Online Decentralised Mechanisms for Dynamic Ridesharing*

Extended Paper with Supplementary Material

Nicos Protopapas
Archimedes/RC Athena
Athens, Greece
n.protopapas@athenarc.gr

Vahid Yazdanpanah
University of Southampton
Southampton, United Kingdom
V.Yazdanpanah@soton.ac.uk

Enrico H. Gerding
University of Southampton
Southampton, United Kingdom
eg@ecs.soton.ac.uk

Sebastian Stein
University of Southampton
Southampton, United Kingdom
ss2@ecs.soton.ac.uk

ABSTRACT

Ridesharing services promise an exciting new future for urban mobility. A carefully designed ridesharing system will decrease congestion levels and increase air quality. However, an effective system needs to capture *online* demand, where users do not schedule their trips in advance but instead appear and ask for a ride right away. Such online demands require rerouting to be efficient. We call this setting with online demands and available rerouting "dynamic ridesharing" and propose a market-based mechanism where the prospective riders are provided with a menu of choices between several available cars. Our algorithm incentivises users to share their rides and guarantees riders' utility by properly compensating riders whose routes change during their journey. We provide numerical results, comparing our algorithm against natural benchmarks representing real-world ridesharing services for several cases involving efficiency, fairness, and environmental impact.

KEYWORDS

Ridesharing, Mechanism Design, Dynamic Algorithms

1 INTRODUCTION

Shared mobility services have the potential to revolutionise urban transportation by offering a more convenient and efficient way for people to get around. Efficient implementations of such systems can reduce traffic congestion and carbon emissions [2, 15], providing substantial economic, environmental and societal benefits in large urban areas. As such, it can substantially aid the efficient implementation of Sustainable Development Goals [31]. However, for such a system to truly be successful, it must be able to handle the unique challenges that come with *dynamic, on-demand* requests from *strategic* participants. This demanding, yet inevitable flow of requests can put a strain on the system, making it difficult to match riders with available cars in a way that meets their preferences and ensures an acceptable level of compromise [7, 10, 35].

To address the challenges of matching cars and riders in ridesharing, auction-based and mechanism design approaches [6, 11, 20, 21, 27, 28, 39] have been proposed in the literature[1]. However, integrating the vehicle-rider matching side of the dynamic ridesharing problem [1] with the routing side is an open challenge.

Related Work on On-demand Routing. Dynamic ridesharing as a concept refers to an automated system matching riders with cars at very short notice [1]. The problem has received much attention from an algorithmic point of view (see, e.g., [3, 5, 19, 22, 37, 38]), but the literature is not so rich when participants' incentives need to be addressed. A series of papers [13, 18, 27] consider the online allocation of passengers, but with the crucial assumption that all requests arrive before a car starts to service these requests. Asghari and Shahabi [4] propose a system capable of handling online demands for allocating riders to drivers, but merely focused on the incentives of drivers, while ignoring incentives for riders. In contrast, Shen et al. [29] recognise the imminent transition to autonomous cars and present an online mechanism with a focus on riders' incentives. However, the riders can merely accept or reject a given allocation, without having the option to evaluate various proposals and choose the one that maximises their utility. Incentives for both the rider and the driver are considered in [30, 34], with the catch that a driver serves at most one rider at a time.

Against this background, in this paper, we propose a mechanism to address dynamic re-routing in ridesharing by offering riders a menu of choices presented by multiple cars. Rather than focusing on profit maximization (like, e.g., in [8]), we follow the perspective of a government-designed not-for-profit service which aims to increase ridesharing among riders. Our framework focuses on the challenging task of handling online arrivals, without reliance on any prior knowledge about the arrivals, with new routes computed on-the-fly to accommodate new passengers. Our main contribution is incentivising ridesharing from the rider's point of view by aggregating their preferences and compensating them for any delay caused by changes to their routes. To the best of our knowledge, we are the first to propose the use of a menu of choices for riders in the ridesharing allocation process[2]. The menu of choices allows riders to express their preferences and priorities, which are then used to optimise the allocation of riders to cars.

We evaluate our proposed mechanism against two natural benchmarks. The first benchmark is a decentralised, peer-to-peer allocation of riders to cars, similar to a taxi service – serving riders one by one on a first-come, first-served basis. The second benchmark is

*Accepted as a full paper in the 23rd International Conference on Autonomous Agents and Multi-Agent Systems.

[1]For detailed surveys, the reader is referred to [14, 25].

[2]A similar approach is also used by [32] under a slightly different context. The main difference with our approach is that their system offers only a single proposal to the users.

based on discounting, a tactic used by real-world ridesharing services[3] where the riders are charged a discounted price for choosing to share a ride with other passengers. We compare the performance of our proposed mechanism against these benchmarks in terms of various metrics, such as the total utility of the riders, fairness between the riders, and the total travelling time. The results of our evaluation show that our proposed mechanism outperforms the benchmarks in most of these metrics; even when it does not, its performance is still close to the benchmarks.

Sustainability. Our work on dynamic ridesharing is aligned with the implementation of Sustainable Development Goals (SDGs) 11, dedicated to *sustainable cities and communities*, primarily, and to SDG 13, dedicated on *climate action*, secondarily. Following the approach of [23], we focus mostly on social and environmental aspects of the ridesharing problem: our work provides a starting point for the analysis of ridesharing strategies from the point of view of a central planner, with the goal to promote ridesharing services through a self-sustainable service that is robust to strategic behaviour.

Technical Contributions. Our work introduces a novel, decentralized, market-based mechanism that dynamically allocates riders to shared cars. Unlike other proposals in the literature, our mechanism does not require riders to schedule their journeys in advance, offers the flexibility of dynamic route re-scheduling, and provides riders with a menu of choices that includes estimated travel time and cost. Crucially, we have opted for a model-free approach, and our proposed mechanism does not require any prior knowledge in order to be efficiently implemented. Our approach provides proper incentives for riders to utilize the system, and we have experimentally evaluated it for efficiency, fairness, and environmental impact, using two natural benchmarks for comparison.

The rest of the paper is structured as follows. Section 2 describes the mathematical modelling of the dynamic ridesharing system. Section 3 presents two mechanisms which work as an introduction to the model and as benchmarks for our main mechanism, presented in Section 4 and formally analysed in Section 5. In Section 6, we present an experimental evaluation of our mechanism compared to the mechanisms from Section 3, and we conclude with some final remarks in Section 7. Some proofs and experiments appear in the appendix. The code used in the experiments is given in this repository.[4]

2 SYSTEM MODEL

In this section, we present in detail the mathematical formulation of our model. We introduce formal notation and definitions, and we explain our goals and assumptions.

Time steps: Let $[k]$ denote the set $\{1, ..., k\}$ for any $k \in \mathbb{N}$. We assume discrete time-steps $t \in [T]$, for some $T \in \mathbb{N}_{\geq 0}$. A discrete time-step is the minimum time for which a rider could decide to use a vehicle instead of walking.

City Map: Let the directed graph $G = (L, E)$ denote a city map. The set L denotes pick-up and drop-off locations. For simplicity, we assume that any location can be used for the pick-up and drop-off of passengers, but our results work independently of this assumption. The set of edges E denotes directed links between the locations, and we assume that each location has at least one incoming and one outgoing edge. We assume a highly discretised map where each edge is traversed in a single time-step. This can be done by identifying pick-up and drop-off locations and combining historical data and expert advice. Let $\delta : L \times L \to \mathbb{N}_{\geq 0}$ be a function which returns the shortest distance in time-steps between two locations $a, b \in L$. Using this abstraction, we can ignore unnecessary details, such as the vehicle's average speed and road congestion, and focus on the travelling time. Formally, $\delta(a, b) := \min_{p \in P(a,b;G)} |p|$, where $P(a, b; G)$ denotes all paths connecting a to b in G. Without loss of generality, we can assume that δ satisfies the triangle inequality, i.e., $\delta(x, y) \leq \delta(x, z) + \delta(z, y)$ for any locations $x, y, z \in L$. We assume that everyone has oracle access to this function and that the shortest paths and shortest times returned by this function are accurate.

Riders: Riders are represented by strategic, autonomous agents. Let N denote the set of all riders. This set is unknown beforehand, and the riders arrive one by one in an unknown order. Each rider is described by her *type*, which is a tuple consisting of both *private* and *public* information. The private information includes (i) the arrival time $t_i \in [T]$ and (ii) the *value of time* (VoT), denoted by $v_i \in \mathbb{R}_{\geq 0}$. This value denotes a patience weight for the rider: a VoT of x implies that the rider suffers a cost of $x \cdot y$, for travelling y more time steps, compared to the optimal option. The type also includes a publicly known pair $(a_i, b_i) \in L \times L$, denoting departure and arrival locations for the rider. We assume that the riders always truthfully report their preferred pick-up and drop-off locations a_i and b_i. For the pick-up location, this can be ensured when the system refuses to accept passengers appearing in other locations than the reported. Similarly, the system can ensure truthful reporting of the destination b_i, by enforcing the rider to disembark at the location they report.[5]

The variable s_i denotes the status of a rider. Each rider first appears as *unallocated*, it is then *allocated* to a vehicle, and once picked-up becomes a *travelling* rider. A rider who reaches her destination is a *finished* rider.

Vehicles and Routes: Let M denote the set of all cars in the system. Vehicles start their trips from designated locations in the city, which we call *car depots* and denote with the set $D \subseteq L$. A *live* vehicle is one with allocated riders, and is either (i) already carrying one or more riders or (ii) empty, but moving to pick up a rider. A vehicle returning empty to the depot is not considered live, but can accept new requests without the need to return. We use m_t to denote the number of the live vehicles at time t, and M_t to denote the set of these vehicles. Each vehicle may have a professional driver or it is autonomous. In both cases, our main assumption is that cars are not strategic. Vehicle $j \in M$ is described by a set of variables: C_j denotes the vehicle's capacity, which is a constant, small integer number. Variable $\ell_{j,t} \in L$ denotes the car's location at time t, and c_j is a scalar value denoting the cost for travelling a unit of time. The set $R^A_{j,t}$ denotes the allocated riders in car j at time t, while the set $R^T_{j,t} \subseteq R^A_{j,t}$, denotes the set of travelling riders.

[3]See, e.g., https://www.uber.com/gb/en/ride/uberx-share/

[4]https://github.com/NicosProtopapas/DynamicRidesharing.git

[5]It could be possible for a rider to report some locations $a'_i \neq a_i$, and $b'_i \neq b_i$, and then use other means to get to their *final* destination. Here, we consider a_i and b_i to mean the source and destination of rider i when using the ridesharing service.

The tuple $\mathbf{r}$, denotes an order of visiting riders' pick-up and drop-off locations. Given a set of riders R, let $L(R) := \{a_i, b_i\}_{i \in R}$. An *order* $\mathbf{r}$ is defined as a sequence of pick-up and drop-off locations of the riders in R. We use the notation $r_i \in L(R)$ to denote the location in the i-th element of the order $\mathbf{r}$, and $r_x^{-1} \in |\mathbf{r}|$ to denote the position of location $x \in L(R)$ in the same order.

For an order to be valid, it must be that (i) $b_i \in \mathbf{r}$ for all $i \in R$, and (ii) $r_{a_i}^{-1} < r_{b_i}^{-1}$. The second condition is always met when $a_i \notin L(R)$. As an example, consider the set of riders $R = \{x, y, z\}$. Then $(a_x, b_x, a_y, b_y, b_z)$, $(a_z, a_y, b_y, b_z, b_x)$ are valid orders, while, $(a_x, a_y, b_y, a_z, b_z)$, and (b_z, b_x) are not valid orders for R.

Given a vehicle $j \in M$, a time step $t \in [T]$, and a valid order $\mathbf{r}$ over the set $R^A_{j,t}$, let $r_0^{j,t} = \ell_{j,t}$, $r_k^{j,t} = r_k$ for $k \in [|\mathbf{r}|]$ and $r_{|r|+1}^{j,t} \in D$. Then the *route* $\rho(\mathbf{r}^{j,t})$ is defined as a concatenated path, created by the shortest paths between the locations $r_k^{j,t}$ and $r_{k+1}^{j,t}$ for $k \in \{0, ..., |r|\}$. For simplicity, we use the simplified notation $\mathbf{r}^{j,t}$ to denote a route.

Let $\tau_i^* := \delta(a_i, b_i)$ be the ideal route cost in time for rider i, i.e., the best case scenario for rider i, e.g., using a private ride. Given a route $\mathbf{r}^{j,t}$, let $\tau(\mathbf{r}^{j,t}) := \sum_{k=1}^{|\mathbf{r}|-1} \delta(r_k, r_{k+1})$ be the total travelling time for the route to complete, without measuring the time for a car to return to a depot. Let $\tau_i(\mathbf{r}^{j,t}) := \sum_{k=1}^{r_{b_i}^{-1}} \delta(r_k, r_{k+1})$ be the total travelling time of rider i, including time spent embarked and the waiting time, starting at t_i. Similarly, we define $\phi_i(\mathbf{r}^{j,t}) := \delta(\ell_{j,t}, r_1) + \tau_i(\mathbf{r}^{j,t}) + t_i$ as the *finishing* time of rider i, and $\phi_i^* := \tau_i^* + t_i$ as the *ideal finishing* time for rider i.

Preferences: Now that we have introduced the notions of routes and cars, we discuss the preference model of the riders. The riders have quasi-linear preferences over their travelling time. The riders are utility maximisers, and the utility of the rider diminishes as the sum of travelling time and the price increases. Formally, given a route $\mathbf{r}^{j,t}$ and a car-dependent price $p_i^{j,t}$, the *utility* for rider i at time t is defined as: $u_i(\mathbf{r}^{j,t}, p_i^{j,t}) := -v_i \cdot (\phi(\mathbf{r}^{j,t}) - \phi_i^*) - p_i^{j,t}$,where the scalar value $v_i \in \mathbb{R}_{\geq 0}$ denotes the sensitivity of rider i to deviations from the optimal route from its own perspective.

In the following, we compare the utility of a rider according to two different points of view. The *projected* utility refers to the utility of a rider at the *allocation time*, before the rider actually travels. In contrast, the *ex-post* utility refers to the utility of the rider *after* she completes her journey. Hence, the former is an estimation of the latter. A rider is called *myopic* if she cares only about maximising her utility at the time of the arrival without considering future arrivals of other riders.

Mechanisms: A mechanism works in a two-step procedure: First, it receives reports from the riders on their (privately known) types, e.g., using a mobile app. Then, it produces multiple proposals, one for a subset of all available cars. More concretely, a mechanism $\mathcal{F}$ is defined using an *allocation* and *payment* component. Upon the arrival of a rider i, each car $j \in M' \subseteq M$, proposes an expansion of its current route to accommodate the new rider, using an allocation policy α^j. The proposed route is accompanied by a proposed payment, calculated according to a payment policy π^j. Then, each rider will accept one of these proposals. Importantly, the allocated car cannot change, and the allocation decision must be made immediately. The selected route and payment might change due to future arrivals, however.

Formally, at time t, and upon the appearance of rider i, each car $j \in M'$ proposes[6] a route $\hat{\mathbf{r}}^{j,t} = \alpha^j(\theta_i; \mathbf{r}^{j,t})$ and a respective payment $\hat{p}^{j,t} = \pi^j(\theta_i; \mathbf{r}^{j,t})$. Then the rider selects a new route and price pair such that: $(\mathbf{z}^{j,t}, q^{j,t}) \in \arg\max_{j \in M'} u_i(\alpha^j(\theta_i; \mathbf{r}^{j,t}), \pi^j(\theta_i; \mathbf{r}^{j,t}))$.

The pair $(\mathbf{z}^{j,t}, q^{j,t})$ is the outcome of mechanism $\mathcal{F}(\boldsymbol{\theta}; t)$ at time t. The final outcome is $\mathcal{F}(\boldsymbol{\theta})$.

Properties: To present desirable properties of the mechanisms, we recall formal definitions rooted in the literature on mechanism design [26]. The first property follows the classic notion of incentive-compatibility.

Definition 2.1. A mechanism $\mathcal{F}$ is Dominant Strategy Incentive Compatible (DSIC) iff $u_i(\mathcal{F}(\theta_i, \boldsymbol{\theta}_{-i})) \geq u_i(\mathcal{F}(\theta_i', \boldsymbol{\theta}_{-i}))$
for every rider $i \in N$, and rider's i types θ_i, θ_i' and any profile of other riders' types $\boldsymbol{\theta}_{-i}$.

A weaker notion of incentive compatibility is concerned only with myopic agents. Under this notion, any rider cannot gain by any misreport, if no other rider is allocated to the same car in the future.

Definition 2.2. A mechanism $\mathcal{F}$ is Dominant Strategy Incentive Compatible for myopic riders iff

$$u_i(\mathcal{F}(\theta_i, \boldsymbol{\theta}_{-i}^{t_i}; t_i)) \geq u_i(\mathcal{F}(\theta_i', \boldsymbol{\theta}_{-i}^{t_i}; t_i))$$

for every rider $i \in N$, rider's i types θ_i, θ_i' and arrival time t_i, and any profile of other riders' types $\boldsymbol{\theta}_{-i}^{t_i}$, which have arrived before i.

Finally, we need to know whether our mechanisms generate enough income to cover the operation costs of the rides. To be fair to the riders, we only care about the cost a car suffers while it is allocated at least one rider.

Definition 2.3. A mechanism $\mathcal{F}$ is Budget-Balanced iff $\sum_{i \in N} p_i \geq \sum_{j \in M} c_j \cdot \chi_j$ where p_i is the ex-post payment of rider i and χ_j is the total travelling time of vehicle j, while allocated with riders, once all riders are delivered.

3 DECENTRALISED MECHANISMS FOR DYNAMIC RIDESHARING

We will explain the functionality of our proposed family of mechanisms with a few simple examples. We first consider a case with only three riders, Alice, Bob, and Carlos, and a single depot. Alice asks for a ride at time t_A, to get from her current location a_A to location b_A. When Alice requests a ride, only a single car is live: Car 1, with a maximum capacity for 4 passengers, and already allocated with Bob and Carlos. Bob is already on-board, while the car is moving to pick up Carlos from location a_C. To be more precise, at the time-step before Alice's request, the car is following the route $\mathbf{r}^{1,t_A-1} = (\ell_{1,t_A-1}, a_C, b_C, b_B)$: Carlos will be picked-up and delivered, and finally Bob will arrive to his destination. Another identical car is available empty at the depot. Based on this instance, we will

[6] The set M' is a subset of all available car which precludes cars that are not valid options, e.g., because the distance between the car and the rider's pick-up location is too large.

Algorithm 1 Vehicles' proposal for FIFO.

input: rider i, current route $\mathbf{r}$ at time t, set of current riders R_j.
output: new route $\mathbf{w}$, payment $p^{i,j}$.
procedure ComputeProposal(i, j, t)
  $\mathbf{w} \leftarrow (\mathbf{r}, a_i, b_i)$ ▷ append the new rider to the end
  $p^{i,j} \leftarrow c_j \cdot (\delta(\ell_{j,t}, w_1) + \tau(\mathbf{w}))$
end procedure

Algorithm 2 Vehicles' proposal for Discount.

input: rider i, current route $\mathbf{r}$ at time t, set of current riders R_j, discount rate α.
output: new route $\mathbf{w}$, payment $p^{i,j}$.
procedure ComputeProposals(i, j, t)
  for $\hat{\mathbf{r}} \in$ RouteGenerator$(i, j, \mathbf{r})$ **do**
    $s^{\hat{\mathbf{r}}} \leftarrow \tau(\hat{\mathbf{r}})$ ▷ compute total travelling time
  end for
  $\mathbf{w} \leftarrow \arg\max_{\hat{\mathbf{r}}}(s^{\hat{\mathbf{r}}})$
  $p \leftarrow c_i \cdot (1 - \alpha) \cdot (\tau_i(\mathbf{w}) + t)$
end procedure

illustrate two simple mechanisms. We will use these mechanisms in Section 6 as benchmarks for our main contribution.

Fixed Pricing FIFO: Probably the simplest solution to allocate riders in a car is a simple *first-in first-out* strategy, where each car adds the new rider's pick-up and drop-off locations at the end of their route, upon their arrival on the system, and the rider selects the best available car.[7] The rider is charged exactly the cost of the ride. We call this the *FIFO* mechanism, and the car proposal procedure is shown in Algorithm 1.

We will explain the FIFO mechanism using the aforementioned scenario. Once Alice arrives, Car 1, which is travelling towards location a_C, proposes the route $\mathbf{r}^{1,t_A} = (\ell_{1,t_A}, a_C, b_C, b_B, a_A, b_A)$, hence scheduling Alice behind Carlos. Alice should pay $p_A^{1,t_A} = c_1 \cdot (\delta(\ell_{j,t_A}, a_A) + \delta(a_A, b_A))$. The car from the depot will propose the route $\mathbf{r}^{2,t_A} = (\ell_{2,t_A}, a_A, b_A)$, at a price $p_A^{2,t_A} = c_2 \cdot (\delta(\ell_{2,t_A}, a_A) + \delta(a_A, b_A))$. Alice then selects the route that maximises her utility.

Despite its simplicity, the Fixed Pricing FIFO satisfies some important properties. The mechanism is truthful, budget-balanced and trivially scalable. The truthfulness of the mechanism is straightforward to establish. The mechanism cares only about the starting time of the rider; hence, the only possibility of misreporting from the rider is to propose a fake arrival time in the future, but this makes little sense to the rider: this might allow future riders to be scheduled before her. On the contrary, by simply proposing her true arrival time, she ensures she is served first. The mechanism is also trivially budget- balanced, as each rider pays exactly the cost of the ride. It is also computationally attractive. A car only needs to compute two shortest paths, to propose a route and price. On the downside, this mechanism does not promote ridesharing at all. Indeed, each rider rides alone, and this can lead to highly inefficient routes. Proofs for these claims are given in Appendix A.

Discounted Payments: A simple way to promote ridesharing is to provide monetary incentives, using *discounted prices*. To that end, we present the following simple mechanism: When a new rider arrives, each car computes the best possible route which can serve all allocated riders if the new rider is included. Then, a rider who chooses to ride in an empty car will be charged the exact cost of the route. In contrast, a rider who chooses to ride in a car which includes $k > 1$ other allocated riders will be charged the total cost of the route, discounted by a value $x_k \in [0, 1]$. We call this the *Discount* mechanism. A description of the proposal phase is given in Algorithms 2 and 4.

Following the previous example, when Alice arrives, car 1 proposes a route which minimises the travelling time of the car, e.g., $\mathbf{r}^{1,t_A} = (\ell_{1,t_A}, a_C, b_C, a_A, b_A, b_B)$. The payment for Alice to use car 1 is now $p_A^{1,t_A} = (1 - x_2) \cdot c_1 \cdot \tau(\mathbf{r}^{1,t_A})$. The car from the depot will propose the route $\mathbf{r}^{2,t_A} = (\ell_{2,t_A}, a_A, b_A)$, at a price $p_A^{2,t_A} = c_d \cdot (\delta(\ell_{2,t_A}, a_A) + \delta(a_A, b_A))$. Hence Alice may have an incentive to use car 1, even if this car would increase her finishing time.

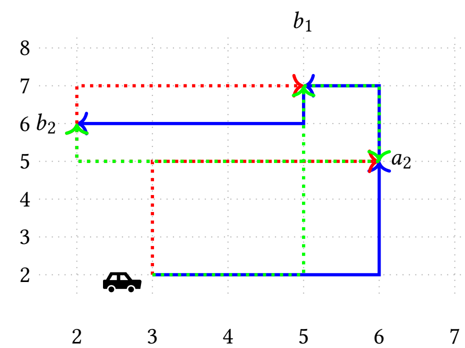


Figure 1: Example of route proposals from Algorithm 3.

It is not hard to see that this mechanism is not truthful. Indeed, a rider who knows that a new rider with a similar itinerary will arrive shortly can delay her arrival to get a trip at a discounted price. However, as we will see in Section 6, this mechanism behaves quite well with respect to total operational time.

4 DYNAMIC MARKET-BASED ALLOCATION

This section proposes our main mechanism for allocating a rider in a car, which we call the *Compensation* mechanism. The main procedure works roughly as follows: when a rider appears in the system, it requests a ride and provides her origin a_i, destination b_i, arrival time t_i and VoT v_i to the system. Then, a group of selected cars are notified about the presence of the new rider, and each car proposes a route to accommodate the request, accompanied by a payment. Notably, the new route should include the already allocated riders' destinations and, for riders not ready picked up, the origins. The payment should allow for compensating any already allocated passenger so that the utility of the already allocated riders does not decrease, and the payment should cover any additional cost added to the route. Finally, the rider accepts the route with the highest utility and pays the proposed payment. Algorithm 3 details this process.

The payment for the rider is calculated following this principle. Whenever the new route increases the finishing times of the already

[7] This is a common approach in the social choice literature [9] and, in particular, is similar to the well-known Random Priority mechanism.

allocated riders, the new rider should pay compensation to each of them. The new rider should cover any extra travelling costs deriving from the re-routing. We present the following example to build intuition on how the routes are proposed.

Consider the simple case with two riders who need to move on a grid. The grid is served by two car depots located at locations $(0,0)$ and $(10,10)$. Adjacent locations in the grid are connected with vertical and horizontal roads, traversable by a car in a single unit of time. All cars have identical costs and capacities, and all riders have the same VoT.

A rider appears at time $t = 0$ in location $(2,2)$ and wants to move to location $(5,7)$. The rider will then receive two proposals: A car from the first depot would propose the route $(0,0) \to (2,2) \to (5,7)$, for 12 pounds, and a projected finishing time at 12. A car from the second depot will similarly propose to the rider the route $(10,10) \to (2,2) \to (5,7)$ for 24 pounds. The ideal route length for the rider is $\tau_1^* = 8$; at time 0, no car is moving, and the only available options for the rider are cars not in use from the two depots. This implies that rider 1 has a utility of $-(12-8)-12 = -16$ for the first proposal, and a utility of $-(24-8)-24 = -40$ for the second proposal. Hence, the first car is selected and starts its trip, with the first stop at $(2,2)$.

At time $t = 5$, a new rider, say rider 2, appears at location $(6,5)$ and wants to travel to location $(2,6)$, with an ideal cost of 5 units of time. The first car has now arrived in location $(3,2)$, with rider 1 embarked (at time 4, from location $(2,2)$). The travelling car now has three available options to propose to the new rider. This situation is depicted in Figure 1.

Option 1: the path $(3,2) \to (6,5) \to (2,6) \to (5,7)$. This route firstly serves rider 2, and then drops-off rider 1. The new route will conclude at time $t = 20$, while the car's current route is expected to finish at time $t = 12$. Rider 2 needs to cover this difference, and must also compensate rider 1 for all the extra time it needs to travel. Indeed, rider 1 is projected to finish at time $t = 20$ while the previous route had proposed a finish for rider 1 at time $t = 12$. Hence, rider 2 should pay a price of $8 + 8 = 16$ pounds, and since rider 2 needs 10 units of time to reach its destination, its projected utility is $-(10-5)-(8+8) = -21$.

Option 2: The path $(3,2) \to (6,5) \to (5,7) \to (2,6)$. This route picks up rider 2, then moves to drop off rider 1, then drops rider 2. This route finishes at time $t = 18$, and rider 1 needs to travel for 2 more units of time than the current route. Thus rider's 2 payment is $6 + 2 = 4$ pounds. Rider 2 gets a projected finishing time of 13 units of time from its appearance. This implies a projected utility of $-(13-5)-(6+2) = -16$.

Option 3: Finally, the path $(3,2) \to (5,7) \to (6,5) \to (2,6)$. In this route, rider 1 is first dropped off; then the second rider is served. Rider 2 pays 8 pounds in total for the part $(5,7) \to (6,5) \to (2,6)$ and no compensation is needed. Under this route, rider 2 finishes 15 units of time after its appearance, which implies a utility of -18.

The mechanism proposes the second route to rider 2, as it is the best route from rider's 2 perspective (shown with a solid line in Figure 1). Alongside this proposal, rider 2 also gets proposals from the cars parked in the two depots: one passing through $(0,0) \to (6,5) \to (2,6)$, with a utility of -27, and one using the path $(10,10) \to (6,5) \to (2,6)$ for a utility of -21. Hence, the rider will eventually choose to travel alongside rider 1.

Algorithm 3 Vehicles' proposal for Compensation.

input: rider i, current route $\mathbf{r}$ at time t, set of current riders R_j.
output: new route $\hat{\mathbf{r}}$, payment $p^{i,j}$.
procedure ComputeProposals(i, j, t)
    for $\hat{\mathbf{r}} \in$ RouteGenerator$(i, j, \mathbf{r})$ **do**
        for $k \in R_j$ **do**
            $\hat{h}_k^{\hat{\mathbf{r}}} \leftarrow \max(0, u_k(\hat{\mathbf{r}}, p^{k,j}) - u_k(\mathbf{r}, p^{k,j}))$
        end for
        $\hat{p}^{\hat{\mathbf{r}}} \leftarrow c_j \cdot \max(0, \phi(\hat{\mathbf{r}}) - \phi(\mathbf{r})) + \sum_{k \in R_j} \hat{h}_k^{\hat{\mathbf{r}}}$
    end for
    $(\mathbf{r}^{j,t}, p^{i,j}) \leftarrow \arg\max_{(\hat{\mathbf{r}}, \hat{p}^{\hat{\mathbf{r}}})} (u_i(\hat{\mathbf{r}}, \hat{p}^{\hat{\mathbf{r}}}))$
end procedure

5 FORMAL PROPERTIES AND ANALYSIS

In this section, we analyse the performance of the Compensation mechanism. The first property formally shows the main idea of the mechanism: that each rider is guaranteed not to lose on their utility because of the other riders.

Theorem 5.1. *For the Compensation mechanism, the ex-post utility is always higher than the projected utility for all riders.*

Proof. First, observe that the utility of rider i can change (compared to the (promised) projected utility by the mechanism) only when a new rider is allocated in the car after rider's i allocation. For the sake of contradiction, assume that some rider k exists for which the utility of rider's i decreases when k is allocated to the same car, say j, at time t. Let ϕ_i and p_i denote the finishing time and the proposed payment for the already allocated rider i at time t, *before* the allocation of rider k. Let also ϕ_i' denote the finishing time for rider i, *after* the allocation of rider k. Furthermore, let R be the set of allocated riders in car j, allocated after rider i, but before rider k. Hence, the difference in the rider's i utility, after and before the allocation of k equals to:

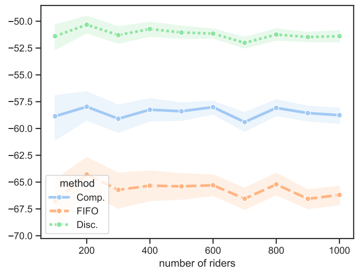
$$- v_i \cdot (\phi_i' - \tau_i^*) - p_i + \sum_{l \in R \cup \{k\}} h_l - \left(-v_i \cdot (\phi_i - \tau_i^*) - p_i + \sum_{l \in R} h_l \right)$$
$$= v_i \cdot (\phi_i - \phi_i') + h_k.$$

When $\phi_i - \phi_i' < 0$, observe that h_k is exactly equal to $v_i \cdot (\phi_i - \phi_i')$, hence the theorem holds. When $\phi_i - \phi_i' \geq 0$, i.e., when rider i finishes earlier, the theorem holds trivially since the compensation h_k received is never negative. □

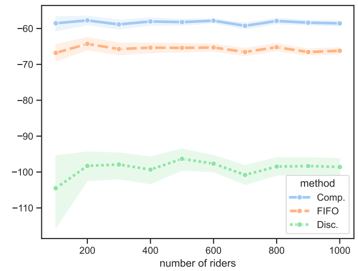
In the next theorem, we show that using the Compensation mechanism, a myopic new rider has no incentive to manipulate the mechanism.

Theorem 5.2. *The Compensation mechanism is DSIC for myopic riders.*

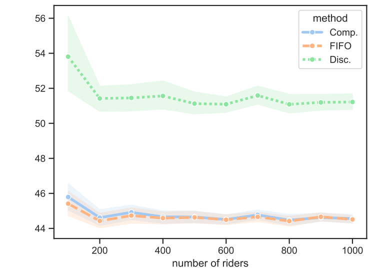
The full proof of the theorem is presented in Appendix B. Here, we briefly explain the high-level intuition. The main idea is to associate any route proposed by a car when a rider acts non-truthfully with a different route, grounded on the riders' true valuations. This new route either provides a better utility for the rider or generates the proper compensation to make this new route a better choice.



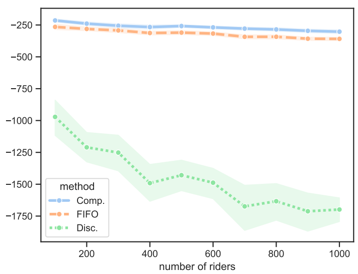
(a) mean proposed utility.



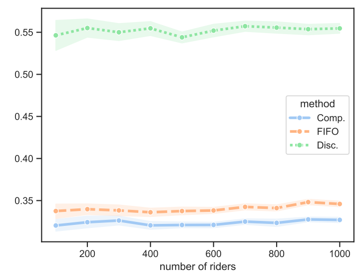
(b) mean ex-post utility.



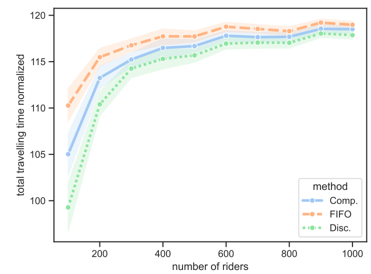
(c) mean ex-post commute time.



(d) minimum ex-post utility.



(e) Gini ex-post utility.



(f) total travelling time.

Figure 2: Comparison of the Compensation (solid, blue), FIFO (dashed, orange) and Discount(dotted, green) mechanisms. Simulations are performed a 50×50 grid, with average time between arrivals $\lambda = 5$, VoT between $B_L = 1$ and $B_U = 10$, and at most 10 identical cars departing from 2 depots. Results are averaged over 20 iterations showing 95% confidence intervals.

Note also that this mechanism is no longer truthful in the ex-post sense. This can be proven using the following counter-example:

Example 5.3. Consider the line graph $(0, 1, ..., k)$, with unit cost on the edges, a single depot at 0 and one available car at the depot. Rider i, who needs to get from location 1 to k, *knows* that another rider, namely i', will appear at time $t_{i'} = t_i + 1$, at location 2, and also needs to go to the same destination, k. When rider i reports truthfully, she will be allocated in the single available car, with the route $\mathbf{r}^{j,t_i} = (0, 1, k)$. Since this is the direct route, the utility of rider i is equal to $-(k + 1)$, i.e., the cost of the ride. This is, in fact, the ex-post utility of rider i. When rider i' appears, the car will propose the route $\mathbf{r}^{j,t_{i'}} = (1, 2, k)$. Note that the total payment for rider i' is 0 as the route cost is already covered by rider i, and no compensation is needed for rider i since the rider does not lose utility with the new ride.

When rider i reports untruthfully $t'_i = t_{i'} + 1$, then the car will first assign rider i' with the route $\mathbf{r}^{j,t_{i'}} = (0, 2, k + 1)$. In the following time-step, when rider i' appears, then route will change to $\mathbf{r}^{j,t'_i} = (3, 1, k)$. This route needs $k + 2$ time-steps to finish, 2 time-steps longer than the previous route. The rider also inflicts an extra cost of $2 \cdot v_{i'}$ to rider i, for which she should compensate her. Hence the payment for rider i becomes $2 \cdot (1 + v_{i'})$. As a consequence, the utility is now $-2 \cdot v_i - 2 \cdot (1 + v_{i'})$, which is at least equal to $-(k + 1)$ with specific values for k, v_i and $v_{i'}$. This holds for $v_i = v_{i'} = 1$ and any $k > 6$.

Algorithm 4 Route Generator.

input: potential rider i, current route $\mathbf{r}$ at time t, set of current riders R_j.
output: Set of new routes P
procedure RouteGenerator($i, j, \mathbf{r}$)
  $\ell = |\mathbf{r}| + 2$ ▷ set the length of the new routes
  for $k \in \{0, ..., \ell\}$ **do**
    for $l \in \{0, ..., \ell\}$ **do**
      $\hat{r}_0 = r_0$ ▷ set the car location
      $\hat{\mathbf{r}}_{1,...,k-1} = \mathbf{r}_{1,...,k-1}$
      $\hat{\mathbf{r}}_k = a_i$ ▷ Add pick-up location
      $\hat{\mathbf{r}}_{k+1,...,l-1} = \mathbf{r}_{k,...,l}$
      $\hat{\mathbf{r}}_l = b_i$ ▷ Add drop-off location
      $\hat{\mathbf{r}}_{l+1...\ell} = \mathbf{r}_{l,...,|\mathbf{r}|}$
      $P \longleftarrow \hat{\mathbf{r}}$
    end for
  end for
end procedure

We also note that the Compensation mechanism is Budget-Balanced, i.e., the travelling cost of the riders is always covered by the payments.

Theorem 5.4. *The Compensation mechanism is Budget-Balanced.*

Proof. This is trivially derived by line 8 in Algorithm 3, which states that any new rider will cover the travelling cost of the car for the whole route. Note that the computed compensations are not needed to cover the travelling cost. □

Finally, we discuss the computational traceability of our proposal. The most crucial part of the algorithm is the RouteGenarator procedure, which computes valid routes for the car to accommodate a new rider. The number of these routes is quite large, exponential to the number of allocated riders, in general. To cope with this, we use a restricted class of routes, which respects the order of the previous riders, generated as follows: given a previous order of size k, the pick-up location index is $i \in \{1, k+1\}$ in the new order. Then the drop-off locations is $j \in \{2, k+2\} \setminus \{i\}$. The remaining k slots are filled according to the previous order. This leaves us with substantially fewer possible orders to explore, squared in the number of allocated riders. The pseudocode for the described implementation is presented in Algorithm 4. An interesting open problem is to quantify how this restriction affects the efficiency of the algorithm and to review our mechanism with more sophisticated route recommendation methods (see, e.g., [36]).

6 EXPERIMENTAL EVALUATION

This section evaluates the performance of our mechanism using a series of simulations.

6.1 Experimental Setting

In our experiments we use a simple k-grid graph where each node $(i, j) \in [k]^2$ is connected to its (at most four) neighbours $(i+1, j)$, $(i-1, j)$, $(i, j+1)$ and $(i, j-1)$, whenever $i \pm 1$ and $j \pm 1$. Each edge is traversed in a single time step. The graph is created using the *NetworkX Python* library [16], and shortest paths are computed using the well-known A^* algorithm [17], using the library's native functions. In each experiment, we use a fixed number of depots whose locations in each iteration are drawn uniformly over the graph without replacements. Whenever a new rider requests a ride, it receives quotes from all live vehicles (vehicles moving in the grid) and one vehicle from each depot.

The time between arrivals is simulated by a Poisson distribution with mean λ. The riders' VoT is uniformly distributed in the interval $[B_L, B_U]$, $B_U \geq B_L \geq 0$. Finally, riders' origin and destinations are uniformly and independently distributed [8]. The cost of the Compensation and FIFO mechanisms routes is equal to their length, multiplied by the car cost per unit of time. For the Discount mechanism, the payment for a new passenger is discounted by 10% when a rider is allocated in the car with 1 other allocated riders on board, and 20% when a rider is allocated in a car with 2 passengers and more.

We evaluate our mechanisms according to various criteria. First, in terms of efficiency, we compare our mechanism according to the (mean) ex-post utility, the utility of the riders when they finish their rider, and the (mean) projected utility, the utility the riders are "offered", at the time of allocation. In terms of fairness, we evaluate the mechanism using two metrics. The first is the minimum ex-post utility, which rates a mechanism based on the least well-treated rider. While this is a natural fairness desideratum, it has limitations and might be affected solely by outliers. Hence, we complement it by using our second fairness measure, the Gini index (see, e.g., [24]). The Gini index is an expressive measure for evaluating the equity of the service and its fairness in the transportation literature [33]. A Gini index of 0 implies total equality, i.e., that the ex-post utility of all riders is equal. On the contrary, a Gini index of 1 implies total inequality, i.e., that a single rider has a very high utility, while the utility of all other riders is very low. Finally, to measure the environmental impact of our proposal, we compare the total travelling time of all cars, including the time needed to return to their designated depot.

Experiments: We have evaluated the mechanism using different experiments. Additional experiments are presented in the extended version of the paper. In each experiment, we randomly create 20×10 groups of riders, with different parameters. The first 20 groups include 100 riders, and the size gradually increases to 1000. We execute the three mechanisms proposed for each group: FIFO, Discount, and Compensation. In all experiments, we use a 50×50 grid. In the first experiment, we use at most 10 identical vehicles, with cost equal to 1 and capacity equal to 4. The mean time difference between arrivals is set to $\lambda = 5$ units of time, the VoT is selected between $B_L = 1$ and $B_u = 10$, and the vehicles depart from three depots. The results are presented in Figure 2. In Figure 3, we present the results from a slightly different setting, where riders' arrivals are denser ($\lambda = 2$), and the mean VoT is higher ($B_L = 1, B_U = 20$), i.e., in expectation, riders are more sensitive in delays, compared to the optimal route. To compensate for this, we allow more vehicles (20 cars are available). The vehicles depart from two depots.

6.2 Results and Analysis

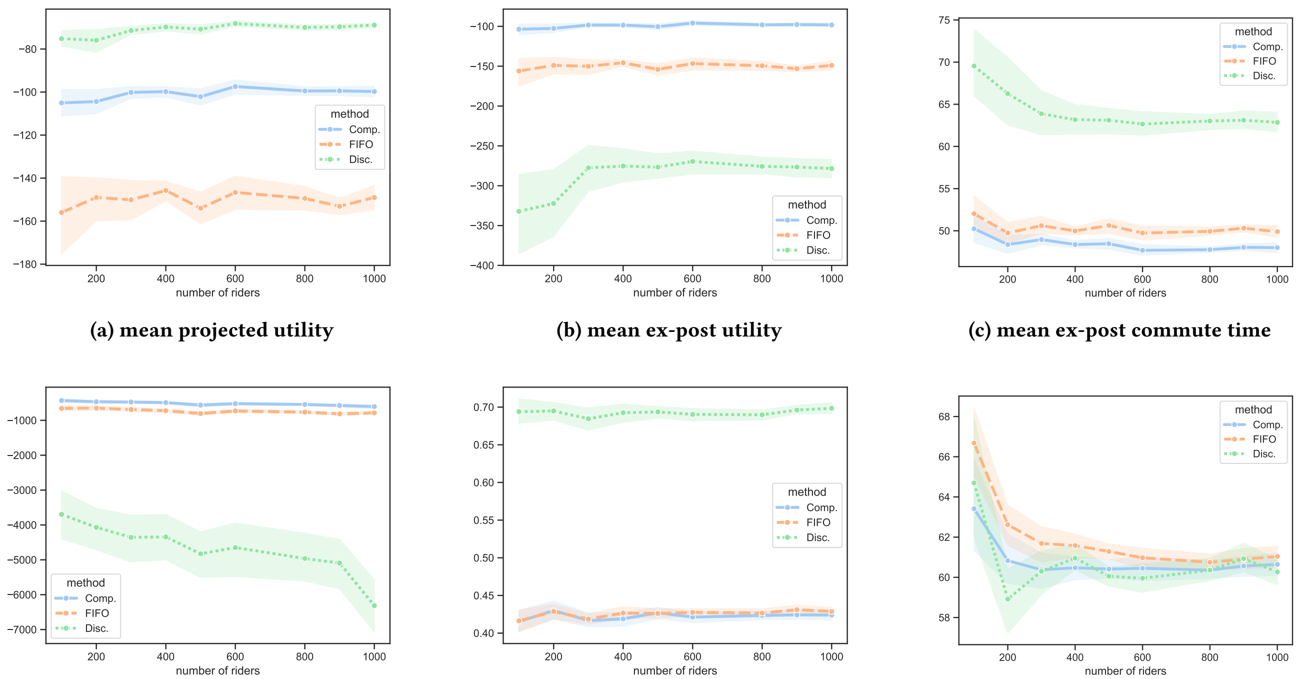
This section briefly analyses our experiments regarding efficiency, fairness and environmental impact.

Efficiency: Apart from guaranteeing the riders their projected utility at the allocation, the Compensation mechanism yields approximately 14% higher ex-post utility compared to the second-best choice, the FIFO strategy in the first experiment. This margin increases to approximately 50% in the second experiment. In contrast, the Discount strategy promises the highest utility at allocation time. Still, it fails to deliver on these promises at the end of the journey. Observe here that the identical graphs for the FIFO mechanism in Figures 2a and 2b (and in 3a and 3b) are not coincident since this mechanism yields the same utility at allocation and after the end of the journey. The Compensation and FIFO mechanisms perform best in commute time, approximately 20% better than the Discount policy in the first experiment and 40% better in the second, where the Compensation mechanism performs slightly better. This illustrates that the good performance of the Compensation mechanism in terms of utility does not happen because a large amount of compensation is received, but it occurs organically.

Fairness: Regarding fairness, the Compensation and FIFO mechanisms also cope quite well in all experiments. The Compensation mechanism performs slightly better in both experiments on the rather pessimistic criterion of minimum ex-post utility. This trend also continues with respect to the Gini Index.

Enviromental Impact: Our final measure compares the mechanisms for their total travelling time, normalised by the number of

[8] In the case of origin and destination, we slightly abuse independence by enforcing different origins and destinations.



(a) mean projected utility **(b) mean ex-post utility** **(c) mean ex-post commute time**

(d) minimum ex-post utility **(e) Gini ex-post utility** **(f) total travelling time**

Figure 3: Comparison of the Compensation (solid, blue), FIFO (dashed, orange), and Discount(dotted, green) mechanisms. Simulations are performed on a 50×50 grid, with average time between arrivals $\lambda = 2$, VoT between $B_L = 1$ and $B_U = 20$, and at most 20 identical vehicles departing from 2 depots. Results are averaged over 20 iterations, showing 95% confidence intervals.

rides. Due to the unit cost of the vehicles, this is also equal to the total operational cost of the system. The Discount method yields the minimum total cost in the first experiment, while the other two mechanisms follow shortly. Interestingly, as the number of riders increases, the total cost per rider approximates a constant. In the second, the total travelling time decreases for all mechanisms, with the Compensation mechanism being slightly more efficient on average, although the confidence intervals for all mechanisms significantly overlap.

7 CONCLUSIONS

We have proposed a model and mechanisms for online ridesharing, focusing on the riders' incentives. Our model focuses on riders with on-demand requests, in the sense that they can appear at unknown times and request a ride that should be served as soon as possible. Our model is decentralised in the sense that riders receive quotes from available cars, and they choose one of them to complete their trip. Our main contributions are the Compensation mechanism and the approach to present a set of options to riders. The Compensation mechanism is truthful for myopic riders, budget-balanced and overruns the two other methods in most experiments, and it is never critically overrun by any of the benchmarks. An interesting open problem regarding our approach is whether meaningful truthful mechanisms for non-myopic riders exist in the model-free setting. We believe that only mechanisms similar to FIFO can be truthful under this demanding environment, and we aim to show this in future work. We also note that to avoid concurrency issues, we have assumed that every single rider appears at each time step, which is, of course, non-realistic. To safely drop this assumption, our work could be enhanced by being analysed under the glance of distributed mechanism design (see e.g., [12]).

ACKNOWLEDGMENTS

The authors acknowledge the use of the IRIDIS High Performance Computing Facility and associated support services at the University of Southampton in completing this work. This work was supported by the UK Engineering and Physical Sciences Research Council (EPSRC) through a Turing AI Fellowship (EP/V022067/1) on Citizen-Centric AI Systems (https://ccais.ac.uk) and the platform grant entitled "AutoTrust: Designing a Human-Centered Trusted, Secure, Intelligent and Usable Internet of Vehicles" (EP/R029563/1). This work has been partially supported by project MIS 5154714 of the National Recovery and Resilience Plan Greece 2.0 funded by the European Union under the NextGenerationEU Program.

ETHICAL STATEMENT

Our proposal for dynamic ridesharing caters to the preferences of users by allowing a diverse range of individuals with varying needs to choose from ridesharing options and prices, as opposed to imposing a single fixed price by service providers. Furthermore, our approach provides flexibility for users to join the service dynamically, allowing them to join during its operational hours. We also advocate for accessibility and equity in mobility services by introducing a fair and manipulation-resistant method for efficient ridesharing. This method aims to optimise total travel time, thereby supporting the sustainability of mobility systems and efforts to mitigate climate change.

REFERENCES

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

[2] Negin Alisoltani, Ludovic Leclercq, and Mahdi Zargayouna. 2021. Can dynamic ride-sharing reduce traffic congestion? *Transportation Research Part B: Methodological* 145 (March 2021), 212–246.

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

[4] Mohammad Asghari and Cyrus Shahabi. 2017. An On-line Truthful and Individually Rational Pricing Mechanism for Ride-sharing. In *Proceedings of the 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS 2017*. Association for Computing Machinery, New York, NY, USA, 7:1–7:10.

[5] Xiaohui Bei and Shengyu Zhang. 2018. Algorithms for Trip-Vehicle Assignment in Ride-Sharing. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018)*. AAAI Press, Washington, DC, USA, 3–9.

[6] Zheyong Bian, Xiang Liu, and Yun Bai. 2020. Mechanism design for on-demand first-mile ridesharing. *Transportation Research Part B: Methodological* 138 (Aug. 2020), 77–117.

[7] Filippo Bistaffa, Christian Blum, Jesús Cerquides, Alessandro Farinelli, and Juan A Rodríguez-Aguilar. 2019. A computational approach to quantify the benefits of ridesharing for policy makers and travellers. *IEEE Transactions on Intelligent Transportation Systems* 22, 1 (2019), 119–130.

[8] Arpita Biswas, Ragavendran Gopalakrishnan, Theja Tulabandhula, Koyel Mukherjee, Asmita Metrewar, and Thangaraj Rajasubramaniam. 2017. Profit Optimization in Commercial Ridesharing. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems (AAMAS '17)*. IFAAMAS, Richland, SC, USA, 1481–1483.

[9] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia. 2016. *Handbook of computational social choice.* Cambridge University Press, Cambridge.

[10] Brian Caulfield. 2009. Estimating the environmental benefits of ride-sharing: A case study of Dublin. *Transportation Research Part D: Transport and Environment* 14, 7 (2009), 527–531.

[11] Shih-Fen Cheng, Duc Thien Nguyen, and Hoong Chuin Lau. 2014. Mechanisms for Arranging Ride Sharing and Fare Splitting for Last-Mile Travel Demands. In *International conference on Autonomous Agents and Multi-Agent Systems (AAMAS '14)*. IFAAMAS, Richland, SC, USA, 1505–1506.

[12] Joan Feigenbaum, Michael Schapira, and Scott Shenker. 2007. Distributed Algorithmic Mechanism Design. In *Algorithmic Game Theory*. Vol. 14. University Press Cambridge, Cambridge, 363–384.

[13] Masabumi Furuhata, Kenny Daniel, Sven Koenig, Fernando Ordonez, Maged Dessouky, Marc-Etienne Brunet, Liron Cohen, and Xiaoqing Wang. 2014. Online cost-sharing mechanism design for demand-responsive transport systems. *IEEE Transactions on Intelligent Transportation Systems* 16, 2 (2014), 692–707.

[14] Masabumi Furuhata, Maged Dessouky, Fernando Ordóñez, Marc-Etienne Brunet, Xiaoqing Wang, and Sven Koenig. 2013. Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological* 57 (2013), 28–46.

[15] Yuhan Guo, Yu Zhang, Youssef Boulaksil, Yaguan Qian, and Hamid Allaoui. 2023. Modelling and analysis of online ride-sharing platforms – A sustainability perspective. *European Journal of Operational Research* 304, 2 (Jan. 2023), 577–595.

[16] Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. 2008. Exploring Network Structure, Dynamics, and Function using NetworkX. In *Proceedings of the 7th Python in Science Conference*. Pasadena, CA, USA, 11 – 15.

[17] Peter E Hart, Nils J Nilsson, and Bertram Raphael. 1968. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics* 4, 2 (1968), 100–107.

[18] Shichun Hu, Maged M Dessouky, Nelson A Uhan, and Phebe Vayanos. 2021. Cost-sharing mechanism design for ride-sharing. *Transportation Research Part B: Methodological* 150 (2021), 410–434.

[19] Tatsuya Iwase, Sebastian Stein, and Enrico H. Gerding. 2021. A Polynomial-time, Truthful, Individually Rational and Budget Balanced Ridesharing Mechanism. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)*. International Joint Conferences on Artificial Intelligence Organization, 268–275.

[20] Renos Karamanis, Eleftherios Anastasiadis, Panagiotis Angeloudis, and Marc Stettler. 2020. Assignment and pricing of shared rides in ride-sourcing using combinatorial double auctions. *IEEE Transactions on Intelligent Transportation Systems* 22, 9 (2020), 5648–5659.

[21] Alexander Kleiner, Bernhard Nebel, and Vittorio A. Ziparo. 2011. A Mechanism for Dynamic Ride Sharing Based on Parallel Auctions. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*. AAAI Press, Washington, DC, USA, 266–272.

[22] Nixie S. Lesmana, Xuan Zhang, and Xiaohui Bei. 2019. Balancing Efficiency and Fairness in On-Demand Ridesourcing. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019*. Curran Associates, Inc., 5310–5320.

[23] Mengya Liu, Vahid Yazdanpanah, Sebastian Stein, and Enrico Gerding. 2024. Sustainability-oriented route generation for ridesharing services. *Computer Science and Information Systems* 21, 1 (2024), 309–33.

[24] Hervé Moulin. 1988. *Axioms of Cooperative Decision Making*. Cambridge University Press, Cambridge.

[25] Abood Mourad, Jakob Puchinger, and Chengbin Chu. 2019. A survey of models and algorithms for optimizing shared mobility. *Transportation Research Part B: Methodological* 123 (2019), 323–346.

[26] Noam Nisan. 2007. Introduction to mechanism design (for computer scientists). In *Algorithmic Game Theory*. Cambridge University Press, Cambridge, Chapter 9, 209–242.

[27] Fernando Ordóñez and Maged M. Dessouky. 2017. Dynamic ridesharing. In *INFORMS TutORials in Operations Research*. INFORMS, 212–236.

[28] Leonardo Y. Schwarzstein and Rafael C.S. Schouery. 2023. Budget-balanced and strategy-proof auctions for ridesharing. *Computers & Operations Research* 151 (2023), 106094.

[29] Wen Shen, Cristina V. Lopes, and Jacob W. Crandall. 2016. An Online Mechanism for Ridesharing in Autonomous Mobility-on-Demand Systems. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI 2016)*. IJCAI/AAAI Press, Washington, DC, USA, 475–481.

[30] Amirmahdi Tafreshian and Neda Masoud. 2022. A truthful subsidy scheme for a peer-to-peer ridesharing market with incomplete information. *Transportation Research Part B: Methodological* 162 (2022), 130–161.

[31] UN DESA. 2021. *The Sustainable Development Goals Report 2021*. UN DESA, New York, USA.

[32] Haoning Xi, Wei Liu, S. Travis Waller, David A. Hensher, Philip Kilby, and David Rey. 2023. Incentive-compatible mechanisms for online resource allocation in Mobility-as-a-Service systems. *Transportation Research Part B: Methodological* 170 (2023), 119–147.

[33] An Yan and Bill Howe. 2020. Fairness in practice: a survey on equity in urban mobility. *A Quarterly bulletin of the Computer Society of the IEEE Technical Committee on Data Engineering* 42, 3 (2020), 49–63.

[34] Pengyu Yan, Chung-Yee Lee, Chengbin Chu, Cynthia Chen, and Zhiqin Luo. 2021. Matching and pricing in ride-sharing: Optimality, stability, and financial sustainability. *Omega* 102 (2021), 102351.

[35] Biying Yu, Ye Ma, Meimei Xue, Baojun Tang, Bin Wang, Jinyue Yan, and Yi-Ming Wei. 2017. Environmental Benefits from Ridesharing: A Case of Beijing. *Applied Energy* 191 (2017), 141–152.

[36] Chak Fai Yuen, Abhishek Pratap Singh, Sagar Goyal, Sayan Ranu, and Amitabha Bagchi. 2019. Beyond Shortest Paths: Route Recommendations for Ride-Sharing. In *The World Wide Web Conference (WWW '19)*. Association for Computing Machinery, New York, NY, USA, 2258–2269.

[37] Chaoli Zhang, Jiapeng Xie, Fan Wu, Xiaofeng Gao, and Guihai Chen. 2020. Pricing and Allocation Algorithm Designs in Dynamic Ridesharing System. *Theoretical Computer Science* 803 (2020), 94–104.

[38] Boming Zhao, Pan Xu, Yexuan Shi, Yongxin Tong, Zimu Zhou, and Yuxiang Zeng. 2019. Preference-Aware Task Assignment in On-Demand Taxi Dispatching: An Online Stable Matching Approach. In *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI 2019)*. AAAI Press, Washington, DC, USA, 2245–2252.

[39] Dengji Zhao, Dongmo Zhang, Enrico H. Gerding, Yuko Sakurai, and Makoto Yokoo. 2014. Incentives in Ridesharing with Deficit Control. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS '14)*. IFAAMAS, Richland, SC, USA, 1021–1028.

A OMITTED PROOF FROM SECTION 3

In this section, we formally discuss the two benchmark mechanisms.

Theorem A.1. *The FIFO mechanism is DISC and Budget-Balanced.*

Proof. Fix a sequence of rider appearances $t_1, ..., t_n$, and assume, for the sake of contradiction, that there exists some rider i which can misreport either its starting time t_i with some t_i' or its value of time v_i' and, as a result, receives a higher projected utility. We will tackle both parameters separately. First, we note that the value of time v_i does not affect the proposed route and prices for the ride. Hence, we can solely focus on misreporting on time, and we assume that a rider proposes to the mechanism not its true arrival time, but some value $t_i' > t_i$, i.e., the rider deliberately delays her arrival.

Consider any two finishing times $\phi_i := \tau(\hat{\mathbf{r}}) + t_i$ and $\phi_i' := \tau(\hat{\mathbf{r}}') + t_i'$. Then rider i has an incentive to propose t_i', rather its true appearance time t_i. Then it must hold that

$$-v_i \cdot (\phi_i' - \phi_i^*) - c_j \cdot \phi_i' > -v_i \cdot (\phi_i - \phi_i^*) - c_j \cdot \phi_i. \quad (1)$$

With some simple algebraic operations, we can see that this is true when $\tau(\hat{\mathbf{r}}') + t_i' = \phi' < \phi = \tau(\hat{\mathbf{r}}) + t_i$. Note, however, that this is not possible: The FIFO mechanism respects the appearance times of the riders' hence, by appearing later, rider i cannot decrease her finishing time. Hence, the FIFO mechanism is DISC.

The mechanism is Budget-Balanced since any new rider covers the total cost of the new route. □

B PROOF OF THEOREM 4.2

Proof. Assume, for the sake of contradiction, that there exists some rider i which can misreport either its starting time t_i with some t_i' or its value of time v_i' and, as a result, receives a higher projected utility. We will tackle both parameters separately.

First, we note that the value of time v_i does not affect the proposed route and prices for the ride: Indeed, the computation of the payment does not involve v_i, and in the final maximisation operator, the maximising route and payment pair is not affected by v_i.

Then, let that a rider provides to the mechanism not its true arrival time but some value $t_i' > t_i$, i.e., the rider deliberately delays her arrival. Let that $\hat{\mathbf{r}}' := \hat{\mathbf{r}}^{j,t_i'}$ be the proposed route for a given car j from Algorithm 3. We will show that there exists some other route $\hat{\mathbf{r}} := \hat{\mathbf{r}}^{j,t_i}$, proposed when rider i reports its true arrival time for which:

$$v_i \cdot (\phi_i(\hat{\mathbf{r}}) - \phi_i(\hat{\mathbf{r}}')) \leq \hat{p}^{\hat{\mathbf{r}}'} - \hat{p}^{\hat{\mathbf{r}}}. \quad (2)$$

This suffices to show that rider's i utility when reporting t_i is at least as good as when reporting t_i'. Intuitively, the weighted *gain in time*, $\phi_i(\hat{\mathbf{r}}) - \phi_i(\hat{\mathbf{r}}')$ should be reflected by any *loss in payment* (and vice versa).

Let $\mathbf{r} := \mathbf{r}^{j,t_i-1} = (\ell_{j,t_i-1}, \mathbf{y})$ be the route of the car immediately before the arrival of rider i. The car needs to move in all locations described by the order $\mathbf{y}$. Consider now the route $\hat{\mathbf{r}}' = (\ell_{j,t_i'}, \mathbf{q}')$, which consists of the car location $\ell_{j,t_i'}$ at time t_i', and the order $\mathbf{q}'$. Note that $\mathbf{q}'$ consists of the pick up and drop off locations for i, a_i and b_i, and some of the locations in $\mathbf{y}$.[9]

From $\hat{\mathbf{r}}'$ we will construct $\hat{\mathbf{r}} = (\ell_{j,t_i}, \mathbf{q})$. The key of this proof is the construction of the order $\mathbf{q}$ for the order $\mathbf{q}'$: while $\mathbf{q}'$ must only include a_i and b_i, $\mathbf{q}$ must also include all locations in $\mathbf{y}$. We will construct order $\mathbf{q}$ by expanding $\mathbf{q}'$, as follows, using a rather unusual but convenient numbering: For a given $\mathbf{q}' = (\mathbf{q}_1', ..., q_{|q|-z}) = (q_z, ..., q_{|q|})$, we construct the order $\mathbf{q} = (q_1, ..., q_{z-1}, q_z, ..., q_{|\mathbf{q}|})$. Note that all locations $q_1, ..., q_{z-1}$ should be visited before t_i'.

For a given order $\mathbf{o}$, we define the function $\sigma_i(\mathbf{o})$, which returns the order's $\mathbf{o}$ index for b_i, i.e., $\mathbf{o}_{\sigma_i(\mathbf{o})} = b_i$. Furthermore, let $\tau_i(\mathbf{o}) := \sum_{k=1}^{\sigma_i(\mathbf{o})-1} \delta(o_k, o_{k+1})$ and $\tau(\mathbf{o}) := \sum_{k=1}^{|\mathbf{o}|-1} \delta(o_k, o_{k+1})$. Using these definitions and the rationale explained in the previous paragraphs, observe that $\tau(\mathbf{q}) - \tau(\mathbf{q}') = \tau_k(\mathbf{q}) - \tau_k(\mathbf{q}') = t_i' - t_i$ for all $k \in R_{t_i} \cup R_{t_i'}$ and $\tau_k(\mathbf{q}) - \tau_k(\mathbf{q}') \leq t_i' - t_i$ for all $k \in R_{t_i} \setminus R_{t_i'}$.

This observation supports proving the following inequality:

$$\begin{aligned}\phi_k(\hat{\mathbf{r}}') - \phi_k(\hat{\mathbf{r}}) &= \delta(\ell_{j,t_i'}, q_1') - \delta(\ell_{j,t_i}, q_1) + (t_i' - t_i) \\ &\quad - \tau_i(\mathbf{q}) + \tau_i(\mathbf{q}') \\ &\geq \delta(\ell_{j,t_i'}, q_1') - \delta(\ell_{j,t_i}, q_1) =: \Delta \end{aligned} \quad (3)$$

for any rider $k \in R_{t_i}$. The final inequality is due to $\tau_i(\mathbf{q}) - \tau_i(\mathbf{q}') \leq t_i' - t_i$, and in particular $\phi_k(\hat{\mathbf{r}}') - \phi_k(\hat{\mathbf{r}}) = \Delta$ for $k \in R_{t_i} \cup R_{t_i'}$, and $\phi(\hat{\mathbf{r}}') - \phi(\hat{\mathbf{r}}) = \Delta$.

Inequality 3 implies that the LHS of 2 is upper bounded by $-\Delta$, i.e.,

$$\phi_i(\hat{\mathbf{r}}') - \phi_i(\hat{\mathbf{r}}) \leq -\Delta. \quad (4)$$

We then turn our attention to the RHS. The payments can be split into the compensation and the cost-covering parts. We start with the compensation part.

$$\begin{aligned}\sum_{k \in R_{t_j'}} h_k^{\hat{\mathbf{r}}'} - \sum_{k \in R_{t_j}} h_k^{\hat{\mathbf{r}}} &= \sum_{k \in R_{t_j} \cap R_{t_j'}} v_k \cdot (\phi_k(\hat{\mathbf{r}}') - \phi_k(\hat{\mathbf{r}})) \\ &\quad + \sum_{k \in R_{t_j} \setminus R_{t_j'}} \max(0, v_k \cdot (\phi_k(\hat{\mathbf{r}}) - \phi_k(\mathbf{r}))) \\ &\geq \Delta \cdot \sum_{k \in R_{t_j} \cap R_{t_j'}} v_k \end{aligned} \quad (5)$$

Finally, note that if $\phi(\hat{\mathbf{r}}') - \phi(\mathbf{r}) > 0$, then $\phi(\hat{\mathbf{r}}) - \phi(\mathbf{r}) > 0$. Then,

$$\begin{aligned}p^{\hat{\mathbf{r}}'} - p^{\hat{\mathbf{r}}} &= \phi(\hat{\mathbf{r}}') - \phi(\hat{\mathbf{r}}) + \sum_{k \in R_{t_j'}} h_k^{\hat{\mathbf{r}}'} - \sum_{k \in R_{t_j}} h_k^{\hat{\mathbf{r}}} \\ &\geq \Delta + \Delta \sum_{k \in R_{t_j} \cap R_{t_j'}} v_k. \end{aligned} \quad (6)$$

The theorem follows by observing that inequalities 4 and 6 imply that inequality 3 always holds. □

[9] It is possible that it includes none up to all locations in $\mathbf{y}$, with any possible permutations.

C EXPERIMENTAL SETTING: FURTHER DISCUSSION

In this section, we provide some extra information on our experiments.

Simulation process: The simulation process is described below. At first, we create k groups of driver types for a given number of riders. Then, each group of riders is used as input to simulate the three mechanisms described in Sections 3 and 4.

Given a fixed number of riders $n \in \mathbb{N}$ and parameters λ, B_L, B_U and X, Y we sample the parameters for each rider $i \in [n]$ in the group as:

- arrival time $1 + t_i$ for $t_i \in \text{Poisson}(\lambda)$,
- pick-up location $a_i \in \{0, ..., X\} \times \{0, ..., Y\}$, selected uniformly at random.
- drop-off location $b_i \in \{\{0, ..., X\} \times \{0, ..., Y\}\} \setminus \{a_i\}$ selected uniformly at random, and
- $v_i \in [B_L, B_U]$, selected uniformly at random.

Let us explain some of the decisions for the above sampling. First, we do not allow more than 1 rider in the same time slot: while this is an unrealistic assumption, more than 1 rider in a single time slot poses additional challenges beyond the scope of this paper. Second, we use the Poisson distribution to model user arrivals in queuing systems. Finally, we do not allow riders to have the same pick-up and drop-off locations.

Each simulation starts by sampling up to d depot locations. The depots are sampled uniformly randomly from the grid, i.e., $d \in \{0, ..., X\} \times \{0, ..., Y\}$, with replacement. Then, for a given set of depots and a group of riders, the simulator executes the three mechanisms sequentially and tracks the variables we use in our plots.

Evaluation measures: We measure the quality of the mechanisms over the three experiments. Our goal is to measure how efficient the system is with respect to utility, how fair our mechanisms are, and how much time the system operates. To help the presentation, we expand our notations for the utility to $u_i^k(\cdot)$ to denote the utility of rider i at iteration k, t_i^k to denote the arrival time of rider i at iteration k, and ϕ_i^k to denote the finishing time of rider i at iteration k. We also let I denote the number of iterations in each experiment and recall that N is the set of all riders.

To measure efficiency, we first measure the projected utility of a rider: Using the notation we have introduced in Section 2, this can written as $u_i(\mathcal{F}(\theta^{t_i}; t_i))$. In Figures 2a,3a and in Figure 4a in the appendix, we present the average projected utility, normalised over the number of riders, i.e., the value

$$\frac{1}{I \cdot |N|} \sum_{i \in N} \sum_{k=1}^{I} u_i^k(\mathcal{F}(\theta^{t_i^k}; t_i^k)). \tag{7}$$

Our basic measure of efficiency is the ex-post utility, which is the value each rider receives after finishing the journey. Formally, this is defined as:

$$\frac{1}{I \cdot |N|} \sum_{i \in N} \sum_{k=1}^{I} u_i^k(\mathcal{F}(\theta^k)). \tag{8}$$

This is presented in Figures 2b,3b and 4b.

Another efficiency measure is the average commute time, i.e., the total time each rider needs to finish her commute. This is defined as

$$\frac{1}{I \cdot |N|} \sum_{i \in N} \sum_{k=1}^{I} (\phi_i^k - t_i^k). \tag{9}$$

This is presented in Figures 2c,3c and 4c.

Moving to the fairness measures, we first start with the natural, yet overly pessimistic measure of the worst-treated rider:

$$\frac{1}{I} \sum_{k=1}^{I} \min_{i \in N} u_i^k(\mathcal{F}(\theta^k)). \tag{10}$$

The minimum ex-post utility in our experiments is depicted in Figures 2d,3d and 4d.

A less pessimistic measure of equity is the Gini index, defined as the relative mean difference absolute difference among the utilities, averaged over the number of iterations:

$$\frac{1}{I} \sum_{k=1}^{I} \frac{\sum_{i \in N} \sum_{j \in N} \left| u_i^k(\mathcal{F}(\theta^k)) - u_j^k(\mathcal{F}(\theta^k)) \right|}{2|N| \cdot \sum_{i \in N} u_i^k(\mathcal{F}(\theta^k))}. \tag{11}$$

We present the average Gini index in Figures 2e, 3e and 4e.

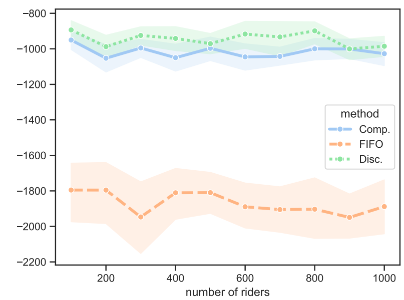
Finally, in Figures 2f, 3f, and 4f we measure the total travelling time of all the cars (including time moving asw an unallocated car) in the system, to quantify the environmental impact of our suggested mechanisms. Formally, let $x_{j,t}^k \in \{0, 1\}$ denote whether car j is moving (i.e., is not parked in any depot). The quantity we measure is:

$$\frac{1}{I \cdot |N|} \sum_{k=1}^{I} \sum_{t=1}^{T} \sum_{j \in M} x_{j,t}^k. \tag{12}$$

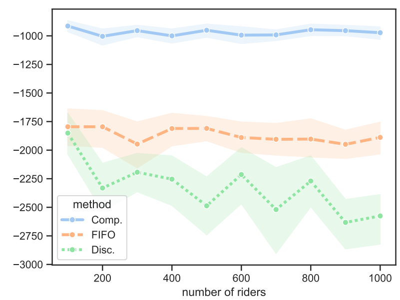
D ADDITIONAL EXPERIMENT

In this section, we present one additional experiment. Compared to the previous two experiments presented in Section 6, this simulation used non-identical cars with variable costs and capacities. The mean arrival time between riders is set to $\lambda = 5$ while the value of time is drawn uniformly with $B_L = 0$ and $B_U = 50$. The cars depart from 3 depots. Their cost is drawn uniformly in the [10, 50] interval. A car allows for up to 4 passengers with probability 50%, up to 6 passengers with probability 10%, up to 8 passengers with probability 30%, and finally, up to 10 passengers with probability 10%. In Figure 3, we present selected measures from the experiment.

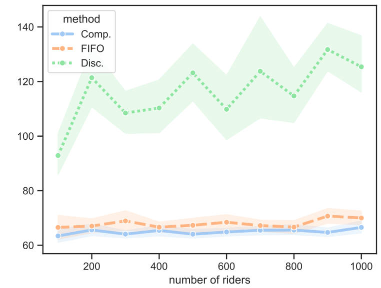
Compared to the identical vehicle case, the compensation mechanism yields an even higher margin on the ex-post utility, ensuring, on average, 75% more than the second-best option. The mechanism is also the best choice, by a slim margin, for the rider's average travelling time. In fairness, the Compensation mechanism replicates the trends of the previous experiments with respect to minimum ex-post utility. Still, it is the second-best option with respect to the Gini index, losing only to the FIFO mechanism. The differences in the total travelling time are very small to call a winner, although



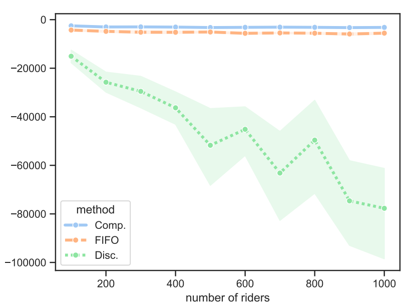
(a) mean ex-post utility



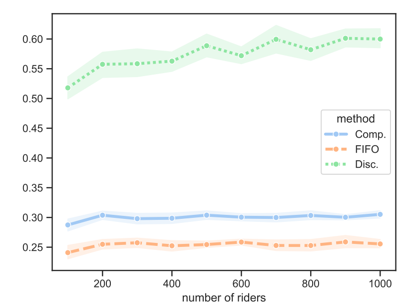
(b) mean ex-post utility



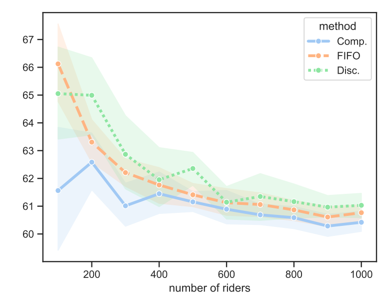
(c) mean ex-post commute time



(d) minimum ex-post utility



(e) Gini ex-post utility



(f) total travelling time

Figure 4: Simulations on a 50×50 grid, with average time between arrivals equal to $\lambda = 2$, value of time in the interval $B_L = 0$ and $B_U = 50$, and at most 10 identical vehicles departing from 2 depots. Variable vehicle costs and capacities. Comparison of the Compensation mechanism (solid, blue) with the FIFO (dashed, orange) and Discount mechanism (dotted, green). The plots depict averaged measures over 20 iterations with a 95% confidence interval.

the Compensation mechanism seems slightly better than the other two.