Multi-Objective Analysis of Ridesharing in Automated Mobility-on-Demand

Michal Čáp and Javier Alonso-Mora

Dept. of Cognitive Robotics, 3ME, TU Delft, the Netherlands

***Abstract*—Self-driving technology is expected to enable the realization of large-scale mobility-on-demand systems that employ massive ridesharing. The technology is being celebrated as a potential cure for urban congestion and others negative externalities of individual automobile transportation. In this paper, we quantify the potential of ridesharing with a fleet of autonomous vehicles by considering all possible trade-offs between the quality of service and operation cost of the system that can be achieved by sharing rides. We formulate a multi-objective fleet routing problem and present a solution technique that can compute Pareto-optimal fleet operation plans that achieve different trade-offs between the two objectives. Given a set of requests and a set of vehicles, our method can recover a trade-off curve that quantifies the potential of ridesharing with given fleet. We provide a formal optimality proof and demonstrate that the proposed method is scalable and able to compute such trade-off curves for instances with hundreds of vehicles and requests optimally. Such an analytical tool helps with systematic design of shared mobility system, in particular, it can be used to make principled decisions about the required fleet size.**

I. Introduction

Ubiquitous connectivity and rapid advances in automation are expected to revolutionize transportation of both goods and people. In particular, urban mobility is being rapidly transformed due to the emergence of new forms of on-demand transportation options, exemplified by services such as Uber or Lyft. In the near future, self-driving technology is expected to enable the realization of large-scale automated mobility-on-demand (AMoD) systems that provide personal point-to-point transportation that is as comfortable and affordable as traveling by private car, but uses a smaller fleet of shared vehicles, which translates to reduction of parking capacity requirements [6, 21, 17].

The benefits of automated on-demand mobility can be further increased by employing *ridesharing*, where multiple passengers traveling in a similar direction can be matched and transported in one vehicle. By employing fewer vehicles to serve fixed transportation demand, ridesharing has potential to reduce energy consumption, congestion and traffic-related pollution [16, 2]. Yet, our ability to quantify the extent of those benefits and to identify under what circumstances can those benefits be achieved is very limited.

We will refer to mobility-on-demand systems that use both autonomous vehicles and ridesharing as Shared Automated Mobility-on-Demand (SAMoD) systems. When designing a SAMoD system, we are typically interested in two main performance metrics. On the one hand, the users of the system are interested in the *quality of service* - users of the system desire to minimize service discomfort, i.e, they prefer to be delivered to their destination as fast as possible. On the other hand, the entity operating the system is interested in the minimization of the *operation cost* - this usually translates in the aim to minimize the fleet size and the total energy consumed by the system.

The service discomfort and operation cost objectives are usually in conflict, i.e, both cannot be minimized simultaneously. Instead, improvement in one criteria must be traded for degradation in the other criteria. For example, on the one hand, user discomfort is minimized by matching each request with a dedicated vehicle. And, on the other hand, the operation cost can be reduced by matching multiple request to a single vehicle, which in turn increases the user discomfort. Even though the two objectives are fundamentally intertwined on multiple levels, the majority of the existing work in SAMoD only considers one of the objectives individually and assumes a fixed fleet with a known number of vehicles. In particular, there is no principled study on how the two objectives interact on the operational level.

In this paper we model ridesharing as a multi-objective vehicle fleet routing problem with two criteria: to maximize service quality and to minimize operation cost. In contrast to single objective optimization, multi-objective optimization problems typically do not have one optimal solution. Instead, we are interested in a set of Pareto-optimal solutions [10]. A solution is called Pareto-optimal if there is no other solution that would achieve better performance in both considered objectives simultaneously. When the Pareto-optimal solutions are represented on the objective plane, with one axis representing the value of discomfort and the other axis the operation cost, we obtain a Pareto curve that graphically describes the best attainable trade-offs between the two objectives. Such a trade-off curve represents the fundamental limits of ridesharing for a particularity problem configuration at hand. In other words, there is no ridesharing strategy that projects below this curve.

A. Contribution

This paper has two main contributions. Firstly, we provide a formalization of ridesharing as a vehicle fleet routing problem with two competing objectives. Secondly, we design a scalable solution method for the problem. The method can generate representative Pareto-optimal solutions for problem instances consisting of a set of requests and a set of vehicles and

consequently recover the shape of the Pareto curve. Moreover, we formally prove the optimality of the method. Finally, we apply the method to compute the shape of Pareto curves both for synthetic problems and for a collection of 427 historical taxi trips in Manhattan.

The proposed analytical tool helps with systematic design of shared mobility system, in particular, it can be used to make principled decisions about the required fleet size. Yet, the proposed method is general and not limited to the analysis of SAMoD systems. It could also be employed to aid systematic design of other multi-robot multi-task assignment problems that include routing of a large vehicle fleet and tight performance constraints.

B. Related Work

The early works in AMoD focused on the development of models and algorithmic tools for on-demand fleets with single-occupancy vehicles [6, 14, 17, 20, 15]. However, one of the main promises of on-demand systems is the ability to implement massive ridesharing. This is, to match multiple customers, which are traveling in a similar direction, to a single vehicle. This translates to a significant reduction in the number of vehicles on the roads.

Ridesharing was traditionally formalized in the framework of Vehicle Routing Problems (VRP) [19], typically as a specific variant of VRP with Pickup and Delivery [4, 5, 3, 8] or a Dial a Ride Problem (DARP) [7, 5]. Yet, the existing exact VRP methods focus on instances with tens of vehicles and requests [13, 9] and as such they are not applicable to management or analysis of large-scale fleets that often consist of thousands of vehicles and requests.

The potential for large-scale ridesharing was studied using the shareability network model [16] revealing that up to 80% of the taxi trips in Manhattan could be pairwise shared such that the travel time is increased by no more than a couple of minutes. The analysis was later extended to other cities [18]. The model assumption of maximum two passengers in a vehicle was later lifted in [2], where Alonso-Mora et al. proposed a scalable technique for finding optimal assignment of requests to a given fixed fleet of vehicles such that the average travel delay is minimized. The problem of predictive routing in MoD systems has been recently also addressed [1, 11].

In this paper, we borrow several algorithmic ideas from [2] to design a multi-objective fleet routing algorithm that can be used to study the trade-off between the operation cost and the travel discomfort experienced by the users of the system. Specifically, the vehicle-group assignment (VGA) component presented in Section IV can be seen as a more concise reformulation of the method by Alonso-Mora et al. [2] that, e.g., avoids RV and RTV graph construction, which makes the VGA algorithm simpler to implement and to analyze.

II. Problem Statement

Consider a fleet of vehicles that has to service a given set of transportation requests. We study the problem of finding a collection of Pareto-optimal system operation plans that represent varying trade-offs between the service quality and the operation cost.

A. Vehicle Fleet and Transportation Demand

There is a fleet of $m \geq 1$ vehicles that can be used to serve the transportation demand. Each vehicle v starts at a given initial position o_v^{veh} at time t_v^{veh}. For convenience, we define the set of vehicle indices as $V = \{1, \dots, m\}$.

The transportation demand is modeled as a set of n transportation requests. The i-th request is a tuple $r_i = (o_i, d_i, t_i)$, where o_i is the origin of request i, d_i is the destination of request i, and t_i is the time when the request i was announced. The set of all request indices is denoted $R = \{1, \dots, n\}$.

Let $tt(x_1, x_2)$ denote the travel time from point x_1 to x_2, where each point represents an origin of a request, a destination of a request, or an initial position of a vehicle.

B. Vehicle Plan

A plan for a vehicle v denoted $\pi_v = ((o_v^{\text{veh}}, t_v^{\text{veh}}), o_1, o_2, \dots)$ encodes the initial spatio-temporal position of the vehicle and a sequence of orders that the vehicle should follow, where each order o_i is either to pickup a request r, or to drop-off a request r. For a plan to be valid, for every order to pickup request r, the plan must contain an order to drop-off the request r later in the sequence, and vice versa, for every drop-off order, the plan must contain a pickup order earlier in the sequence. The set of all requests served by a valid vehicle plan π_v is denoted as $\text{req}(\pi_v)$. The set of all valid plans for vehicle v is denoted by Π_v.

C. System Plan

A system plan assigns a particular plan to each individual vehicle. For a system plan to be valid, each vehicle plan must be valid and furthermore, all requests must be served by exactly one vehicle. Thus, the set of all valid system plans is denoted by $\overline{\Pi}$ and defined as

$$\overline{\Pi} := \Big\{ (\pi_1 \in \Pi_1, \dots, \pi_m \in \Pi_m) : \bigcup_{v \in V} \text{req}(\pi_v) = R \text{ and } \forall i, j \in V \; i \neq j : \; \text{req}(\pi_i) \cap \text{req}(\pi_j) = \emptyset \Big\}.$$

Observe that if there are no further constraints, then a valid system plan exists for any transportation demand. Consequently, a system plan that minimizes any desired cost criterion is guaranteed to always exist.

D. Optimization Criteria

There are two types of agents that have interest in the choice of the system plan: a) the users of the system and b) the operator of the system. While each of the users is interested in maximization of the quality of service, the operator is interested in the minimization of the operating cost. Furthermore, we will use the term "service discomfort" to represent the "negative" of service quality.

We assume that the discomfort perceived by user who issued request r depends only on the plan of the vehicle that serves the request.

Let $q_r(\pi_v)$ be a chosen service discomfort metric that measures the discomfort experienced by the customer who issued request r when the request r is served by vehicle v that follows plan π_v. To declutter the notation, we define the same metric over system plans and use $q_r(\overline{\pi})$ as a shorthand for $q_r(\pi_v)$, where $\pi_v \in \overline{\pi}$ is the plan of vehicle that serves request r.

We will define the discomfort metric as the time that elapses between the request announcement and the drop-off at the desired destination, i.e., $q_r(\pi_v) := t_r^{\text{dropoff}}(\pi_v) - t_r$, where $t_r^{\text{dropoff}}(\pi_v)$ is the time when the request r is dropped off under plan π_v.

Furthermore, let $s_v(\pi_v)$ denote the cost that the system operator has the bear when vehicle v executes plan π_v. For simplicity, we will define $s_v(\pi_v)$ to be equal to the time that the vehicle v spends in operation when following the plan π_v.

E. Baseline Plan

The system plan that minimizes the total user discomfort, or equivalently, the average drop-off time, is referred to as a baseline "maximum comfort" plan $\overline{\pi}^{BL}$ and it is defined as

$$\overline{\pi}^{BL} := \operatorname*{argmin}_{\overline{\pi} \in \overline{\Pi}} \sum_{r \in R} q_r(\overline{\pi}).$$

Furthermore, let $q_r^{BL} := q_r(\overline{\pi}^{BL})$ be the discomfort that request r experiences under the baseline plan.

Note that in some of our target scenarios, a baseline plan can be constructed trivially. Consider, for example, a system with a depot storing $m \geq n$ vehicles that need to serve a set of n requests. Then, the baseline plan is obtained by letting each request to be served by a dedicated vehicle. In a peer-to-peer ridesharing scenario, each passenger can either realize the trip using his own private vehicle or rideshare and travel in private car of another passenger. Thus, for n individual trips to be made, there are $m = n$ available vehicles. The baseline plan then corresponds to a situation in which no two rides are shared.

F. Trading Discomfort for Operation Cost

We can now proceed to the formalization of the problem of finding the set of system plans that trade-off discomfort for operation cost. The discomfort induced by ridesharing for a particular request r can be measured by considering the difference in discomfort relative to the baseline assignment. Let $\delta_r(\pi_v) := q_r(\pi_v) - q_r^{BL}$ represent the induced discomfort that the request r experiences under vehicle plan π_v. Now, we can define the service quality metric $c_s : \overline{\Pi} \to \mathbb{R}_{\geq 0}$ as

$$c_s(\overline{\pi} = (\pi_1, \ldots, \pi_m)) := \sum_{v \in V} \sum_{r \in \text{req}(\pi_v)} \delta_r(\pi_v)$$

and operation cost metric $c_o : \overline{\Pi} \to \mathbb{R}_{\geq 0}$ as

$$c_o(\overline{\pi} = (\pi_1, \ldots, \pi_m)) := \sum_{v \in V} s_v(\pi_v).$$

The above service quality metric aims to represents so-called social optimum, i.e, it aims at minimization of total user discomfort, which is equivalent to minimization of average discomfort across all users. This objective necessarily leads to solutions that distribute the discomfort unequally among individual requests. In result, some customers suffer from induced discomfort more than others. However, in many practical systems, the variance in induced discomfort as assigned to individual requests must be controlled. For example, human passengers are particularly sensitive to discomfort and are likely to switch to an alternative mode of transport if the induced discomfort is deemed significantly higher than the discomfort that other users need to bear. Therefore, we also introduce a bound on maximum induced discomfort of each transportation request and use $\delta_r^{\max}$ to denote the bound on maximum allowed induced discomfort assigned to a single transportation request r.

Our goal is to study the dynamics of the interaction between the induced discomfort and operation cost objectives subject to maximum induced discomfort constraints on individual requests. In particular, we would like to know what are the best possible trade-offs between the two criteria that can be potentially achieved. This can be expressed in a framework of multi-objective optimization as follows.

Problem 1 (Multi-objective Fleet Routing)**.** Given a fleet of vehicles, a set of transportation requests, and a travel time function, solve

$$\operatorname*{argmin}_{\overline{\pi} \in \overline{\Pi}} \ (c_s(\overline{\pi}),\ c_o(\overline{\pi})) \quad \text{subject to}$$
$$\delta_r(\overline{\pi}) \leq \delta_r^{\max}, \quad \forall r \in R.$$

A solution to the above problem is a set of all Pareto-optimal system plans, each representing a particular trade-off between the service discomfort and operation cost.

III. Solution Approach

Finding all Pareto-optimal solutions for a large-scale discrete multi-objective optimization problem, such as the one formulated in Problem 1, is not feasible in practice [10]. In order to obtain an approximation of the shape of the Pareto front for our problem, we apply a popular solution technique known as scalarization [10]. In this approach we solve a family of single-objective optimization problems parametrized by a weight parameter w, each asking for a system plan that minimizes a convex combination of the two considered objectives:

Problem 2 (Single-objective Fleet Routing)**.** Given a weight parameter $w \in [0, 1]$ solve

$$\operatorname*{argmin}_{\overline{\pi} \in \overline{\Pi}} \ w \cdot c_s(\overline{\pi}) + (1 - w) \cdot c_o(\overline{\pi}) \quad \text{subject to}$$
$$\delta_r(\overline{\pi}) \leq \delta_r^{\max}, \quad \forall r \in R.$$

An optimal solution of the above single-objective optimization problem is a Pareto-optimal solution for Problem 1. By finding a solution to the scalarized version of the problem for a

sequence of weights $w_0 = 0 < w_1 < \ldots < w_{k-1} < w_k = 1$, we can recover a collection of representative Pareto-optimal solutions that can be used to recover the shape of the Pareto front.

It should be noted, however, that although all solutions of scalarized problem are Pareto-optimal solutions, the opposite does not hold. In particular, the scalarization technique is able to generate all Pareto-optimal solutions that lie on the convex hull of the feasible set in the objective plane, but it is unable to generate Pareto optimal solutions that lie strictly inside the convex hull. However, such an approximation is often sufficient, because it is capable of describing the dynamics of the interaction between the two criteria and moreover, there is typically a good choice of representative solutions on the convex hull to choose from. Most importantly, using scalarization, one can efficiently recover the shape of Pareto-front even for large problem instances, as we will discuss in the following section.

IV. Vehicle-Group Assignment Method

The single-objective optimization problem stated in Problem 2 is a specific variant of a vehicle routing problem with multiple vehicles and time windows, a class of problems that are known to be NP-hard. In result, it cannot be solved efficiently in general. In this section, we introduce a method, that we will refer to as Vehicle-Group Assignment Method (VGA), that can solve many large-scale real-world instances of the problem optimally in practical time.

Vehicle-group assignment method finds optimal solution to Problem 2 by generating all possible groups of requests that each vehicle can serve and then by finding an optimal assignment of such groups to individual vehicles.

We will refer to a set of request as a group. We say that a group $G \subseteq R$ is feasible for vehicle v if the vehicle can serve all requests from the group without violating the maximum induced discomfort constraints. If a group G is feasible, we use $c(v, G)$ to denote the cost of minimum-cost plan for vehicle v that serves all requests in group G. The actual optimal plan for vehicle v to serve requests in group G is denoted as $\pi(v, G)$. In order to determine the feasibility and cost of vehicle v serving all requests in group G, one needs to solve a vehicle routing problem for a single vehicle starting at a given initial position and visiting pickup and destination positions of each request $r \in G$, such that a) the pickup point of each request is visited before the drop-off point and b) the maximum drop-off delay of each request is not exceeded.

Then, we can define $F_v \subseteq \mathcal{P}(R)$ to be a set of all feasible groups for vehicle v, where $\mathcal{P}(R)$ denotes the set of all subsets, i.e., the power set, of the set R.

A useful property of group feasibility, initially observed in [2], is the following: For a group to be feasible, all its subgroups must be feasible as well. More formally, for any vehicle v, we have $G \in F_v$ only if $\forall\ G' \subset G :\ G' \in F_v$. This observation can be exploited to design a procedure that iteratively generates the sets $F_v^1, F_v^2, \ldots$ containing feasible groups of size $1, 2, \ldots$ for vehicle v. The pseudocode of the group generation procedure is given in Algorithm 1.

Algorithm 1: Iterative generation of groups for vehicle v. The boolean-valued function $\mathrm{feasible}(v, G)$ evaluates to true, if vehicle v can serve all requests in group G without violating the maximum induced discomfort constraints.

```
F_v^0 ← {∅};
F_v^1 ← ∅;
for i ∈ R do
    if feasible(v, {i}) then
        F_v^1 ← F_v^1 ∪ {i};
k = 2;
while F_v^{k-1} ≠ ∅ do
    F_v^k ← ∅;
    forall G ∈ F_v^{k-1}, r ∈ F_v^1 do
        if ∀G' ⊂ G ∪ {r}, |G'| = k − 1 : G' ∈ F_v^{k-1} and
          feasible(i, G ∪ {r}) then
            F_v^k ← F_v^k ∪ {G ∪ {r}};
    k ← k + 1;
F_v ← F_v^0 ∪ F_v^1 ∪ F_v^2 ∪ ··· ∪ F_v^k;
```

After feasible groups have been generated for all vehicles, we need to choose a single group for each vehicle such that every request is served by exactly one vehicle. A vehicle-group assignment, typically denoted by a, is a mapping $V \to \mathcal{P}(R)$. For example, an assignment $a = \{(1, \{2, 3\}), (2, \{1\}), (3, \emptyset)\}$ represents the fact that vehicle 1 will serve requests 2 and 3, vehicle 2 will serve request 1 and vehicle 3 will be idle. The minimum-cost vehicle-group assignment a^* can be obtained by solving the following optimization problem:

Problem 3 (Vehicle-Group Assignment)**.** Given a set of feasible groups and a group cost function for each vehicle, solve

$$\operatorname*{argmin}_{a(1)\in F_1,\ldots,a(m)\in F_m} \sum_{v\in V} c(v, a(v)) \quad \text{subject to}$$
$$a(i) \cap a(j) = \emptyset \quad \forall i, j \in V,\ i \neq j.$$

Then, the optimal system plan $\overline{\pi} = (\pi_1, \ldots, \pi_m)$ can be recovered by taking $\pi_1 = \pi(1, a^*(1)), \ldots, \pi_m = \pi(m, a^*(m))$.

To solve Problem 3 using off-the-shelf ILP solvers, we can make the following straightforward conversion to a binary integer linear program:

$$\operatorname*{argmin}_{\{x_{v,G}\}} \sum_{v\in V} \sum_{g\in F_v} x_{v,G} \cdot c(v, G) \quad \text{subject to}$$
$$\sum_{G\in F_v} x_{v,G} = 1 \quad \forall v \in V$$
$$\sum_{v\in V} \sum_{G\in F_v} x_{v,G} \cdot \mathbb{1}_{F_v}(r) = 1 \quad \forall r \in R$$
$$x_{v,G} \in \{0, 1\} \quad \forall v \in V,\ \forall G \in F_v,$$

where $\mathbb{1}_S(x)$ is the indicator function, i.e., $\mathbb{1}_S(x) = 1$ if $x \in S$ and $\mathbb{1}_S(x) = 0$ otherwise.

The decision variables $x_{v,G}$ represent all possible vehicle-group assignments. The first constraints enforces that there is exactly one group assigned to each vehicle, while the second constraint enforces that every request is assigned to exactly one vehicle. If $x_{v,G} = 1$ in the optimal solution, then we have $a^*(v) = G$.

The vehicle-group assignment formulation turns out to be beneficial when the constraints on maximum drop-off delay become tight. Such constraints effectively eliminate groups containing requests that are far away from the vehicle, because the vehicle cannot arrive to the request in time. Furthermore, in some settings, such constraints will also effectively eliminate formation of larger groups of requests.

V. Theoretical Analysis

The Vehicle-Group Assignment method is an optimal solution algorithm for Problem 2, as stated by the following theorem.

Theorem 4. *If and only if a^* is a solution to Problem 3, then $(\pi(1, a^*(1)), \ldots, \pi(m, a^*(m)))$ is a solution to Problem 2.*

Proof: Recall the definition of Problem 2: $\operatorname{argmin}_{\overline{\pi} \in \overline{\Pi}} \; w \cdot c_s(\overline{\pi}) + (1 - w) \cdot c_o(\overline{\pi})$ subject to $\delta_r(\overline{\pi}) \leq \delta_r^{\max}, \; \forall r \in R$ and define $c_v^w(\pi_v) := w \cdot \sum_{r \in \mathrm{req}(v)} \delta_r(\pi_v) + (1 - w) \cdot s_v(\pi_v)$. Then, the objective criterion can be expressed as a sum of cost functions over single-vehicle plans as follows: $w \cdot c_s(\overline{\pi}) + (1 - w) \cdot c_o(\overline{\pi})$ $= w \cdot \sum_{v \in V} \sum_{r \in \mathrm{req}(v)} \delta_r(\pi_v) + (1 - w) \cdot \sum_{v \in V} s_v(\pi_v)$ $= \sum_{v \in V} \left(w \cdot \sum_{r \in \mathrm{req}(v)} \delta_r(\pi_v) + (1 - w) \cdot s_v(\pi_v) \right)$ $= \sum_{v \in V} c_v^w(\pi_v)$.

Similarly, the constraint $\delta_r(\overline{\pi}) \leq \delta_r^{\max}, \; \forall i \in R$, can be equivalently expressed as $\delta_r(\pi_v) \leq \delta_r^{\max}, \; \forall r \in \mathrm{req}(v) \; \forall v \in V$. Recall the definition of the set of all valid system plans $\overline{\Pi}$ and make the constraint forcing that every request is served by at most one vehicle explicit. We obtain the following reformulation of the above problem:

$$\begin{aligned} \operatorname*{argmin}_{\pi_1 \in \Pi_1, \ldots, \pi_m \in \Pi_m} \sum_{v \in V} c_v^w(\pi_v) \quad & \text{subject to} \\ \delta_r(\pi_v) \leq \delta_r^{\max}, \quad & \forall r \in \mathrm{req}(\pi_v) \; \forall v \in V. \\ \mathrm{req}(\pi_i) \cap \mathrm{req}(\pi_j) = \emptyset \quad & \forall i, j \in V, \; i \neq j. \end{aligned}$$

Define $\Pi_v(G) := \{\pi \in \Pi_v : \mathrm{req}(v) = G\}$,
$f_v(G) := \exists \pi \in \Pi_v(G) : \; \forall r \in \mathrm{req}(\pi_v) : \; \delta_r(\pi) \leq \delta_r^{\max}$,
$\pi_v(G) := \min_{\pi \in \Pi_v(G)} c_v^w(\pi)$ s.t. $\delta_r(\pi_v) \leq \delta_r^{\max}, \quad \forall r \in \mathrm{req}(\pi_v)$,
$c_v(G) := c_v^w(\pi_v(G))$. Assume arbitrary partitioning of requests to m disjoint groups. That is, let $G_1 \subseteq R$, …, $G_m \subseteq R$ such that $\forall i, j \in V, \; i \neq j : \; G_i \cap G_j = \emptyset$ and $\bigcup_{v \in V} G_v = R$. Given such partitioning, the optimal system plan $\overline{\pi}(G_1, \ldots, G_m) = (\pi'_1, \ldots, \pi'_m)$ is a solution to

$$\begin{aligned} \operatorname*{argmin}_{\pi_1 \in \Pi_1(G_1), \ldots, \pi_m(G_m) \in \Pi_m} \sum_{v \in V} c_v^w(\pi_v) \quad & \text{subject to} \\ \delta_r(\pi_v) \leq \delta_r^{\max}, \quad & \forall r \in \mathrm{req}(v) \; \forall v \in V. \end{aligned}$$

Observe that the optimization problem is decoupled and thus it is feasible if and only if $\forall_{v \in V} \; f_v(G_v)$. If it is feasible, we can obtain the optimal value for each optimization variable π_v, $v \in V$ independently as

$$\pi_v = \begin{array}{ll} \operatorname*{argmin}_{\pi'_v \in \Pi_v(G_v)} c_v^w(\pi'_v) & \text{subject to} \\ \delta_r(\pi'_v) \leq \delta_r^{\max}, & \forall r \in \mathrm{req}(v) \end{array} = \pi_v(G).$$

The minimum of the objective value can be obtained as $\sum_{v \in V} c_v^w(\pi_v) = \sum_{v \in V} c_v(G_v)$.

Now we prove the two directions of the equality.

1) By contradiction. Let $\pi_1^*, \ldots, \pi_m^*$ be a solution to Problem 2. Now assume that a^* is a solution to Problem 3, but $\exists v \in V$, such that $\mathrm{req}(\pi_v^*) \neq a_v^*$. This implies that there is another partitioning to groups that admits lower value of objective function. This is impossible because the solution to Problem 3 is a partitioning that minimizes the objective function.

2) By contradiction. Let $a_1^*, \ldots, a_m^*$ be a solution to Problem 3. Now assume that π^* is a solution to Problem 2, but $\exists v \in V$, such that $\mathrm{req}(\pi_v^*) \neq a_v^*$. There are two possibilities: a) It holds that $\forall v \in V : \; \mathrm{req}(\pi_v^*) = a_v^*$. This implies, that $\pi^*$is not an optimal plan given partitioning a^*. This is impossible because given a particular feasible partitioning the two formulations have been shown to be equivalent. b) It holds that $\exists v \in V : \; \mathrm{req}(\pi_v^*) \neq a_v^*$, i.e., the optimal solution to Problem 2 lies in different partitioning than the partitioning corresponding to the solution to Problem 3. This is impossible, because the problem is formulated such that all feasible partitionings are explored and the one containing minimum cost solution is selected. ■

VI. Experimental Analysis

In this section, we demonstrate the applicability of the proposed multi-objective optimization technique. We use the algorithm to obtain insights into the dynamics of interaction between the quality of service and the operation cost in a given transportation system. We first apply the algorithm in context of an idealized transportation system operating on an Euclidean plane and then demonstrate the applicability of the method to solve a real-world ridesharing problem.

A. Ridesharing in Euclidean Plane

We start by analyzing the behavior of the algorithm using synthetic instances that represent a fleet of holonomic vehicles moving at constant unit speed in the Euclidean plane and that have to service a collection of randomly generated travel requests in a rectangular region of the plane. More specifically, we generate n requests such that the origin point and destination point are sampled uniformly from region $[0, 100] \times [0, 100]$ and the announcement time is sampled from interval $[0, 50]$. We consider the peer-to-peer ridesharing scenario, i.e., each of the generated requests is assumed to have its own vehicle available at the origin of the request at the time of announcement of the requests. In result, each generated instance has n vehicles and n requests.

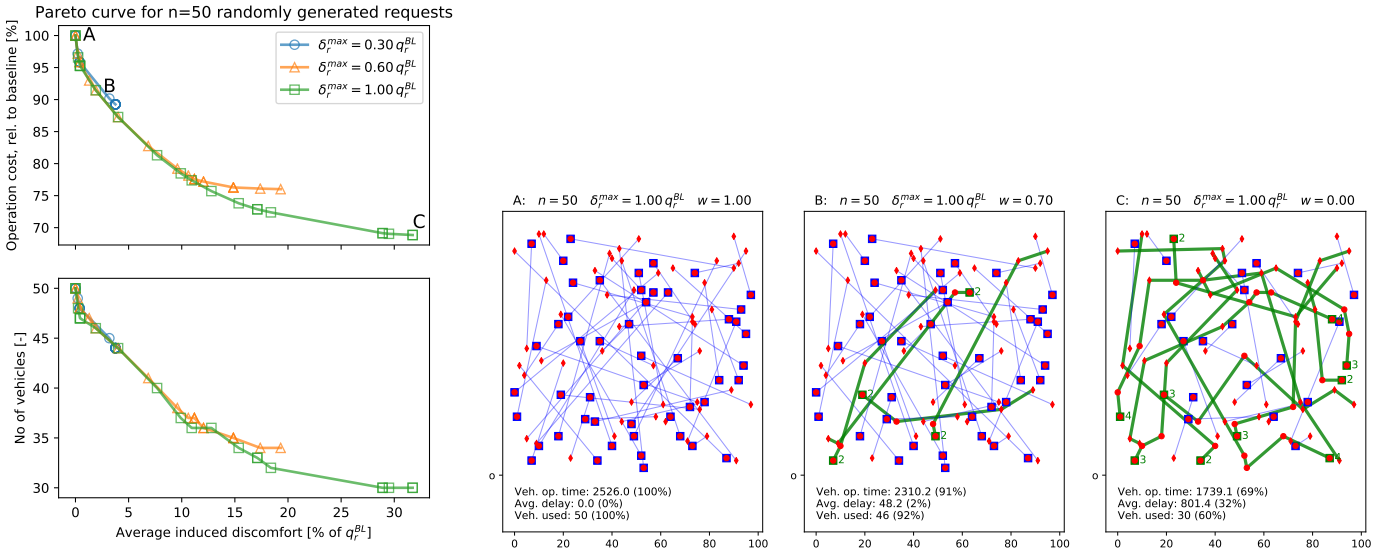

Figure 1. Ridesharing in Euclidean Plane. **Top-left:** Pareto curves in objective plane. Markers represent individual Pareto-optimal solutions. The line represent convex lower-bound on the Pareto-front. **Bottom-left:** The dependency between the average induced discomfort and fleet size used by Pareto-optimal solutions. **Right:** Three representative Pareto-optimal solutions for $\delta_r^{\max} = 1\, q_r^{BL}$. The origin of each request is denoted by a red circle and its destination is denoted by a red diamond. Blue boxes represent vehicles. The routes of vehicles that carry only one vehicle are shown by thin blue lines. The routes of vehicles that carry multiple requests are highlighted in green.

Illustration of Solution

Next, we illustrate how can be the VGA method used the recover the shape of Pareto front for a random synthetic problem instance. Figure 1-A shows the random instance in consideration. The instance consists of 50 random requests with origin points shown as red circles and destination points shown as red diamonds and 50 vehicles depicted as blue squares, initially located at the origin of each request. Figure 1-A also shows the baseline system plan: In this case, each vehicle picks-up its nearest request and drives directly to the drop-off point of the request.

To obtain a set of Pareto-optimal solutions, we solve Problem 2 for a sequence of weight parameter values ranging from $w = 0$ to $w = 1$. We repeat the process for three different bounds on induced discomfort $\delta_r^{\max}$. The three resulting Pareto curves, one for each value of $\delta_r^{\max}$, are shown in the top-left plot in Figure 1. Another parameter of interest that can be measured for each Pareto-optimal solution is the number of vehicles used in the system plan. The bottom-left plot in Figure 1 shows the dependency between the induced discomfort and the number of active vehicles.

In Figure 1, plots A, B, and C, we show Pareto-optimal system plans corresponding to three selected points on Pareto front for $\delta_r^{\max} = 1\, q_r^{BL}$. We can see that the system plan A, that projects to the point at the top-left end of the Pareto front, employs no ridesharing, since this plan optimizes solely the service quality. When we move in down along the Pareto curve, the respective Pareto-optimal plans contain an increasing number of trips that were merged together and are served by a single vehicle as exemplified by system plan B. The system plan C corresponds to the bottom extreme point of the Pareto curve and achieves the lowest operating cost that can be achieved subject to constraints on maximum induced discomfort, in this case $\delta_r^{\max} = 1\, q_r^{BL}$.

Even though the shown system plans are Pareto-optimal with respect to operation cost, that we define as total operation time over all vehicles, we can see that minimization of operation cost also indirectly leads to reduction of the fleet. This is because the only way to minimize operation cost is to share more rides, which in turn reduces the number of active vehicles in the solution.

Another phenomena that can be observed in our illustrative example is that the Pareto curve for an instance with a induced discomfort bound $\delta_r^{\max} = 0.6\, q_r^{BL}$ approximately coincides with the Pareto front for instance with relatively looser bound $\delta_r^{\max} = 1\, q_r^{BL}$ at the top part of the curve, but diverges at its bottom end. This divergence results from the conflict between the service quality objective, that asks for the minimization of the average induced discomfort, and the constraint that bounds the maximum induced discomfort of individual requests. Some Pareto-optimal solutions from the loosely constrained instance are not feasible in the case of more tightly constrained instances, because they distribute induced discomfort unequally among the individual requests and consequently violate the maximum induced discomfort bounds.

Application to Fleet Sizing

In the previous section, we have illustrated how we can generate a set of Pareto-optimal solutions for a problem instance consisting of a fixed set of requests and a fixed fleet of vehicles. Each such Pareto-optimal system plan implicitly prescribes how many and which specific vehicles from the

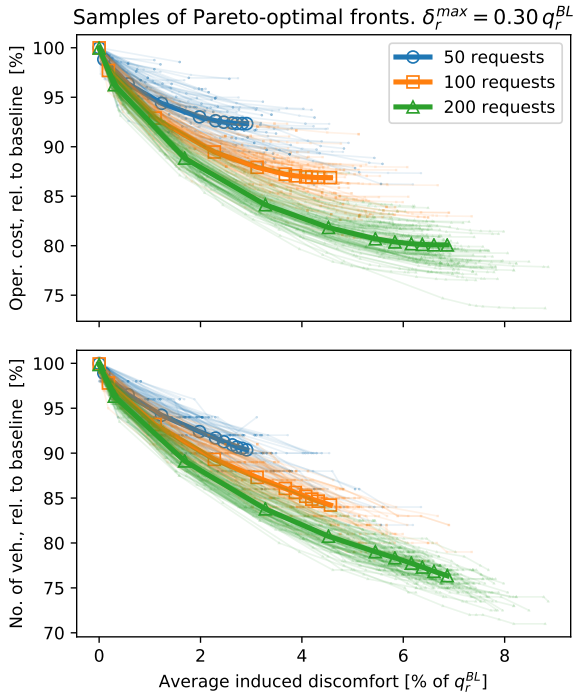

Figure 2. **Top:** Pareto curves for instances with $n \in \{50, 100, 200\}$ randomly generated travel requests in the same spatio-temporal region. The thick lines represent the "expected" Pareto curve for given density. **Bottom:** The number of vehicles (100% represents n vehicles) used in Pareto optimal solution corresponding to the individual Pareto curves.

fleet should serve the requests in order to achieve a particular Pareto-optimal trade-off between the operation cost and service discomfort.

The multi-objective fleet optimization approach can therefore be used for the purpose of operational fleet optimization, i.e., it answers the questions of how many and which of the currently available vehicles should be used to achieve optimal performance when serving a given, deterministically known transportation demand.

In the following, we would like to demonstrate that the method can be also used to get insights in the issue of strategic fleet optimization, i.e, to determine what is the optimal size and composition of a fleet of a transportation system before the transportation demand is revealed. For most transportation system, we have access either to historical data or to statistical information about the transportation demand to be served. Then, we can either use the historical data or take a sample from a demand model and use our method to obtain a set of Pareto-optimal system plans. By computing such Pareto-optimal system plans for different realizations of demand to be served, we can study which vehicles are active in each Pareto-optimal system plan. Then, we can estimate the distribution of different fleet parameters, e.g., we can recover the distribution of fleet sizes. Such information can be used to determine the appropriate fleet size for a system in hand. One can, for example, choose the fleet size such that it is larger than the size of 95% of optimal fleets achieving discomfort of 2% and at the same time larger than 99% of optimal fleets achieving discomfort of 5%.

To illustrate the value of the above approach, we will use it analyze how does the spatio-temporal density of demand in the system influence the cost savings that can be realized by employing ridesharing. We first generate a set Euclidean ridesharing instances with $n = 50$, $n = 100$, and $n = 200$ requests randomly generated in a spatio-temporal region of fixed size as described in Section VI-A. Then, we compute a set of Pareto-optimal solutions for each such instance. The Pareto curves for sampled instances are shown by thin lines in the top plot in Figure 2, where each demand density is plotted in a different color. The thick line represents "expected" Pareto curve for an instance with the indicated number of requests, obtained by averaging the values of the operation cost and average induced discomfort over the individual samples for individual values of $w = 0, \ldots, 1$. We can see that the Pareto curves for instances with different number of request (i.e., instances having different demand density) occupy different parts of the objective plane. We can further observe that when demand-density is increased, solutions having both lower operation cost and lower user discomfort can be found. More specifically, for 50 requests in the given space-time region, one can reduce operation cost by 6% in exchange for average 2% discomfort degradation. In contrast, for 200 requests in the same region, one can reduce operation cost by 12% in exchange for 1.8% discomfort. The bottom plot in Figure 2 shows the number of vehicles used in sampled Pareto-optimal solutions for different demand densities. Again, when the demand density is higher, we can find more favorable trade-offs between the fleet size and induced discomfort. As we can see, such an experiment provides a quantitative justification for the intuition suggesting that the benefits of ridesharing are best realized in areas with high density of travel requests.

B. Case Study: Ridesharing in Manhattan

In this section, we demonstrate the applicability of the proposed technique for analysis of a real-world transportation system. More specifically, we analyze the potential of ridesharing among taxi passengers in Manhattan. We base our analysis on the dataset released by NYC Taxi and Limousine Commission that contains a pickup time and origin and destination geo-coordinates for each passenger trip served by any of the 13 586 yellow taxis in New York City [12]. From this dataset, we select a 60-second slice of data from Tuesday, May 7th 2013 between 9:00:00 am and 9:01:00am, which consists of 427 requests across Manhattan. Furthermore, we consider the complete roadgraph of Manhattan consisting of 4 092 nodes and 9 453 edges. The travel time along each edge is estimated using the method described in [16]. The travel time between any two points on the map is then computed by finding the minimum-time path between the two given points on the roadgraph. We apply the proposed multi-objective optimization method to compute the Pareto-curve that represents best attainable trade-offs between operation

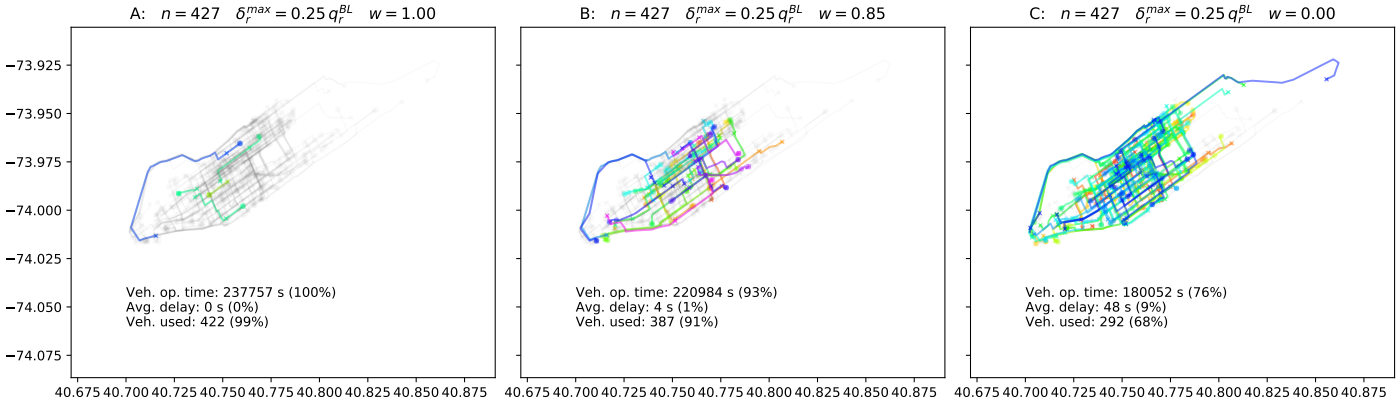

Figure 3. Manhattan Case Study. Three representative Pareto-optimal system plans for $\delta_r^{\max} = 0.25\, q_r^{BL}$. The routes of vehicles that carry only single passenger are plotted in semi-transparent grey. The routes of vehicles that carry multiple passengers are plotted by thicker line in color.

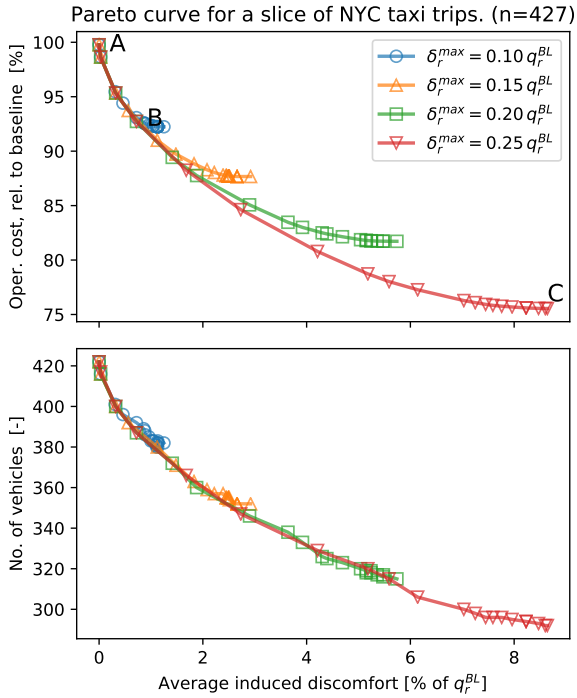

Figure 4. Manhattan Case Study. **Top:** Pareto curves in objective plane for different values of $\delta_r^{\max}$. The markers represent projections of representative Pareto-optimal solutions. **Bottom:** The number of vehicles used in each Pareto-optimal system plan.

cost and induced discomfort for such set of travel requests.

The results for different bounds on maximum individual discomfort are shown in Figure 4. The system plans corresponding to three different points (denoted A, B, and C) on the Pareto front for $\delta_r^{\max} = 0.25\, q_r^{BL}$ are shown in plots A, B, and C in Figure 4. The paths of taxis that carry one passenger are shown in light gray, the paths of taxis that were assigned multiple passengers are highlighted in color. We can see that for $\delta_r^{\max} = 0.25\, q_r^{BL}$, it is possible to reduce the operation cost of the fleet to 76% of the baseline operation cost (i.e., each request travels alone) and reduce the number of active vehicles from 452 to 292, while the average induced discomfort per request would increase to 48 s, which corresponds to $\delta_r = 0.09\, q_r^{BL}$.

VII. Conclusion

Urban mobility is being transformed by newly emerging forms of on-demand transportation. Self-driving technology, in particular, is expected to enable the operation of large centrally-controlled vehicle fleets. In this paper, we argued that the potential for ridesharing arises when the system operation cost can be traded-off for user discomfort and we studied the dynamics of the interaction between the two competing objectives. In particular, we formulated the problem as a multi-objective fleet routing problem and designed a computational method based on the idea of vehicle-group assignment. The method can compute a set of representative Pareto-optimal system plans to achieve different trade-offs between cost of operation and user discomfort. We gave a formal proof of optimality of the proposed method. Furthermore, we showed that the method is remarkably scalable and is capable of computing such trade-off curves for instances consisting of hundreds of requests and vehicles. In particular, we applied the method to a set of 427 taxi requests that were issued in Manhattan in a 60-second long time window.

In future work, we will investigate how the proposed method can be adapted, possibly by introducing approximations or heuristics, to compute the trade-off between operation cost and quality of service in even larger instances of the problem. We will also study what is the best way to use the information provided by our method to design shared automated mobility-on-demand systems and to appropriately select the required number of vehicles in the fleet.

Acknowledgements: The work presented in this paper was supported by Amsterdam Institute for Advanced Metropolitan Solutions (AMS).

References

[1] J. Alonso-Mora, A. Wallar, and D. Rus. Predictive routing for autonomous mobility-on-demand systems with ride-sharing. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3583–3590, September 2017.

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

[3] Roberto Baldacci, Enrico Bartolini, and Aristide Mingozzi. An Exact Algorithm for the Pickup and Delivery Problem with Time Windows. *Operations Research*, 59(2):414–426, April 2011.

[4] Gerardo Berbeglia, Jean-François Cordeau, Irina Gribkovskaia, and Gilbert Laporte. Static pickup and delivery problems: A classification scheme and survey. *TOP*, 15(1):1–31, July 2007.

[5] Gerardo Berbeglia, Jean-François Cordeau, and Gilbert Laporte. Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202(1):8–15, April 2010.

[6] L. D Burns, W. C. Jordan, and B. A. Scarborough. Transforming Personal Mobility. Technical report, Earth Institute, Columbia University, January 2013.

[7] Jean-François Cordeau and Gilbert Laporte. The dial-a-ride problem: Models and algorithms. *Annals of Operations Research*, 153(1):29–46, September 2007.

[8] L. Grandinetti, F. Guerriero, F. Pezzella, and O. Pisacane. The Multi-objective Multi-vehicle Pickup and Delivery Problem with Time Windows. *Procedia - Social and Behavioral Sciences*, 111:203–212, February 2014.

[9] Monirehalsadat Mahmoudi and Xuesong Zhou. Finding optimal solutions for vehicle routing problem with pickup and delivery services with time windows: A dynamic programming approach based on state-space-time network representations. *Transportation Research Part B: Methodological*, 89(Supplement C):19–42, July 2016.

[10] Kaisa Miettinen. *Nonlinear Multiobjective Optimization*. Springer Science & Business Media, 1999.

[11] J. Miller and J. P. How. Predictive positioning and quality of service ridesharing for campus mobility on demand systems. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1402–1408, May 2017.

[12] NYC Taxi and Limousine Commission. TLC Trip Record Data. http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml.

[13] Sophie N. Parragh and Verena Schmid. Hybrid column generation and large neighborhood search for the dial-a-ride problem. *Computers & Operations Research*, 40(1):490–497, January 2013.

[14] Marco Pavone, Stephen L Smith, Emilio Frazzoli, and Daniela Rus. Robotic load balancing for mobility-on-demand systems. *The International Journal of Robotics Research*, 31(7):839–854, June 2012.

[15] A. Prorok and V. Kumar. Privacy-preserving vehicle assignment for mobility-on-demand systems. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1869–1876, September 2017.

[16] Paolo Santi, Giovanni Resta, Michael Szell, Stanislav Sobolevsky, Steven H. Strogatz, and Carlo Ratti. Quantifying the benefits of vehicle pooling with shareability networks. *Proceedings of the National Academy of Sciences*, 111(37):13290–13294, September 2014.

[17] Kevin Spieser, Kyle Treleaven, Rick Zhang, Emilio Frazzoli, Daniel Morton, and Marco Pavone. Toward a Systematic Approach to the Design and Evaluation of Automated Mobility-on-Demand Systems: A Case Study in Singapore. *Road Vehicle Automation (Lecture Notes in Mobility)*, April 2014.

[18] R. Tachet, O. Sagarra, P. Santi, G. Resta, M. Szell, S. H. Strogatz, and C. Ratti. Scaling Law of Urban Ride Sharing. *Scientific Reports*, 7:42868, March 2017.

[19] Paolo Toth and Daniele Vigo. *Vehicle Routing: Problems, Methods, and Applications, Second Edition*. SIAM, December 2014.

[20] K. Treleaven, M. Pavone, and E. Frazzoli. Asymptotically Optimal Algorithms for One-to-One Pickup and Delivery Problems With Applications to Transportation Systems. *IEEE Transactions on Automatic Control*, 58(9):2261–2276, September 2013.

[21] Shared use Mobility Center. Shared-use Mobility - Reference Guide. Technical report, Shared-use Mobility Center, October 2016.