Online Fair Allocation in Autonomous Vehicle Sharing

Wenjie Bai, Jinqi Quan, Luoyi Fu, Xiaoying Gan, Xinbing Wang
Dept. of Electronic Engineering, Shanghai Jiao Tong University, P.R. China
Emails:{baiwenjie, quanjnq, yiluofu, ganxiaoying, xwang8}@sjtu.edu.cn

***Abstract*—Autonomous vehicle industry is developing by leaps and bounds. We envisage a situation where autonomous vehicles act as taxis to transport clients. During the transportation, vehicle sharing is enabled and encouraged. We propose fair allocation in payment for passengers of a shared vehicle. The allocation is endowed with the characteristics of envy freeness and maximin utilities. Furthermore, we devise working mechanisms for autonomous vehicles, including (i) planning routes with minimum detour; (ii) choosing clients to recommend vehicle sharing; (iii) carrying passengers continuously. Our work may serve as a good preparation for the marketization of autonomous taxis.**

I. Introduction

The transportation industry is about to undergo unprecedentedly dramatic changes due to the technology of autonomous vehicles or self-driving cars. Since The Google self-driving car project (Waymo) [1] was revealed in 2010, this technology has been given priority by some business magnates such as Daimler [2], Nissan [3], Baidu [4]. Startups like drive.ai [5] and Otto [6] have been bent on relevant research and development. Support from government has been extended and intensified.

Though in its testing stage, high expectation is placed on this safe, efficient and intelligent transportation form. It is commonly envisioned to be a remedy or even a ultimate solution for current traffic problems such as congestion, road accidents and will certainly benefit all aspects of transportation industry. We are interested in possible benefits autonomous drive can bring for taxi market when vehicle sharing is allowed. Consider the following scenario.

A client named Alice was about to commute to her destination, so she called an autonomous taxi to pick her up. A few minutes after Alice's boarding, the vehicle-mounted system gave her an offer—"a nearby client named Bob is calling for taxi service right now, his destination is roughly in the same direction as we are going, but it will take a detour of 4 blocks to pick Bob up. Are you willing to share this vehicle in exchange for a reduction of 10$ in payment?" There was plenty of room in the vehicle and she was not in a hurry. Most importantly, the discount in payment is quite appealing. If Alice is environmentally conscious, the fact that vehicle sharing is beneficial to the environment and alleviates traffic pressure might pop up in her mind. Driven by all these impetuses altruistically, environmentally and financially, Alice accepted the offer.

A few problems arise from the above scenario.

- Why it is Bob who was recommended in the offer instead of someone else who was calling for taxi service at that time? Namely, how to select clients locating in different places and heading for different destinations to recommend vehicle sharing?
- How to design algorithms for vehicle routing to transport clients with minimum detour distance?
- How much should the system charge different clients to facilitate ungrudging vehicle sharing despite of possible detour? In other words, how to incentive clients to participate in vehicle sharing?

Although these problems are introduced in a special case of two clients, they virtually abound in general case where consecutive clients call for services, get on and off.

In order to tackle these problems, we improve the extant minimum detour algorithm to plan routes for a shared vehicle aiming for multiple destinations. An scheme is designed to select clients for vehicle sharing with the guarantee of non-negative utilities and without the violation of detour distance ceiling. Furthermore, we implement fair allocation in payment to incentive clients to share vehicles. Our allocation solution guarantees envy freeness and maximizes the minimum of utilities. In the end, mechanisms are devised on which the transportation decisions of autonomous vehicles are based. Being incentivized to share vehicles, passengers could gain financially and help reduce discharge of vehicle effluvia.

A. Contribution

This paper studies online price allocation in the scenario of autonomous vehicle sharing with clients' uncertain arrivals and designated departures. Operation mechanisms of autonomous vehicles acting as shared taxis are also devised. The contributions of this paper are summarized as follows.

- To our best knowledge, it is the first work to deal with payment allocation in vehicle sharing from a economical perspective. Our work may serve as a preparation for the marketization of autonomous taxis.
- We establish conditions for feasible allocations which guarantee that clients can gain non-negative utilities at the expense of acceptable detour distance. Not content with feasibility, we pursue fairness which is endowed with the characteristics of envy freeness and maximin utilities.

978-1-5090-5019-2/17/$31.00 ©2017 IEEE

- we use finite state machine to delineate the sharing process of autonomous vehicles and to explain inherent algorithm for the sake of explicitness and effectiveness.

B. Related Work

Allocation or division problem has been one of the most resilient and intriguing problems in economic scene, which involves a group of sharers and what they are about to share—whether it be resources or responsibilities. Based on whether it is true or not that all sharers are present at the time of allocation, allocation problems are classified into offline problems [7]– [9] and online ones [10]– [12]. In recent work, YA'AKOV (KOBI) GAL et al. [9] studied rent division problem: to match housemates with rooms and to charge different rooms so that each housemate is assigned a room and the price of all rooms sum up to the total rent. The author designed polynomial-time algorithms for optimization under the envy freeness constraint and applied the algorithms in a fair division website—Spliddit (www.spliddit.org) to measure practical benefits. Other pricing schemes are considered in [14] and [16] from the perceptive of game theory and auction theory, respectively.

Walsh [10] is the first to study online cake cutting problem where agents arrive and depart during the process of dividing a resource. He applied the procedures such as cut-and-choose and the Dubins-Spanier moving knife to fair division of online problems. A comparison is also made between the online version of two procedures in fields of resistance to collusion and practical performance and the results showed favour to the online cut-and choose procedure. Friedman et al. [11] studied dynamic fair division with the requirement of disrupting the allocation of less existing agents whenever a new one arrives. Optimal recursive mechanisms are proposed to allocate a divisible resource, which proved to satisfy desirable game theoretic properties.

This paper is organized as follows. Section II presents the system model. Section III deals with the special case of two clients and Section IV handle cases involved in consecutive clients in general cases. In section V, we present simulation results. Finally, section VI concludes the whole paper.

II. System Model

We ideally model city street network using grid graph, which is formally defined as a planar graph whose vertices are assigned integer coordinates such that adjacent vertices (neighbors) only differ by one in either horizontal axis or vertical axis. For a grid graph $G(V,E)$, where V denotes the set of vertices and E the set of edges, vertices indicate locations and edges indicate streets, and we refer the length of a single edge as a block. The graph we are using is necessarily a connected graph—because no traffic location in a city is unreachable—but does not have to be a complete graph. A path of length k in a grid graph G is a sequence $v_0, e_1, v_2 \ldots, e_k, v_k$ of vertices and edges, such that $e_i = \{v_{i-1}, v_i\}$. We simplify the sequence as $v_0, v_2 \ldots, v_k$ with the implication that there exists a edge between adjacent vertices in the sequence. A n-stops path $(V_1, \ldots, V_n)$ of length k is a path $v_0, v_2 \ldots, v_k$, such that $v_0 = V_1, v_k = V_n$ and $\exists v_j, j \in [2, k-1]$ s.t. $v_j = V_i, \forall i \in [2, n-1]$. In short, a n-stops path of length k is a path with n vertices fixed and $k-n$ vertices alterable. For the source vertex S whose coordinate is (x_S, y_S) and destination vertex D coordinating in (x_D, y_D), $d_M(S,D)$ denote the Manhattan distance. Thus, $d_M(S,D) = |x_S - x_D| + |y_S - y_D|$. The neighbors of an intermediate vertex u, according to their Manhattan distance to destination vertex D, fall into two partitions. A neighbor v of u is said to be D-positive if $d_M(v,D) = d_M(u,D) - 1$ and is D-negative if $d_M(v,D) = d_M(u,D) + 1$. Intuitively, D-positive neighbors of u are closer to destination vertex D by one in terms of Manhattan distance while D-negative neighbors are farther by one. The detour number t of a path from S to u consisting of vertices $v_0 = S, v_1, \ldots, v_n = u$ equals to the number of vertices v_i for which v_i is a D-negative neighbor of v_{i-1}.

Controlled by an online system, autonomous vehicles act as taxis and provide clients transportation services. Clients who are in need of such service submit their request $\boldsymbol{r}$ online which include three elements: current location S, destination D and client type θ—a quantity to measure how a client discount detour distance in her/his utility. Particularly, a client with type $\theta = 0$ is considered as totally altruistic for she/he would like to share a vehicle no matter how long the detour might be. Based on information in a client's request and the availability of nearby vehicles, the online system decides: (i) whether to pick up that client at once or keep her/him waiting for a moment, (ii) whether to seat her/him in a shared vehicle or dispatch a totally vacant one. During the transportation autonomous vehicles always drive along the shortest path between two locations, hence unnecessary detour is not involved.

Clients are informed of the price for rides without companies in advance and the charged price would reduce considerably when they share vehicles with other clients. Passengers on a vehicle, along with the client who is calling for service, are offered a fair allocation, whenever such an allocation exists, conditional on sharing the vehicle together. An allocation is a vector of prices $\boldsymbol{p}^s$ with its each entry $p_i^{s_{i-}}(\boldsymbol{r}_i)$ assigned corresponding clients ($i-$ denotes all clients with whom client i share a vehicle). This price vector updates in entries and enlarges in dimension when a new client gets on board.

III. The Case Of Two Clients

We start with the case of two clients. For the readability and convenience of narration, we still adopt convention of names such as Alice and Bob to refer to the first and second client respectively. Alice submitted transportation request $\boldsymbol{r}_1 = (S_1, D_1, \theta_1)$, and before long the online system replied with confirmation and informed her of the expected price $p_1^{s_0}(\boldsymbol{r}_1)$ for her jaunt with path length $d(S_1, D_1)$ (she did not need to pay at once). Not long after Alice's boarding, Bob uttered his transport demand from S_2 to D_2 and reports his type θ_2 (the vehicle is located at I_1 at that time). Then the online system offers both Alice and Bob an allocation plan $(p_1^{s_2}(\boldsymbol{r}_1), p_2^{s_1}(\boldsymbol{r}_2))$ (with the prerequisite of feasibility and fairness of the allocation, which we will formally define later) where $p_i^{s_{i-}}(\boldsymbol{r}_i)$ is the price for client i under the condition of agreement on vehicle sharing with the other client. As $\boldsymbol{r}$ is usually obvious from the context, we simply write $p_i^{s_{i-}}$ instead of $p_i^{s_{i-}}(\boldsymbol{r}_i)$. Both Alice and Bob have two options: (i) accept the allocation and share the vehicle with the other; (ii) refuse the allocation and commute by herself/himself alone (it means Bob has to call an unoccupied vehicle instead).

A. Feasibility

Suppose the allocation is accepted by both Alice and Bob, the taxi then automatically change its predefined route from (S_1, D_1) to $(S_1, I_1, S_2, D_1, D_2)$ or $(S_1, I_1, S_2, D_2, D_1)$, depending on which path is shorter. Therefore the actual path length of Alice's jaunt is

$$\begin{aligned} d_1 &= d(S_1, I_1) + \min\{d(I_1, S_2, D_1), d(I_1, S_2, D_2, D_1)\} \quad (1)\\ &= d(S_1, I_1, S_2) + \min\{d(S_2, D_1), d(S_2, D_2) + d(D_2, D_1)\}. \end{aligned}$$

Note that if the vehicle chooses path $(S_1, I_1, S_2, D_1, D_2)$, it drops Alice off at location D_1, hence the path Alice experienced is (S_1, I_1, S_2, D_1). Similarly, the expected path length for transporting Bob is

$$\begin{aligned} d_2 &= \min\{d(S_2, D_1, D_2), d(S_2, D_2)\} \\ &= \min\{d(S_2, D_1) + d(D_1, D_2), d(S_2, D_2)\}. \end{aligned} \tag{2}$$

By sharing the vehicle with Bob, Alice obtained utility

$$u_1(\theta_1, d_1) = p_1^{s_0} - p_1^{s_2} - \theta_1 d_1^{s_2}, \tag{3}$$

where $d_1^{s_2}$ is the detour distance of Alice for sharing the vehicle, explicitly,

$$d_1^{s_2} = d_1 - d(S_1, D_1). \tag{4}$$

Similarly, the utility of Bob for taking a vacant seat of a occupied vehicle is

$$u_2(\theta_2, d_2) = p_2^{s_0} - p_2^{s_1} - \theta_2 d_2^{s_1}. \tag{5}$$

where $p_2^{s_0}$ is the price for Bob's transportation provided he would take another vehicle by himself alone, and $d_2^{s_1}$ is the detour distance of Bob for boarding a shared vehicle instead of calling for a new one, explicitly,

$$d_2^{s_1} = d_2 - d(S_2, D_2). \tag{6}$$

There is no reason that Alice Bob share the same vehicle if they obtain negative utilities by doing so. Hence a necessary condition for the feasibility of an allocation is individual rationality. An allocation satisfies individual rationality constraint if each client of type θ receives a non-negative utility by accepting the allocation, i.e.,

$$u_i(\theta_i, d_i) \geq 0, \quad \forall i \in \{1, 2\}. \tag{7}$$

We simply write $u_i(\theta_i, d_i)$ as u_i when it is clear in context. Apart from individual rationality, the maximum detour distance for each client must be bounded, for no one would like to share a vehicle with someone whose expected path deviates far from current path no matter how low the charged price is. Therefore, we have

$$d_i^{s_{i-}} \leq C_d, \quad \forall i \in \{1, 2\}, \tag{8}$$

where C_d is the ceiling of detour distance. In order to make an allocation feasible, the prices of all clients must sum up to the total price, which is the essence of sharing. We assume that the online system charges according to path length, thus

$$\sum_{i=1}^{2} p_i^{s_{i-}} = p_0 d, \tag{9}$$

where p_0 is the price for driving through one block, and d is the expected path length for the entire transportation. The connotation of equation (9) is hereafter termed as sharing principle.

Definition 1: An allocation is feasible if and only if it satisfies individual rationality, the constraint of maximum detour, and sharing principle.

Feasibility in itself is insufficient to guarantee satisfaction of clients. Although clients who share a vehicle indeed obtain a positive amount of utilities, their utilities may differ greatly. Clients who are transported directly might gain more than clients who took a long detour, this is apparently unfair and would inevitably incur complaint and discontent. therefore, an allocation has to be considered as "fair" in order to incentive clients to approve.

B. Fairness

After identifying the feasibility of an allocation, we now turn our attention to fairness of an allocation. Envy freeness is an useful notion for fairness, implying that no client would like to "switch" allocations with other client. In the scenario of vehicle sharing, we formally define envy freeness as follows:

Definition 2: An allocation is envy free (EF) if the utility of each client for her/his jaunt at its price is at least as high as other's jaunt at the price of that jaunt. Namely,

$$u_i(\theta_i, d_i) \geq u_{i-}(\theta_i, d_{i-}), \quad \forall i \in \{1, 2\}. \tag{10}$$

Though compelling envy freeness is, we would also like to bring in other fairness criterions to refine EF solutions. Intuitively, maximizing the minimum utility and minimizing the difference in utilities are two ways to approach freeness. Our objective is then to optimize a function of clients' utilities under feasibility and EF constraints such that an allocation is practically feasible and regarded as fair enough by all clients. Therefore, we have the optimization problem.

$$\begin{aligned} \max_{p_1^{s_2}, p_2^{s_1}} \quad & f\left(u_1(\theta_1, d_1), u_2(\theta_2, d_2)\right) \\ \text{s.t.} \quad & p_1^{s_0} - p_1^{s_2} - \theta_1 d_1^{s_2} \geq p_2^{s_0} - p_2^{s_1} - \theta_1 d_2^{s_1}, \\ & p_2^{s_0} - p_2^{s_1} - \theta_2 d_2^{s_1} \geq p_1^{s_0} - p_1^{s_2} - \theta_2 d_1^{s_2}, \\ & p_1^{s_0} - p_1^{s_2} - \theta_1 d_1^{s_2} \geq 0, \\ & p_2^{s_0} - p_2^{s_1} - \theta_2 d_2^{s_1} \geq 0, \\ & p_1^{s_2} + p_2^{s_1} = P, \\ & d_1^{s_2} \leq C_d, \\ & d_2^{s_1} \leq C_d. \end{aligned} \tag{11}$$

Envy freeness is enforced by the starting two constraints, and the rest constraints guarantee that the allocation is feasible. If our priority concern is to maximize the minimum utility, then we set

$$f(u_1(\theta_1, d_1), u_2(\theta_2, d_2)) = \min\{u_1, u_2\}. \tag{12}$$

Such a solution is hereafter called maximin solution. One may test other fair criterions as well.

Note that though we introduce feasibility and fairness of an allocation in the case of two clients, these conception will still be valid in general case with minor adjustments in mathematical forms.

IV. General Case

A. Finding the Shortest Path

Because the mechanism of vehicle sharing necessitate shortest path. We would like to analysis how to find the shortest path in a grid graph before dealing with the general case. We improve the minimum detour algorithm proposed in [13] to find the shortest path in a grid graph, which runs between $o(\sqrt{n})$ and $o(n)$. The algorithm uses to two stacks, the P and the N stacks, to store D-positive neighbors and D-negative neighbors respectively. Initially, all vertices are unlabeled, we use U_L to denote the set of unlabeled veritces. A vertex is labeled only when it is visited. Labeled vertices are added in an array $List$. The algorithm searches a path with detour number $0, 1, 2 \ldots$, in order to find the shortest one. We briefly recount the details in Algorithm 1 and Algorithm 2.

B. Consecutive clients for one vehicle

Different from the special case of two clients, in practice, clients get on and off a vehicle in a consecutive manner without exceeding vehicle capacity N—the maximal number of clients that can be concurrently carried. And d_i and $d_i^{s_{i-}}$ updates with the agreement of new allocations. The procedures we used to deal with the case of two clients can be extended to implement fair allocation in general case.

Suppose a shared vehicle with n passengers is on its way. Passenger i got on the vehicle at t^i and will get off at time t_i (they are expected to do so). At some time point $t \in [\max\{t^i\}, \min\{t_i\}]$, client j submitted service request. We let set of passengers who are on board be P. And P' denotes union of passenger set and the client who is calling for taxi service at that time, namely, $P' = P \cup \{j\}$. The previous allocation for n passengers who have already been seated is $\boldsymbol{p^n} = \{p_i^{s_{i-}}\}$, $i \in P$, $i-$ denotes $P/\{i\}$. The new allocation is $\boldsymbol{p^{n+1}} = \{p_i^{s_{i-}}\}$, $i \in P'$, $i-$ denotes $P'/\{i\}$. Therefore the utility of passenger i for accepting the new allocation is

$$u_i(\theta_i, d_i) = \boldsymbol{p_i^n} - \boldsymbol{p_i^{n+1}} - \theta_i d_i^{s_{i-}}, \quad \forall i \in P, \tag{13}$$

where $\boldsymbol{p_i^{n+1}}$ and $\boldsymbol{p_i^n}$ denote the ith entry of price vectors. The utility of client j for accepting the new allocation is

$$u_j(\theta_j, d_j) = p_j^{s_0} - \boldsymbol{p_{n+1}^{n+1}} - \theta_j d_j^{s_{j-}}. \tag{14}$$

Suppose the new allocation is accepted by all previous passengers $i \in P$ and client j. Then client j is allowed to board as a passenger. Accordingly, travel route of the vehicle is redesigned and time of arrive t_i is recalculated. Then the expected path length of her/his jaunt is

$$d_i = d[t^i, t_i], \tag{15}$$

where $d[t^i, t_i]$ is the length of path along which the vehicle travels in time window $[t^i, t_i]$. S_i, D_i is designed and travel route is preplanned. Hence, $d[t^i, t_i]$ can be computed, though time window $[t^i, t_i]$ is not totally predetermined. Compared to the case of two clients, the implication of $d_i^{s_{i-}}$ is extended in a general sense. $d_i^{s_{i-}}$ is the total detour distance of passenger i, in other words, it is the difference in length between actual

Algorithm 1 Finding the labeled list

Require: S, D, $G(V, E)$
Ensure: $List$
initial $u \leftarrow S$, $d \leftarrow 0$, $U_L \leftarrow V$,
repeat
  $List[end+1] \leftarrow u$, $U_L \leftarrow U_L/\{u\}$
  for all v such that $d_M(v, D) = d_M(u, D) + 1$ **and** $v \in U_L$ **do**
    PUSH v in N-stack
  end for
  $NEXT \leftarrow \emptyset$
  for all v such that $d_M(v, D) = d_M(u, D) - 1$ **and** $v \in U_L$ **do**
    if $NEXT = \emptyset$ **then**
      $NEXT \leftarrow v$
    else
      PUSH v in P-stack
    end if
  end for
  if $NEXT \neq None$ **then**
    $u \leftarrow NEXT$
    CONTINUE
  end if
  repeat
    if P-stack is empty **then**
      if N-stack is empty **then**
        return No path exists from S to D
      else
        P-stack <–> N-stack
        $d = d + 1$
      end if
    end if
    POP v from P-stack
  until $v \in U_L$
  $u \leftarrow v$
until $u = D$
return $List$

Algorithm 2 Finding the shortest path in a grid graph

Require: $List$
Ensure: Path (S, D)
repeat
  $next = \emptyset$
  for $i = L[index(v) + 1] : L[end]$ **do**
    if $L[i]$ is a neighbor of v **then**
      $next \leftarrow L[i]$
    end if
  end for
  if $next = \emptyset$ **then**
    REMOVE v from $List$
    POP v from Path
  else
    PUSH v in Path
    $v \leftarrow next$
  end if
until $v = D$
return Path

path from S_i to D_i and the shortest path. Namely,

$$d_i^{s_i-} = d[t^i, t_i] - d(S_i, D_i). \tag{16}$$

Expected time t_i updates when new client boards the shared vehicle, d_i and $d_i^{s_i-}$ may change as well.

The new allocation $\boldsymbol{p^{n+1}}$ is feasible if it satisfies the following constraints.

$$u_i(\theta_i, d_i) \geq 0, \quad \forall i \in P', \tag{17}$$

$$d_i^{s_i-} \leq C_d, \quad \forall i \in P', \tag{18}$$

$$\sum_{i\in P'} p_i^{s_i-} = p_0 d[\min\{t^i\}, \max\{t_i\}]. \tag{19}$$

Therefore, the linear programming problem that the online system confront is formulated as $LP(n+1)$

$$\begin{array}{ll} \max\limits_{\boldsymbol{p}_1^{\boldsymbol{n+1}},\ldots,\boldsymbol{p}_{n+1}^{\boldsymbol{n+1}}} & \min\{u_1,\ldots,u_{n+1}\} \\ \text{s.t.} & \sum\limits_{\forall i\in P'} p_i^{s_i-} = p_0 d[\min\{t^i\}, \max\{t_i\}], \\ & u_i(\theta_i, d_i) \geq u_j(\theta_i, d_j), \forall i,j \in P', \\ & u_i(\theta_i, d_i) \geq 0, \forall i \in P', \\ & d_i^{s_i-} \leq C_d, \forall i \in P'. \end{array} \tag{20}$$

Envy freeness is guaranteed by the second form of constraint. For a $LP(n+1)$ problem, there are a total number of $A_{n+1}^2 + 2n + 3$ constraints. It is obvious that $d_i^{s_i-}$ is determined by all $S_i, D_i, i \in P'$. Hence in order to solve $LP(n+1)$, we could first compute $d_i^{s_i-}, \forall i \in P'$ and examine the condition $d_i^{s_i-} \leq C_d, \forall i \in P'$. If these conditions are not satisfied, $LP(n+1)$ has no feasible solution. If they do hold, we then solve the relaxed linear programming $LP'(n+1)$.

$$\begin{array}{ll} \max\limits_{\boldsymbol{p}_1^{\boldsymbol{n+1}},\ldots,\boldsymbol{p}_{n+1}^{\boldsymbol{n+1}}} & m \\ \text{s.t.} & \sum\limits_{i\in P'} p_i^{s_i-} = p_0 d[\min\{t^i\}, \max\{t_i\}], \\ & u_i(\theta_i, d_i) \geq u_j(\theta_i, d_j), \forall i \in P', \\ & m \leq u_i(\theta_i, d_i), \forall i \in P', \\ & u_i(\theta_i, d_i) \geq 0, \forall i \in P'. \end{array} \tag{21}$$

If $LP'(n+1)$ admits a optimal solution $\boldsymbol{p}^s$, it is also the optimal solution of $LP(n+1)$.

As illustrated in Fig. 1, we use a finite state machine to demonstrate the sharing process and explain inherent algorithm. Given the vehicle capacity N, the state of a vehicle can be classified into $N+1$ states according to the number of passengers. Basically, the state of a shared vehicle transits to next state (except for STATE N) due to trigger of haphazard event. In particular, only when a new client requires taxi service and a fair allocation does exist, will the vehicle pick up that new client. The state of a shared vehicle transits to last state (except for STATE 0) only when one of passengers is dropped off at her/his destination. Many a client might get on or off at the same time, this case is regarded as a continuous transition of state with infinitesimal sojourn time for intermediate state. Details about state transition conditions could be found in the following.

Initial state—**STATE 0**: drive aimlessly or parking at a specific spot when no one calls for taxi service; head for S_1 when client 1 submits a transportation request, after client 1's boarding, transit to STATE 1.

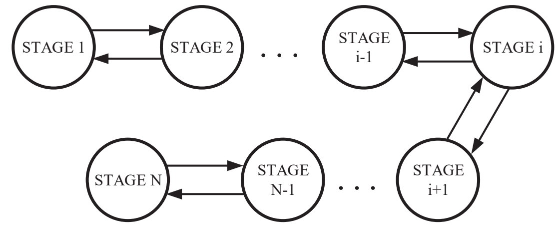


Fig. 1. States of A Shared Vehicle

STATE 1: head for D_1, Transit to STATE 0 when reaching D_1. If client 2 submits a transportation request, solve $LP(2)$. If a fair allocation exists, then head for S_2 to pick up client 2, after client 2's boarding, transit to STATE 2; if there exists no optimal solution, ignore client 2's request.

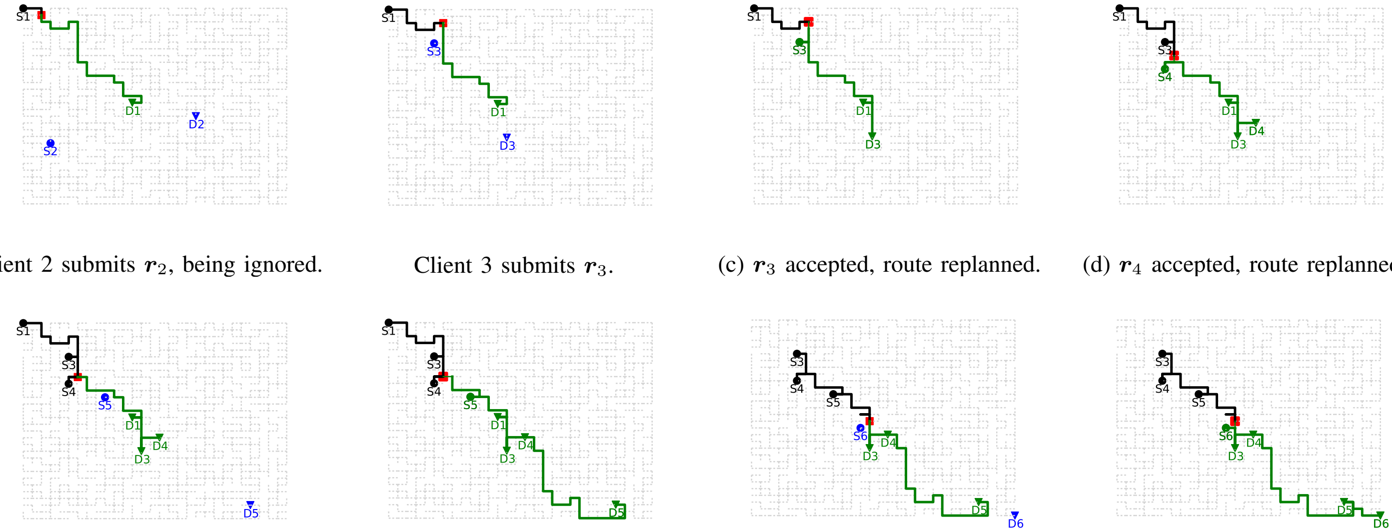
STATE i$(1 < i < N)$: head for the closest location among passengers' destinations, Transit to STATE i-1 when reaching that location. If client j submits a transportation request, solve $LP(i+1)$. If fair allocation exists, then head for S_j to pick up client j, after the boarding of client j, transit to STATE i+1; if there exists no optimal solution, ignore request from client j.

STATE N: head for the closest location among passengers' destinations, transit to STATE N-1 when reaching that location. If client k submits a transportation request, ignore request from client k.

With the fluctuation of state, an shared vehicle runs ceaselessly to transport passengers and pick up new ones. During the process fair allocations are implemented.

V. Simulation Results

A connected grid graph is generated randomly to explicate our mechanisms. We perform simulations with the following parameters: $N = 4$, $\theta_1 = 0.2$, $\theta_2 = 0.3$, $\theta_3 = 0.3$, $\theta_4 = 0.2$, $\theta_5 = 0.3$, $\theta_6 = 0.4$, $p_0 = 1$, $C_d = 10$. The sharing process is illustrated in Fig. 2. We use filled circles to indicate starting points and triangles to indicate destinations. Black line represents the path which has been passed, and green line represents the expected path which lies beyond. Red square denotes current location of the vehicle. Requests (represented by a pair of blue circle and triangle) are consecutively submitted and responded as time passes by. Client 1 is initially charged 30 for $d(S_1, D_1) = 30$. The request of client 2 is ignored (as shown in Fig. 2 (a)) because no fair allocation exists. The request of client 3 is accepted (as shown in Fig. 2 (b)(c)) and fair allocation is $p_1^{s_3} = 23$, $p_1^{s_3} = 15$. We list the rest allocations and actual detour distance in Table I and Table II, respectively. We observe the payment reduces remarkably at the expense of reasonable detour distance. For example, client 1 only needs to pay 16.35 when she gets off (almost half of the original payment 30). Note also that the travel distances of client 3 and client 4 are nearly the same while client 3 pays less than client 4, this is because the number of passengers



Client 2 submits $\boldsymbol{r}_2$, being ignored. Client 3 submits $\boldsymbol{r}_3$. (c) $\boldsymbol{r}_3$ accepted, route replanned. (d) $\boldsymbol{r}_4$ accepted, route replanned.

(e) Client 5 submits $\boldsymbol{r}_5$. (f) $\boldsymbol{r}_5$ accepted. (g) Client 1 gets off; client 6 submits $\boldsymbol{r}_6$. (h) $\boldsymbol{r}_6$ accepted.

Fig. 2. Sharing process of an autonomous vehicle.

TABLE I
FAIR ALLOCATIONS

Events	$\boldsymbol{r}_4$ accepted	$\boldsymbol{r}_5$ accepted	$\boldsymbol{r}_6$ accepted
Allocations	$p_1^{s_{3,4}} = 19$, $p_3^{s_{1,4}} = 11$, $p_4^{s_{1,3}} = 16$	$p_1^{s_{3,4,5}} = 16.35$, $p_3^{s_{1,4,5}} = 8.35$, $p_4^{s_{1,3,5}} = 13.35$, $p_5^{s_{1,3,4}} = 37.95$	$p_3^{s_{4,5,6}} = 2.1375$, $p_4^{s_{3,5,6}} = 7.1375$, $p_5^{s_{3,4,6}} = 32.3375$, $p_6^{s_{3,4,5}} = 28.3875$

TABLE II
ACTUAL DETOUR DISTANCE

Client	1	3	4	5	6
$d(S_i, D_i)$	30	22	20	40	34
$d_i^{s_{i-}}$	8	10	10	8	8

with whom client 3 shares the vehicle is one more than that of client 4.

VI. Conclusion

This paper envisions the scenario where autonomous vehicles serve as taxis in city transportation. We study online price allocation when vehicle sharing is allowed. Clients call for services haphazardly and are heterogeneous in the willingness of vehicle sharing. We design working mechanisms for autonomous vehicles to help them make decisions about how to respond requests from clients and how to plan routes. During the sharing process, fair allocations are implemented, which guarantee envy freeness and maximin utilities.

Hopefully, our work can be a good preparation for commercial operation of autonomous taxis in the future. Passengers would benefit from vehicle sharing financially; in addition, emission of greenhouse gases could be reduced because occupancy of previously vacant seats leads to less traffic mileage. In future work, we model vehicle network as ad hoc network [15] to bring in additional rules to take overall control and handle dilemmas.

Acknowledgment

This work was supported by NSF China (No. 61532012, 61325012, 61521062, 61672342, 61602303 and 91438115) and Science and Technology Innovation Program of Shanghai (Grant 17511105103).

References

[1] https://www.google.com/selfdrivingcar/
[2] https://www.daimler.com/innovation/autonomous-driving/mercedes-benz-future-truck.html
[3] http://nissannews.com/en-US/nissan/usa/releases/nissan-announces-unprecedented-autonomous-drive-benchmarks
[4] http://usa.baidu.com/adu/
[5] https://www.drive.ai/
[6] https://www.ottomotors.com/
[7] D. K. Herreiner, C. D. Puppe, "Envy freeness in experimental fair division problems," *Theory and Decision*, 2009, 67(1), pp. 65-100.
[8] D. A. Procaccia and J. Wang, "Fair enough: guaranteeing approximate maximin shares," in *Proc. of the fifteenth ACM Conf. on Economics and computation*, New York, USA, 2014, pp. 675-692.
[9] Y. Gal et al., "Which is the fairest (rent division) of them all?" in *Proc. 7th ACM Conf. on Economics and Computation*, New York, USA, 2016, pp. 67-84.
[10] T. Walsh, "Online cake cutting" in *International Conf. on Algorithmic Decision Theory* Berlin Heidelberg, German, 2011, pp. 292-305.
[11] E. Friedman, C. A. Psomas, S. Vardi, "Dynamic fair division with minimal disruptions," in *Proc. 6th ACM Conf. on Economics and Computation*, New York, USA, 2015, pp. 697-713.
[12] G. Goel, V. Mirrokni, R. P. Leme, "Clinching auctions with online supply,"in *24th Acm-Siam Symposium on Discrete Algorithms*, 2012, pp. 605-619.
[13] F. O. Hadlock, "A shortest path algorithm for grid graphs", *Networks*, vol. 7, no. 4, pp. 323-334, 1977.
[14] L. Gao, X. Wang, Y. Xu, et al. "Spectrum Trading in Cognitive Radio Networks: A Contract-Theoretic Modeling Approach," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 4, pp. 843-855, 2011.
[15] X. Wang, W. Huang, S. Wang, et al. "Delay and Capacity Tradeoff Analysis for MotionCast," *IEEE/ACM Transactions on Networking*, vol. 19, no. 5, pp. 1354-1367, 2011.
[16] L. Gao, Y. Xu, X. Wang, "MAP: Multiauctioneer Progressive Auction for Dynamic Spectrum Access", *IEEE Transactions on Mobile Computing*, vol. 10, no. 8, pp. 1144-1161, 2010.