]. In this case, neither of tasks can be carried out by individual agents in isolation. Thus, problem solving necessitates agents' interaction. User agents inquire an appropriate number of service agents to perform the matching task. Service agents maintain and has access to active passengers' information including origin, destination, and flexibility. To be precise, matching agents find a compatible match for a passenger willing to leave.

The remainder of this paper is organized as follows. In section II, formulates the model to solve online ridesharing problem and presents our proposed algorithm. Experimental results are discussed in Section III. In Section IV, we review related work and represent how our approach is different from others. Finally, we summarize conclusions and describe future works in Section V.

II. PROBLEM DEFINITION

We model the ridesharing problem as an online single decision maker problem. The road network modeled as a weighted graph, with vertices representing locations, and weighted edges representing the distance between locations. Passengers appear at random time instants, with random source and destination and we fix a flexibility factor which models the overhead in travel time. We assume that a central decision maker keeps track of this information for each passenger in real-time, and is tasked with matching

P. Ehsani and J. Y. Yu, are with the Concordia Institute for Information Systems Engineering, Concordia University Montreal, Canada. Email: p_ehsan@encs.concordia.ca, jiayuan.yu@concordia.ca

passengers in an online fashion. For simplicity, we first assume that each passenger is either matched to one other passenger or travels alone. Moreover, once a trip started, matches are no longer allowed. In this model, the decision is made of two components: how long to make each passenger wait, and which another one (if any) to match to when the waiting time has elapsed. Once passengers are matched, we assume that a nearby taxi is dispatched to fulfill the trip. A number of objectives can be considered: from a social welfare perspective. We want to minimize for each passenger the expected distance traveled in single occupancy, subject to the flexibility constraint. From the perspective of an operator like UberPool, the goal is to maximize the expected profit for each passenger. We propose to estimate the distribution of the random arrival time, source and destination from openly available data. Given this distribution and travel times on the road network, we propose an online algorithm that first optimizes the waiting time for each newly appeared passenger, and then matches a passenger whose waiting time has elapsed by using the minimum cost perfect matching algorithm. To model this problem, we present a set of passengers $P = \{p_1, \dots, p_N\}$ where $p_i = (s_i, d_i)$, $i \in \{1, \dots, N\}$. Each passenger has a starting point s_i and a destination point d_i . The central ride sharing system assigns either one or two passenger in a single car. In the former, the taxi picks up passenger p_i and moves along the shortest path from his starting point s_i to his desired destination d_i . Conversely, if a matching occurs, the taxi picks up passenger i , travels to lift passenger j and drops them off in their respective destinations. The taxi always opts for the shortest path available between two points. We assume that starting points and destinations are i.i.d. random variables and the distributions are known. In addition, partial sums of exponential random variables e_k , $k \in \{1, \dots, N\}$ represent time elapsed between the appearance of consecutive passengers. t_i denotes passenger p_i 's time of arrival.

$$t_i = \sum_{k=1}^i e_k. \quad (1)$$

Assumption 2.1: We assume all random variables are generated i.i.d. and that distribution functions are known.

Let $G = (V, E)$ be a q by q weighted grid representing a road network. This graph illustrates starting points and destinations of passengers currently in the system. Let π_i denote the shortest path travel time for passenger i to move from his starting point to his desired destination. We assume that each passenger p_i is willing to incur at most ϵ fraction above the shortest path. Consequently, the set of allowable paths for every arrived p_i is denoted:

$$\Pi_i \triangleq \{\pi : |\pi| \leq (1 + \epsilon_i)|\pi_i|\} \quad (2)$$

The set Π_i contains every allowable path from s_i to d_i . In this paper, we design an algorithm to find the optimal waiting time for every passenger p_i arriving and then decide on the best matching to optimize the system-wide overall overhead. The optimum waiting time for passenger p_i before

moving on is a function of s_i, d_i , and ϵ_i which is indicated by $\tau_i(s_i, d_i, \epsilon_i)$. For simplification, we denote this by τ_i . We assume that cars always show up by the end of τ_i and no passenger waits longer because a taxi arrives late. Different values of τ_i may yield more compatible matches and consequently lower overheads. A passenger p_i spends a longer time before reaching his destination as a result of participating in a ride sharing arrangement. By denoting actual travel time as ω_i the following inequality must hold for any passenger entering the system.

$$\tau_i + \omega_i \leq (1 + \epsilon_i)\pi_i \quad (3)$$

As we increase the waiting time, more passengers appear in the pool of waiting passengers; however, as the waiting time for a given passenger increases, the tolerable detour distance for that passenger decreases, such that fewer matches within the pool are feasible due to the flexibility constraint. We define set W_t including all passengers who are still in the system at any instance t . This set excludes matched up passengers who have left the system before t .

$$W_t = \{p_i \mid t_i \leq t < t_i + \tau_i\}, \quad i \in \{1, \dots, N\} \quad (4)$$

During τ_i , some passengers may enter the system, and some of them proper for share a ride with p_i based on their source and destination. These nominated passengers add to set $Q_i \subset W$ which is initially null for every passenger p_i . By the end of τ_i , the central ride sharing system decides to assign p_i to the best match to optimize system-wide overhead.

Definition 2.1: (Compatible Matching). A match between p_i and p_j are compatible if there exist two paths $\pi_i \in \Pi_i$ and $\pi_j \in \Pi_j$ which:

$$\begin{aligned} & d_i, s_j \in V(\pi_i \cap \pi_j) \text{ or } s_j, d_j \in V(\pi_i \cap \pi_j) \\ & \text{or } s_i, d_i \in V(\pi_i \cap \pi_j) \text{ or } d_j, s_i \in V(\pi_i \cap \pi_j) \end{aligned} \quad (5)$$

Definition 2.2: (Order of sharing). Define $O(p_i, p_j)$ as the cost associated with the order of pick up and drop off of the passengers such that the first element in the pair represents the first passenger picked up and the second element denotes the first passenger dropped off in his desired destination. It should be noted that if p_i does not share a ride, then the cost is simply the cost of going alone on the shortest path, denoted by $O(p_i, 0)$.

Definition 2.3: (Cost of a matched pair). Define the travel cost for matched up passengers p_i and p_j , $j \in Q$:

$$C(p_i, p_j) = \begin{cases} \frac{1}{2} \min_{m, n \in \{i, j\}} \{O(p_m, p_n)\} & \text{if } i \neq j \\ O(p_i, 0) & \text{if } i = j \end{cases} \quad (6)$$

It should be noted that the cost is proportional to the distance covered by vehicles like fuel cost. We assume that total overhead is comprised of two parts, total waiting overhead and total travel overhead. As described before, an increase in waiting time has a limiting effect on feasible ride sharing partners for passenger i ; therefore, influences the probability of finding a match. The optimal waiting time

(in the expected sense) for every passenger i is as follows:

$$\tau_i = \arg \min_{u \in [1, \infty)} \Psi_i(u) + \Gamma_i(u) \quad (7)$$

In which $\Psi_i(u)$ expectation shows the case which $p_j = \{s_j, d_j\} \in V \times V$ is not matched to passenger p_i and is defined as follows:

$$\begin{aligned} \Psi_i &\triangleq \mathbb{E} \left\{ \omega_i \mid \forall p_j \in V \times V : \left(p_i \text{ and } p_j \text{ are not compatible} \right) \right\} \\ &= \sum_{p_j \in V \times V} \left\{ \mathbb{P} \left(d_i, s_j \notin \pi_i \cap \pi_j \text{ and } s_j, d_j \notin \pi_i \cap \pi_j \right. \right. \\ &\quad \left. \left. \text{and } s_i, d_i \notin \pi_i \cap \pi_j \text{ and } d_j, s_i \notin \pi_i \cap \pi_j \right) \right. \\ &\quad \left. C(p_i, p_i) \right. \\ &\quad \left. + \mathbb{P} \left(d_i, s_j \in \pi_i \cap \pi_j \text{ or } s_j, d_j \in \pi_i \cap \pi_j \right. \right. \\ &\quad \left. \left. \text{or } s_i, d_i \in \pi_i \cap \pi_j \text{ or } d_j, s_i \in \pi_i \cap \pi_j \right) \right. \\ &\quad \left. \left(1 - \mathbb{P}(t_i \leq t_j \leq t_i + u) \right) C(p_i, p_i) \right\} \quad (8) \end{aligned}$$

On the other hand, Γ_i represents a case where a matching can occur between p_i and $p_j = \{s_j, d_j\} \in V \times V$. The expected value is shown in:

$$\begin{aligned} \Gamma_i &\triangleq \mathbb{E} \left\{ \omega_i(Q_i) \mid \forall p_j \in V \times V : \left(p_i \text{ and } p_j \text{ are compatible} \right) \right\} \\ &= \sum_{p_j \in V \times V} \left\{ \mathbb{P} \left(d_i, s_j \in \pi_i \cap \pi_j \text{ or } s_j, d_j \in \pi_i \cap \pi_j \right. \right. \\ &\quad \left. \left. \text{or } s_i, d_i \in \pi_i \cap \pi_j \text{ or } d_j, s_i \in \pi_i \cap \pi_j \right) \right. \\ &\quad \left. \mathbb{P}(t_i \leq t_j \leq t_i + u) C(p_i, p_j) \right\} \quad (9) \end{aligned}$$

We assumed passengers arrival time are independent and identically distributed exponential random variables, $X \sim \exp(\lambda)$ and $Y \sim \exp(\lambda)$. We name the random variable $Z = X - Y$ and we know the CDF: $\mathbb{P}(X - Y \leq u) = 1 - \frac{e^{-\lambda u}}{2}$. By knowing the distribution of random variables s_j and d_j , the probabilities in the above equation are determined. We consider a case where the travel cost is divided evenly between pairs in any arrangement. Equation 7 prevent the algorithm to assign long waiting time to passenger as the waiting time becomes longer, because of equation 3 must be held for passenger, the number of incompatible passengers increased, and it causes an increase in $\Psi_i(u)$.

Definition 2.4: (Leaving candidates). Define a passenger leaving the system with the minimum amount the time of arrival and the optimum waiting time among all the current passengers.

$$k^* = \arg \min_{\{i \mid p_i \in W\}} \{t_i + \tau_i\} \text{ and } \theta = \min_{\{i \mid p_i \in W\}} \{t_i + \tau_i\} \quad (10)$$

The number k^* indicates the nominated passenger is leaving the system by the end of his optimal waiting time. θ denotes the latest departure time for the nominated passenger..

A. Proposed Algorithm

In this section, we describe how our algorithm works to find the best matching over all passengers.

Algorithm 1: Finding the ride sharing matching

Input:

- A graph G and a positive number N
- A set of pairs $P = \{p_1, \dots, p_N\}$, where $p_i = (s_i, d_i)$ and $\forall i \in \{1, \dots, N\} : s_i, d_i \in V$
- Pairs of $O(p_i, p_j)$ which is the cost associated with the order of pick up and drop off of passengers
- $i, j \in \{1, \dots, N\}$ - Distribution functions of $s_i, d_i, \epsilon_i, e_i$
- A cost function $C : P^2 \rightarrow \mathbb{Z} \cup \{0\}$

Output:

- The optimal waiting time, i.e. value of τ_i for every passenger p_i
- A matching $M \subseteq P^2$ such that $\forall i \in \{1, \dots, N\}$ there is precisely one pair in M such that p_i appears in that pair (note that c_i could be in both coordinates of the pair, in which case a passenger is paired with itself).

Initialization:

Create set of the current passengers such that

$$W_t = \{p_1\}$$

Evaluate optimal waiting time for the first passenger, τ_1 as in equation 7.

Update the potential leaving candidate information, k^* , and θ as in equation 10

$j \leftarrow 2$

while $|M| < N$ **do**

while $\theta \geq t_j$ **do**

- Add the arrived passenger to the set of current ones in the system, $W_\theta \leftarrow W_\theta + \{p_j\}$
- Evaluate the optimal waiting time for the newly arrived passenger, τ_j
- Update the leaving candidates and put it in set k^*
- Update the time of leaving for the candidate and tag it as θ
- $j \leftarrow j + 1$

- Create a weighted graph T with $|W_\theta|$ nodes and for each candidate add edge to compatible passengers in W_θ with a weight equal to the cost function C .
 - Use Blossom algorithm to find minimum cost perfect matching in T . Remove matched up passengers from W_θ and add them as a pair to set M .
 - Update the information about leaving candidate, k^* and θ
-

III. EXPERIMENTAL RESULTS

To test the behavior of our algorithm, we used the data provided by NYC Taxi & Limousine Commission [NYC Taxi & Limousine Commission, 2016] for yellow cabs in January 2016. This dataset contains more than 77 million trips information. Thus, we extracted 10,000 trips and used Kernel Density Estimation [Wand and Jones, 1993] to estimate the probability distribution function of starting points and destinations of all passengers. Note that the database size

of 10,000 entries is a practical case and closely resembles the current daily amount of matches in a city. To extend the data set to our approach of arbitrary origin and destination locations, we applied the estimated PDF to estimate the likelihood of appearance of pickup and drop-off points of all passengers. For each starting point which we picked up randomly from the region with the estimated probability, a destination point is located in the same way. Figure 1 shows the likelihood function behavior for three different passengers in a short period which passengers can wait for a match. The function evaluated as discussed before in equation 7. Figure 2 compares the original nodes locations

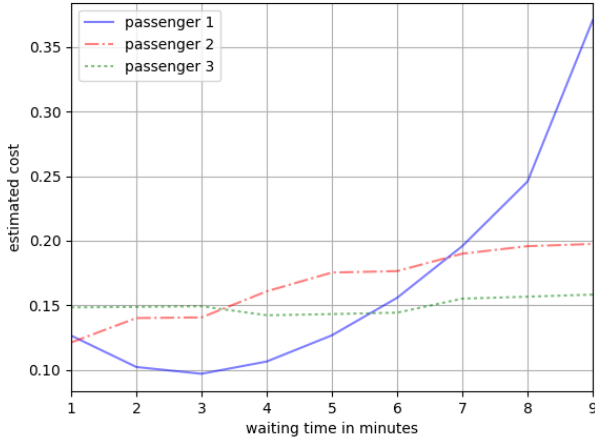


Fig. 1 : Estimated $\Psi_i(u) + \Gamma_i(u)$

on the left to the right showing a population density plot of NYC to support the validity of the perturbation.

Our approach for estimating the waiting time applied for a set of 1000 passengers in March 2016 and as shown in figure 3 for most of the passengers the waiting time is less than 2 minutes which is reasonable. Experiments on 16 datasets and each of them with the 1000 passengers from April shows that our algorithm has a significant reduction in costs using ridesharing. This plot shows that using ridesharing could result in a decrease in costs for a taxi provider company and its could motivate passengers to use their service more frequently. In 4 we compare cost reduction via our approach with a greedy offline algorithm which is for the greedy offline model, passengers arrival time are given and used to find the best waiting time from 7. Blossom algorithm is used to find the proper matching of the passengers in the offline algorithm. For calculating the reduction in cost we define total travel distance of passengers as $\sum_{i=1}^n |\pi_i|$ where n is the number of passengers in the dataset, and the total travel distance after finding the optimal matching as M by algorithm 1 as $\sum_{e_i \in M} w_{e_i}$ and calculate the reduction in cost by $1 - \frac{\sum_{e_i \in M} w_{e_i}}{\sum_{i=1}^n |\pi_i|}$. In figure 5 we compare the reduction in costs on 16 test sets based on the waiting time of the passengers. As the waiting time increase for a specific passenger in figure 6 we increase the ϵ from zero to one in

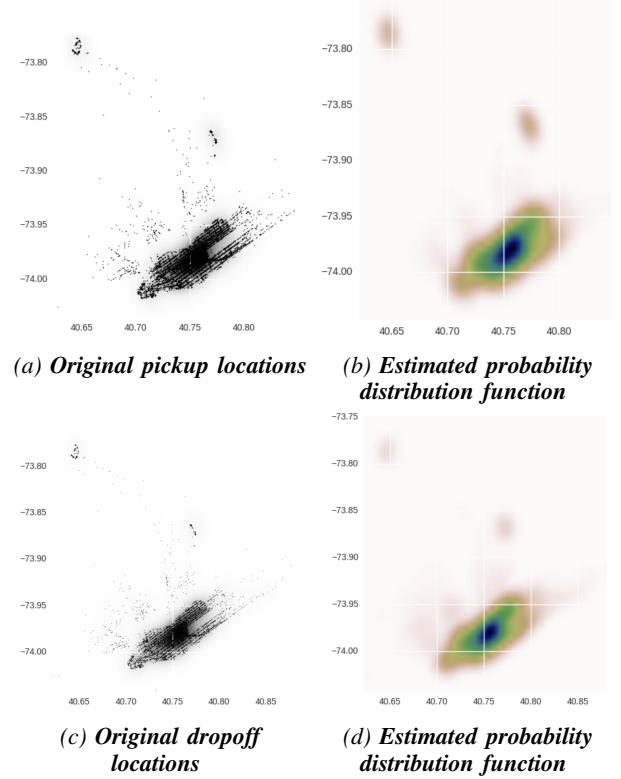


Fig. 2 : Original nodes and estimated PDFs

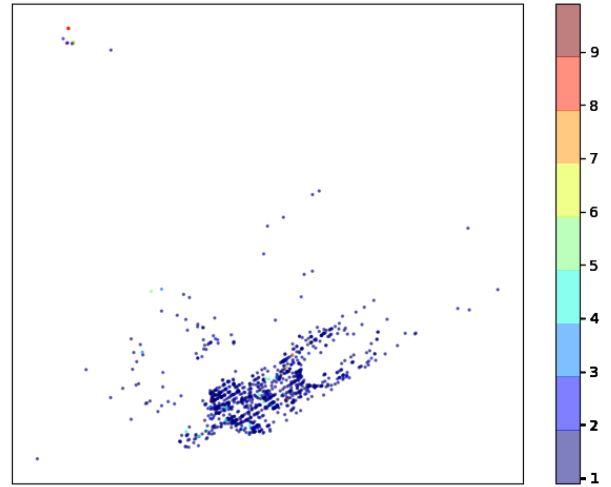


Fig. 3 : Waiting time for passengers in minutes

order to see how ϵ changes the results. We choose $\epsilon = 0.6$ based on experimental results. By setting $\epsilon = 0.6$ the average waiting time for passengers based on 16 test sets equal to 1 minutes and 31 seconds and the average travel time increased by 3 minutes and 23 seconds.

In figure 7 constant waiting time assigned to all the passengers to compare our method for estimating the best waiting time with constant waiting time. Our results are close to two minutes constant waiting time and we achieve similar results with 1 minutes and 31 seconds as the average of our waiting time.

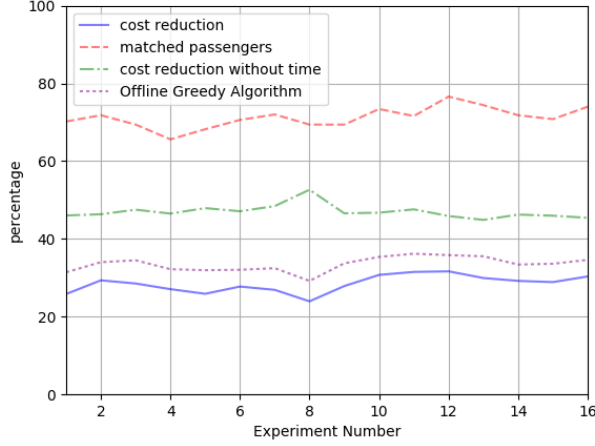


Fig. 4 : The reduction in costs using ridesharing

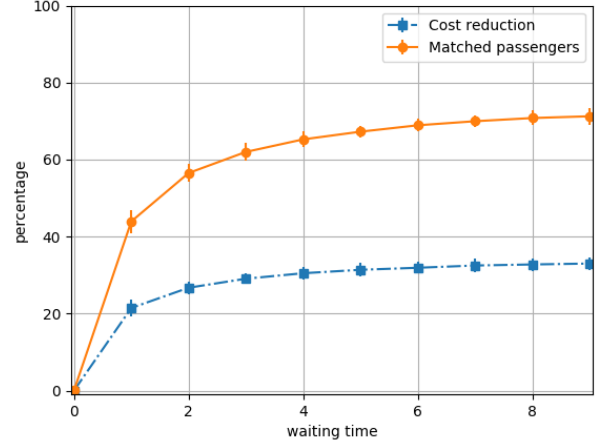


Fig. 7 : Constant waiting time

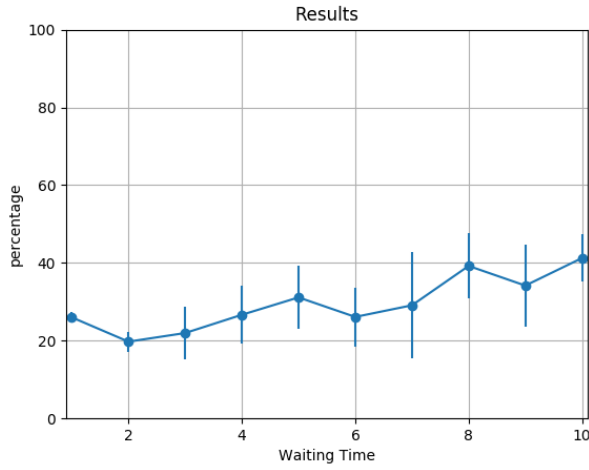


Fig. 5 : The relation between waiting time and cost reduction

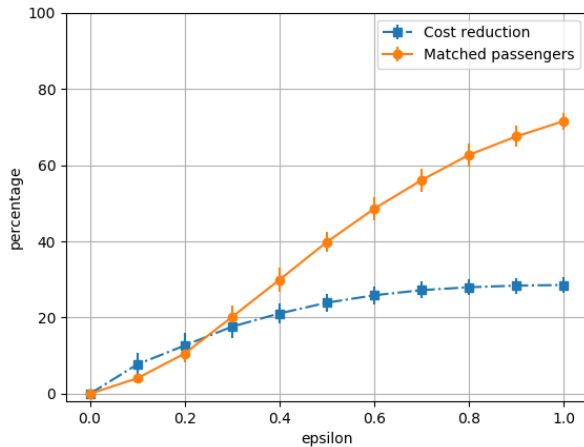


Fig. 6 : The relation between ϵ and cost reduction

IV. RELATED WORK

Online ridesharing defines a scheme in which passengers arrive into the system as time goes on. At each time instance, only the information related to active passengers are known. Surveys of online ridesharing problem can be found in [Agatz et al., 2011], [Agatz et al., 2012] presenting optimization challenges and [Furuhata et al., 2013] addressing a classification of the current ridesharing systems.

Online ridesharing is in contrast with the case when entire data is available at the very beginning [Bernstein and Rajagopalan, 1993]. In the traditional systems passengers have fixed schedules, starting points, and destinations and basically, all the passenger's information is available at any time instance [Michalak et al., 1994]. Our problem investigates practical situations in the real world, and it is more challenging, as information emerges continuously and passengers must be matched optimally in a short notice. The automated ridesharing operator matches up potential passengers in a short notice.

This problem is very similar to pick up and delivery problems described in [Berbeglia et al., 2010]. It can be referred to like a particular case of dial-a-ride problem [Attanasio et al., 2004]. DARP mainly revolves around routing vehicles subject to time constraints on pickup and delivery. The majority of research in this category is carried out on static DARP where all the information is ready before initiating the system. These portions use two-phase scheduling strategy [Attanasio et al., 2004], [Horn, 2002] to route the taxi fleet. This is distinguished from our problem as we focus on passengers but in DARP scheduling taxis are considered as the main concerns. Moreover, here we aim to maximize matching of passengers while optimizing overall overheads, an approach that taxi-sharing problem ignores.

Li et al. [Li et al., 2014] considered a problem in which a shared vehicle delivers a passenger and a parcel to destinations while persons have priority over parcels. They avoid serving two passengers simultaneously in a taxi. They present an exact MILP formulation to solve this problem

in tiny instances to maximize taxi provider's profit in a deterministic configuration. They proposed a solution based on large neighborhood search heuristic for a case in which travel times and delivery locations are stochastic [Li et al., 2016].

Ma et al. [Ma et al., 2013] defined an algorithm to dispatch the best vehicle in a taxi sharing scheme when passengers send real-time queries. They provided a searching algorithm to schedule taxis to satisfy queries incurring minimum additional travel time. They showed using shared rides, it is possible to increase vehicle occupancy rate (25% in their paper) having a relatively small increase in travel distances (only 13%). Our problem differs from this research as beyond having real-time requests from passengers, we consider maximizing ridesharing partners and minimizing overheads. Taxi sharing problem (i.e., DARP, SARP) usually discussed regarding a scheduling problem, rather than a matching problem. We propose our problem in the sense of online roommate problem and online matching problem on general graphs [Bernstein and Rajagopalan, 1993].

Tao [Tao, 2007] addressed the problem where a set of passengers travel from different origins to a fixed destination on a regular basis and then dispatched from this fixed place to their origins in the reverse paths. The matching is done in real-time. A call center matches the closest taxi to the predetermined matching list of passengers. We also emphasize on the role of central coordinator to match passengers in a way the total cost of a fleet is minimized. However, passengers and matching sets are at the center of focus, not routing vehicles. Also, in our paper, passengers have distinguished starting points, and destinations and vehicle serve at most two passengers simultaneously.

[Herbawi and Weber, 2012] presented a genetic algorithm to solve the multi-criteria optimization model of minimizing total distance covered by drivers and incurred travel time of passengers while maximizing the size of matching. Detours of drivers and passengers time limitations are predetermined. Conversely, we calculate the optimal waiting time for every passenger such that resulting matching leads to optimal overall overhead. Additionally, we assume that passengers do not use a private vehicle to travel.

Agatz et al. [Agatz et al., 2011] designed optimization approaches to solve real-time matching in an online ridesharing scheme. They showed that using sophisticated optimization techniques benefits the performance of ridesharing systems. However, they opt for the case where passengers use their own vehicles to move. In our paper, we consider a limited fleet of independent taxis controlled by a central coordinator. In our paper, when the passengers start their trip they leave the system and will not participate in matching. But in [Ghoseiri et al., 2011] riders can be matched while en-route.

Notwithstanding the fact that many ridesharing projects have been initiated in the past few years, many of them have been proved to be unsuccessful in practice [Ghoseiri et al., 2011]. A ridesharing system must be flexible, efficient and economical to be substituted with the private cars having the advantage of immediate door-to-door access [Agatz et al.,

2011]. Our model provides this features to passengers.

While the role of technological advances like GIS and communication networks and the potential of ridesharing are vastly studied [Sarraino et al., 2008], [Amey, 2011], developing sophisticated optimization and matching algorithms to arrange ridesharing partnerships in real-time has been largely neglected. In this paper, we mainly focus on designing an algorithm to maximize the rate of participation in ridesharing (matching passengers) while minimizing system-wide overheads.

A mixed integer optimization is developed by [Ghoseiri et al., 2011] to maximize the matching of partners while considering individual preferences. In our paper, minimizing costs is a vital part of the objective function. waiting time is a predetermined factor manually selected by riders, but in our paper, it is computed optimally aligned with our perspective.

Some people may opt for the ride sharing merely for cost savings. Individuals not owning private cars have limited access to public transportation options can be other targets of ridesharing. People are likely to participate in ridesharing arrangements if they pay less fare in comparison with taking a taxi alone. Also, they have to accept a deviation from their shortest path too. There are different methods to share the travel-expenses among entities. The most straightforward intuitive way is to divide the costs evenly between passengers [Geisberger et al., 2009]. This approach would not be fair as the portion of the shared path could be different. To overcome this, passengers may divide the related costs proportional to the distances they ride autonomously and the part they shared the car.

Among the topics investigated in this field, most studies capture passengers time preference by denoting an accepted time window. Agatz et. al [Agatz et al., 2011] considers an acceptable period between earliest departure time and latest arrival time to the destination defined by every passenger. Some papers limit the actual travel time by defining a fixed threshold deviation in traveling time [Amey, 2011] and a maximum ride time [Baldacci et al., 2004]. In contrast, we propose a strategy where the central coordinator computes and declares an optimal waiting time by the advent of a passenger leading to optimal overall cost. A passenger's of tendency to move along longer paths to get to his desired destination is the first element to compute his optimal waiting time.

[Xiang et al., 2008] introduced a ridesharing scheme considering a decentralized agent based system. In this paper, passenger agents try to find a partner every two minutes among driver agents. They delineated that given a high number of drivers, travel time for a single passenger decrease. We face a problem where passengers use taxis to commute not their own vehicles. Additionally, there is no emphasize on optimizing in [Xiang et al., 2008] research.

V. CONCLUSION AND OUTLOOK

In this work, we have modeled a particular online ridesharing problem as a matching problem and provided an algorithm to solve it optimally. The case where passengers only

use taxis and no one drives his private car is considered in this paper. In addition, we have engineered NYC Taxis & Limousine Commission dataset for testing the performance of our algorithm and predict real-time data. By using our algorithm, we have shown that maximized participation in ridesharing arrangements decreases the total distance covered by taxis and consequently minimizes total costs in comparison with the case all passengers take taxis autonomously. This is in favor of the central taxi provider since this scheme decreases the likelihood of losing requests as a result of lack of sufficient number of fleet. There are many directions to extend this work. One is to implement other matching and optimization techniques to solve the same problem. Another direction is to extend the capabilities the algorithm to use the full capacity of the taxis by adjoining more passengers. One interesting research venue is to make the algorithm more flexible by enabling en route matchings. These initiatives can lead to more realistic scenarios.

REFERENCES

- [Agatz et al., 2012] Agatz, N., Erera, A., Savelsbergh, M., and Wang, X. (2012). Optimization for dynamic ride-sharing: A review. *European Journal of Operational Research*, 223(2):295–303.
- [Agatz et al., 2011] Agatz, N. A., Erera, A. L., Savelsbergh, M. W., and Wang, X. (2011). Dynamic ride-sharing: A simulation study in metro atlanta. *Transportation Research Part B: Methodological*, 45(9):1450–1464.
- [Amey, 2011] Amey, A. (2011). A proposed methodology for estimating rideshare viability within an organization, applied to the mit community. In *TRB Annual Meeting Proceedings*, pages 1–16.
- [Attanasio et al., 2004] Attanasio, A., Cordeau, J.-F., Ghiani, G., and Laporte, G. (2004). Parallel tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem. *Parallel Computing*, 30(3):377–387.
- [Baldacci et al., 2004] Baldacci, R., Maniezzo, V., and Mingozzi, A. (2004). An exact method for the car pooling problem based on lagrangean column generation. *Operations Research*, 52(3):422–439.
- [Berbeglia et al., 2010] Berbeglia, G., Cordeau, J.-F., and Laporte, G. (2010). Dynamic pickup and delivery problems. *European journal of operational research*, 202(1):8–15.
- [Bernstein and Rajagopalan, 1993] Bernstein, E. and Rajagopalan, S. (1993). The roommates problem: Online matching on general graphs. Technical report, Technical Report CSD-93-757, University of California, Berkeley, CS Department.
- [Di Gianni, 2015] Di Gianni, J. J. (2015). *Exploration of the current state and directions of dynamic ridesharing*. PhD thesis, Montclair State University.
- [Furuhata et al., 2013] Furuhata, M., Dessouky, M., Ordóñez, F., Brunet, M.-E., Wang, X., and Koenig, S. (2013). Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological*, 57:28–46.
- [Gargiulo et al., 2015] Gargiulo, E., Giannantonio, R., Guercio, E., Borean, C., and Zenezini, G. (2015). Dynamic ride sharing service: are users ready to adopt it? *Procedia Manufacturing*, 3:777–784.
- [Geisberger et al., 2009] Geisberger, R., Luxen, D., Neubauer, S., Sanders, P., and Volker, L. (2009). Fast detour computation for ride sharing. *arXiv preprint arXiv:0907.5269*.
- [Ghoseiri et al., 2011] Ghoseiri, K., Haghani, A. E., Hamed, M., and Center, M. (2011). *Real-time rideshare matching problem*. Mid-Atlantic Universities Transportation Center.
- [Herbawi and Weber, 2012] Herbawi, W. and Weber, M. (2012). The ridematching problem with time windows in dynamic ridesharing: a model and a genetic algorithm. In *2012 IEEE Congress on Evolutionary Computation*, pages 1–8.
- [Horn, 2002] Horn, M. E. (2002). Fleet scheduling and dispatching for demand-responsive passenger services. *Transportation Research Part C: Emerging Technologies*, 10(1):35–63.
- [Levofsky and Greenberg, 2001] Levofsky, A. and Greenberg, A. (2001). Organized dynamic ride sharing: The potential environmental benefits and the opportunity for advancing the concept. In *Transportation Research Board 2001 Annual Meeting*, pages 7–11.
- [Li et al., 2014] Li, B., Krushinsky, D., Reijers, H. A., and Van Woensel, T. (2014). The share-a-ride problem: People and parcels sharing taxis. *European Journal of Operational Research*, 238(1):31–40.
- [Li et al., 2016] Li, B., Krushinsky, D., Van Woensel, T., and Reijers, H. A. (2016). The share-a-ride problem with stochastic travel times and stochastic delivery locations. *Transportation Research Part C: Emerging Technologies*, 67:95–108.
- [Ma et al., 2013] Ma, S., Zheng, Y., and Wolfson, O. (2013). T-share: A large-scale dynamic taxi ridesharing service. In *Data Engineering (ICDE), 2013 IEEE 29th International Conference on*, pages 410–421.
- [Michalak et al., 1994] Michalak, S., Spyridakis, J., Haselkorn, M., Goble, B., and Blumenthal, C. (1994). Assessing users’ needs for dynamic ridesharing. *Transportation Research Record*, pages 32–32.
- [NYC Taxi & Limousine Commission, 2016] NYC Taxi & Limousine Commission (2016 (accessed November 17, 2016)). *TLC Trip Record Data*. http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml.
- [Saranow, 2006] Saranow, J. (2006). Carpooling for grown-ups—high gas prices, new services give ride-sharing a boost; rating your fellow rider. *Wall Street Journal*.
- [Sarraino et al., 2008] Sarraino, J. et al. (2008). Spatial analysis of commuting patterns at the university of toronto’s st george campus: Identifying ridesharing and public transportation potential. *University of Toronto Sustainability Office, University of Toronto*, 510.
- [Tao, 2007] Tao, C.-C. (2007). Dynamic taxi-sharing service using intelligent transportation system technologies. In *2007 International Conference on Wireless Communications, Networking and Mobile Computing*, pages 3209–3212. IEEE.
- [Wand and Jones, 1993] Wand, M. and Jones, M. (1993). Comparison of smoothing parameterizations in bivariate kernel density estimation. *Journal of the American Statistical Association*, 88(422):520–528.
- [Witzgall and Zahn, 1965] Witzgall, C. and Zahn, C. T. (1965). Modification of edmonds maximum matching algorithm. *J. Res. Nat. Bur. Standards Sect. B*.
- [Xiang et al., 2008] Xiang, Z., Chu, C., and Chen, H. (2008). The study of a dynamic dial-a-ride problem under time-dependent and stochastic environments. *European Journal of Operational Research*, 185(2):534–551.
- [Xing et al., 2009] Xing, X., Warden, T., Nicolai, T., and Herzog, O. (2009). Smize: a spontaneous ride-sharing system for individual urban transit. In *German Conference on Multiagent System Technologies*, pages 165–176. Springer.