

# Routing and Rebalancing Intermodal Autonomous Mobility-on-Demand Systems in Mixed Traffic

Salomón Wollenstein-Betech<sup>ID</sup>, Student Member, IEEE, Mauro Salazar<sup>ID</sup>, Member, IEEE, Arian Houshmand<sup>ID</sup>, Marco Pavone<sup>ID</sup>, Member, IEEE, Ioannis Ch. Paschalidis<sup>ID</sup>, Fellow, IEEE, and Christos G. Cassandras<sup>ID</sup>, Life Fellow, IEEE

**Abstract**—This paper studies congestion-aware route-planning policies for intermodal Autonomous Mobility-on-Demand (AMoD) systems, whereby a fleet of autonomous vehicles provides on-demand mobility jointly with public transit under mixed traffic conditions (consisting of AMoD and private vehicles). First, we devise a network flow model to jointly optimize the AMoD routing and rebalancing strategies in a congestion-aware fashion by accounting for the endogenous impact of AMoD flows on travel time. Second, we capture the effect of exogenous traffic stemming from private vehicles adapting to the AMoD flows in a user-centric fashion by leveraging a sequential approach. Since our results are in terms of link flows, we then provide algorithms to retrieve the explicit recommended routes to users. Finally, we showcase our framework with two case-studies considering the transportation sub-networks in Eastern Massachusetts and New York City, respectively. Our results suggest that for high levels of demand, pure AMoD travel can be detrimental due to the additional traffic stemming from its rebalancing flows. However, blending AMoD with public transit, walking and micromobility options can significantly improve the overall system performance by leveraging the high-throughput of public transit combined with the flexibility of walking and micromobility.

**Index Terms**—Mobility-on-demand, system-optimal routing, rebalancing, mixed autonomy.

## I. INTRODUCTION

SINCE the last century, urban mobility has been dominated by the use of *private* vehicles. The success of this mode of transport relies on its fast and convenient point-to-point transportation service. However, even though

Manuscript received 22 February 2021; revised 7 July 2021; accepted 13 August 2021. Date of publication 20 September 2021; date of current version 9 August 2022. This work was supported in part by the NSF under Grant ECCS-1931600, Grant DMS-1664644, Grant CNS-1645681, Grant IIS-1914792, and Grant CMMI-1454737; in part by the Air Force Office of Scientific Research (AFOSR) under Grant FA9550-19-1-0158; in part by the Advanced Research Projects Agency-Energy (ARPA-E) under Grant DE-AR0001282; in part by the NEXTCAR Program under Grant DE-AR0000796; in part by the MathWorks; in part by the Office of Naval Research (ONR) under Grant N00014-19-1-2571; in part by the National Institutes of Health (NIH) under Grant R01 GM135930; and in part by the Toyota Research Institute (TRI). The Associate Editor for this article was A. Nunez. (*Corresponding author:* Salomón Wollenstein-Betech.)

Salomón Wollenstein-Betech, Arian Houshmand, Ioannis Ch. Paschalidis, and Christos G. Cassandras are with the Division of Systems Engineering, Boston University, Boston, MA 02215 USA, and also with the Center for Information and Systems Engineering, Boston University, Boston, MA 02215 USA (e-mail: salomonw@bu.edu; ariahm@bu.edu; yannisp@bu.edu; cgc@bu.edu).

Mauro Salazar is with the Department of Mechanical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands (e-mail: m.r.u.salazar@tue.nl).

Marco Pavone is with the Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94325 USA (e-mail: pavone@stanford.edu). Digital Object Identifier 10.1109/TITS.2021.3112106

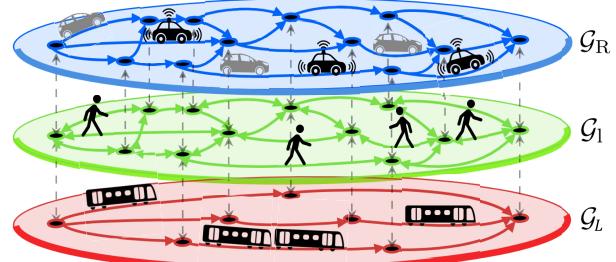


Fig. 1. Intermodal AMoD network (supergraph) consisting of three layers: the road network (blue with black AMoD cars and grey private vehicles, respectively), walking pathways (green) and subway lines (red); the dashed arrows represent switching arcs.

this technology has been widely adopted, it is also severely criticized due to its dependency on gasoline, its harmful emissions to the environment, its underutilization (according to [1], private vehicles are parked for more than 95% of the time), its impact on traffic congestion, and its land and infrastructure requirements for wider roads and parking spaces. Hence, some have acknowledged that private vehicles are an unsustainable solution for urban mobility [2]. As we think and plan for the cities of the future, mobility-on-demand (MoD), or Autonomous Mobility-on-Demand (AMoD) systems enabled by autonomous vehicles, offer a new way to provide a comparably fast and comfortable point-to-point service while maintaining low congestion levels. As defined by the Department of Transportation (DOT) of the United States, a MoD system is an “innovative, user-focused approach which leverages emerging mobility services, integrated transit networks and operations, real-time data, connected travelers, and cooperative Intelligent Transportation Systems (ITS) to allow for a more traveler-centric, transportation system-of-systems approach, providing improved mobility options to all travelers and users of the system in an efficient and safe manner.” [3].

In this paper we focus on methodologies that optimize the operations of AMoD systems with the goal of reducing traffic congestion. To achieve this, we develop a coordinated intermodal routing procedure that seeks to minimize the overall commuters travel time while ensuring that all travelers are being served by the same platform. In particular, we study the *routing* and *load-balancing* processes of a fleet of vehicles belonging to an AMoD service when they interact with self-interested vehicles in the network. In contrast to commonly used platforms today (e.g., Uber, Lyft, DiDi), our objective is to take these two decisions *jointly* rather than separately. If the vehicles belonging to the fleet are controlled

by self-interested humans, this joint optimization is much more difficult to carry out as compensation schemes would have to be designed in order to steer the selfish behavior to a system-optimal solution [4], [5]. However, as the level of automation of these platforms increases, with many of them already testing their Connected and Automated Vehicles (CAVs) in our streets [6], [7], the question of addressing these processes in a joint fashion attains high significance.

*Literature Review:* We first review techniques to solve the routing and load-balancing (also referred to as *rebalancing*) problems individually and then focus on the joint problem.

Current drivers in MoD platforms, such as Uber, Lyft or DiDi, choose their paths by using routing apps (e.g., Waze and Google Maps). These apps recommend routes using traditional shortest path algorithms such as Dijkstra's *et al.* [8], Bellman [9], and incremental graph [10] that find provable optimal routes. Also, widely employed are heuristics such as A-star [11], tabu search [12] and genetic algorithms [13] given that they provide a balance between solution quality and computation time. This *User-Centric* (UC) approach to routing, in which every driver minimizes their own travel time, is suboptimal compared to *System-Optimal* (SO) routing schemes achievable when vehicles are coordinated by a central controller. The bounds on the inefficiencies of the UC solution compared to the SO are explored in [14], showing that for linear travel time functions, the cost of the UC is bounded by  $4/3$  the cost of the SO solution. The gap between UC and SO is commonly known as the *Price of Anarchy* and is studied for real networks in [15]. With the aid of SO routing algorithms it has been established that mild modifications to the UC Traffic Assignment Problem (TAP) can solve the SO [16]. Therefore using algorithms to solve the TAP such as the Method of Successive Averages [16], Frank *et al.* [17] LeBlanc *et al.* [18] and Florian [19], or the Traffic Assignment by Paired Alternative Segments (TAPAS) [20] is sufficient to solve a SO problem. A relevant feature of SO routing is its fairness implications (users taking longer routes rather than the shortest), which is studied in [21] where an optimization algorithm termed Partran (a revised version of the Frank-Wolfe method) is proposed. In a mixed traffic setting, the interaction between a fleet of CAVs using SO routing coupled with reactive UC private vehicles is investigated theoretically by [22], where a reduction in headways is considered thanks to the adaptive cruise control technology included in CAVs. However, this analysis requires a network configuration of parallel links and is not suitable for general transportation networks. To overcome this, [23]–[25] propose an iterative approach to find a solution between these two classes of vehicles known as *diagonalization scheme*. In [23] the authors show that both CAVs and private vehicles can achieve a better performance in terms of travel time and energy savings as the percentage of CAVs in the network increases. However, neither of these approaches addresses the rebalancing of CAVs nor do they consider the possibility of intermodal (or multimodal) routing. Both limitations are addressed in this paper.

Aside from routing, rebalancing is tackled in practice by providing drivers with a real-time heat-map of the users' demand such that the driver is incentivized to relocate to

an area that will maximize its profits. Rebalancing has been studied by researchers using *proactive* (or *planning*) strategies that redistribute the fleet across regions in order to meet a forecasted demand.<sup>1</sup> Using this perspective, [29] shows that rebalancing is necessary to avoid building unbounded customer queues and to stabilize the system. Reference [29] proposes a rebalancing policy that minimizes the empty vehicle travel time under static (steady-state) conditions using a fluidic model. Furthermore, [30] proposes a queueing-theoretical approach to account for customers leaving the system when their waiting times are long. This method repeatedly solves a Linear Program (LP) that balances the fleet availability across the regions. Moreover, the authors show Pareto optimal curves relating the desired quality of service and fleet size. Similarly, [31] proposes a method that minimizes the number of customer dropouts instead of the empty driven miles to focus on service quality. Different from these queueing models, simulation-based methods are also employed [32]–[34].

More recently, schemes that consider the effects of rebalancing in routing and congestion have been analyzed. Threshold approximations of the travel time function have been used to study congestion effects [35], sometimes capturing the interaction with public transit [36], [37], or with the power-grid [38], [39]. These threshold schemes work as binary decisions allowing for the use of a road (or not), depending on whether the flow has exceeded the threshold or not, but do not capture different travel times for different flow levels on each link. To account for flow-based routing schemes most work leverages the classical Bureau of Public Roads (BPR) congestion model [40] together with network optimization methods. In particular, [41] provides a Frank-Wolfe algorithm, where dummy nodes are added to the transportation network to account for rebalancing flows and where the BPR function is evaluated when designing routes. However, this approach cannot include other modes of transportation such as walking, micromobility options, or public transit. Against this backdrop, a piecewise-affine approximation of the travel time function is introduced in [42] which converts the joint problem to a quadratic program. In this work, we extend this approximation in order to account for more accurate, fast and implementable models.

*Statement of Contributions:* The contribution of this paper is threefold. First, we present a method to optimize inter-modal congestion-aware routing and rebalancing policies of an AMoD service. The objective is to improve the quality of service by jointly reducing the overall user travel time while ensuring vehicle availability in every region. We allow AMoD users to use multiple modes of transportation such as public transportation, walking or micromobility (e.g., bikes and e-scooters) in order to reduce the overall travel time. To solve the routing and rebalancing problem, we approximate the non-linear travel latency function with a piecewise-affine function. This slight modification allows us to write the problem as a tractable quadratic program and subsequently

<sup>1</sup>Note that this process finds good coverage of vehicles over regions of the system and it is not focused on *matching* or *assigning* vehicles to customers. This vehicle-passenger *assignment* is explored in [26]–[28].

to a relaxed linear program, making it easier and faster to solve. We prove that this approximation is asymptotically optimal in the number of segments defining the piecewise function and we leverage origin-based formulations of the problem to improve the computational efficiency. Second, we extend the joint formulation to a mixed traffic setting capturing the interaction between AMoD users and private vehicle users. This model provides the routing decisions for the AMoD users which anticipate the behavior of the private vehicles. To do so, we leverage a sequential method that finds a steady-state solution for these two user types. Third, given that the proposed methods retrieve solutions expressed in terms of traffic flows, we propose distributed algorithms to convert the flows to viable routes, enabling real-time route recommendations for the AMoD users. Finally, we present experiments to (*i*) empirically show the asymptotic behavior of the approximated model; (*ii*) capture the trade-off between the benefits of SO routing and the excess flow due to rebalancing; (*iii*) observe the effect of intermodality on travel times; and (*iv*) show the applicability of the route-recovery strategies.

Building on the model and preliminary analysis in [43], this paper provides the following contributions: (*i*) extension of the approximated model from 3-segment to an  $n$ -segment model; (*ii*) new theoretical and empirical results on the asymptotic behavior of the approximated model; (*iii*) introduction of an origin-based formulation to improve the computational efficiency; (*iv*) development of route-recovery algorithms from the flow-based solution; (*v*) additional experimental results including a case study on a larger network of New York City incorporating the subway system as an intermodal option.

*Organization:* The remainder of the paper is organized as follows: In Section II we present the models used and the problem formulation. In Section III we develop the piecewise-affine approximation formulation along with the main analytical results of this paper. In Sections IV and V, we provide a framework for the mixed traffic problem and route-recovering strategies, respectively. Finally, we present experiments using the Eastern Massachusetts and New York City transportation networks in Section VI and we conclude in Section VII.

## II. PROBLEM FORMULATION

Consider an AMoD system which provides a mobility service through multiple modes of transportation (e.g., autonomous taxi-rides, walking and mass transit). To model the system, let  $\mathcal{G}$  be a network composed by the road layer and  $L$  additional layers, each representing a different transportation mode (Fig. 1). We denote by  $\mathcal{G}_R = (\mathcal{V}_R, \mathcal{A}_R)$  the road layer and by  $\mathcal{G}_l = (\mathcal{V}_l, \mathcal{A}_l)$ , for  $l = 1, \dots, L$ , the other layers where  $(\mathcal{V}_R, \mathcal{A}_R)$  and  $(\mathcal{V}_l, \mathcal{A}_l)$  are the sets of vertices and arcs for each layer. Then, the supergraph  $\mathcal{G} = (\mathcal{V}, \mathcal{A})$  is composed of all layers and a set of *switching* arcs, denoted by  $\mathcal{A}_S$ , that connect the network layers to allow AMoD users to switch modes (see dotted lines in Fig. 1). Formally  $\mathcal{G}$  is composed of the set of vertices  $\mathcal{V} = \mathcal{V}_R \cup \mathcal{V}_1 \cup \dots \cup \mathcal{V}_L$  and arcs  $\mathcal{A} = \mathcal{A}_R \cup \mathcal{A}_1 \cup \dots \cup \mathcal{A}_L \cup \mathcal{A}_S$ .

In order to model the demanded trips, let  $\mathbf{w} = (w_s, w_t)$  denote an Origin-Destination (OD) pair and  $d_{\mathbf{w}} \geq 0$  the demand rate at which customers request service per unit time

from origin  $w_s$  to destination  $w_t$ . Let  $W$  be the total number of OD pairs and  $\mathcal{W} = \{\mathbf{w}_k : \mathbf{w}_k = (w_{sk}, w_{tk}), k = 1, \dots, W\}$  the set of OD pairs. Let a vectorized version of the demand be  $\mathbf{g} = (d_{\mathbf{w}_k}; k = 1, \dots, W)$ .

To keep track of the AMoD user flow on a link, we let  $x_{ij}^{\mathbf{w}}$  denote the AMoD flow induced by OD pair  $\mathbf{w}$  on link  $(i, j) \in \mathcal{A}$ . Given that the AMoD system needs to rebalance its vehicles to ensure service, we let  $x_{ij}^r$  be the *rebalancing flow* on link  $(i, j) \in \mathcal{A}_R$ . Finally, to consider the interaction between the AMoD provider and the other (private) vehicles, we let  $x_{ij}^p$  be the self-interested *private vehicle* flow on  $(i, j) \in \mathcal{A}_R$ . We use the term “private” as we assume that self-interested users must arrive at their destination with their vehicle and do not have the option of switching transportation mode since they have a parking constraint. To simplify notation, we let the AMoD user flow on any link be

$$x_{ij}^u = \sum_{\mathbf{w} \in \mathcal{W}} x_{ij}^{\mathbf{w}}, \quad \forall (i, j) \in \mathcal{A}, \quad (1)$$

and the total flow on a link be

$$x_{ij} = x_{ij}^u + x_{ij}^r + x_{ij}^p, \quad \forall (i, j) \in \mathcal{A}. \quad (2)$$

Note that neither rebalancing flow  $\mathbf{x}^r$  nor private vehicle flow  $\mathbf{x}^p$  exists on layers  $l = 1, \dots, L$  nor on the switching links in Fig. 1. Hence, for those links we set  $x_{ij}^r = x_{ij}^p = 0$  for all  $(i, j) \in \mathcal{A} \setminus \mathcal{A}_R$ .

We specify the time it takes to cross link  $(i, j)$  as  $t_{ij}(x) : \mathbb{R}_+^{|\mathcal{A}|} \mapsto \mathbb{R}_+$ . Using the same structure as in [44], we characterize  $t_{ij}$  as a *travel time* function that maps the flow  $x_{ij}$  on a link to a travel time as follows:

$$t_{ij}(x_{ij}) = t_{ij}^0 f(x_{ij}/m_{ij}), \quad (3)$$

where  $m_{ij}$  is the link capacity,  $t_{ij}^0$  is the free-flow travel time on link  $(i, j)$ , and  $f(\cdot)$  is a strictly increasing, positive, and continuously differentiable function. To ensure that the travel time is equal to the free-flow travel time when there is no flow on the link, we consider functions with  $f(0) = 1$ . These functions are typically increasing polynomials that are hard to estimate [45]. Despite this, a widely used function by transportation engineers is the *Bureau of Public Roads (BPR)* function [40] denoted by

$$t_{ij}(x_{ij}) = t_{ij}^0 (1 + \alpha(x_{ij}/m_{ij})^\beta). \quad (4)$$

where typically  $\alpha = 0.15$  and  $\beta = 4$ . For a discussion on how to estimate these functions see [45].

Throughout this paper, we use this function to decide the routes of AMoD users and private vehicles, given the network flow levels. However, our analysis allows for any strictly increasing travel time function. For the  $L$  layers (excluding the road layer) we consider a constant travel time (independent of the flow) on every link.

### A. System-Optimal Routing and Rebalancing of AMoD Systems

Recall that our goal is to find the system-optimal congestion-aware routes and rebalancing policies of an AMoD provider. Let  $d_{\mathbf{w}}^u$  be customer rate requests to the AMoD system for passengers traveling from  $w_s$  to  $w_t$ , and  $\mathbb{1}_{i=j}$  be

the indicator function equal to 1 when  $i = j$  and 0 otherwise. The problem we aim to solve is then expressed by

$$\min_{\mathbf{x}^W, \mathbf{x}^r} J(\mathbf{x}) := \sum_{(i,j) \in \mathcal{A}} t_{ij}(x_{ij})x_{ij}^u + \sum_{(i,j) \in \mathcal{A}_R} c_{ij}x_{ij}^r \quad (5a)$$

$$\text{s.t. } \sum_{i:(i,j) \in \mathcal{A}} x_{ij}^W + \mathbb{1}_{j=w_s} d_w^u = \sum_{k:(j,k) \in \mathcal{A}} x_{jk}^W + \mathbb{1}_{j=w_t} d_w^u, \\ \forall \mathbf{w} \in \mathcal{W}, j \in \mathcal{V}, \quad (5b)$$

$$\sum_{i:(i,j) \in \mathcal{A}_R} (x_{ij}^r + x_{ij}^u) = \sum_{k:(j,k) \in \mathcal{A}_R} (x_{jk}^r + x_{jk}^u), \\ \forall j \in \mathcal{V}_R, \quad (5c)$$

$$\mathbf{x}^W, \mathbf{x}^r \geq 0, \quad (5d)$$

where we use bold notation  $\mathbf{x}$  to represent a vector containing all the elements of  $x_{ij}$ . The dimensions of the decision vectors  $\mathbf{x}^r$  and  $\mathbf{x}^W$  are given by  $\mathbf{x}^r \in \mathbb{R}^{|\mathcal{A}_R|}$ , and  $\mathbf{x}^W = \{\mathbf{w} \in \mathbb{R}^{|\mathcal{A}|} \mid \mathbf{w} \in \mathcal{W}\}$ . Constraints (5b) take care of flow conservation and demand compliance as in a multi-commodity transportation setting. Constraints (5c) ensure the rebalancing of the AMoD fleet (only on the road network). The last sets of constraints (5d) restrict the flows to non-negative values.

The objective  $J(\mathbf{x})$  is composed of two terms. The first term considers the total travel time of AMoD users. This evaluates the travel time function  $t_{ij}(x_{ij})$  with respect to the total flow given by (2) which includes variables corresponding to private vehicle flow  $x_{ij}^p$  (assumed to be fixed), and the rebalancing flow  $x_{ij}^r$ . When taking the product  $t_{ij}(x_{ij})x_{ij}^u$ , we obtain a non-convex function which makes the problem hard to solve. To address this issue, we use a piecewise-affine approximation of  $t_{ij}(x_{ij})$  which is further developed in Section III. The second term acts as a linear regularizer whose purpose is to penalize rebalancing flows. This will ensure that a cost for rebalancing of the fleet is taken into account in the optimization problem. We use  $c_{ij} = \lambda x_{ij}^0$  where  $\lambda$  is a constant. Therefore, we use a small  $\lambda$  to guide the rebalancing flow through good paths, without dominating the AMoD user routing decisions. Note that, if normalization is needed to ensure a good regularization parameter, we can always bound each component on (5a) using the link capacities and a large enough value for  $t(\cdot)$ .

### B. Private Vehicles Flow Modeling

Aiming to understand the interaction between a SO AMoD fleet and private vehicles, we assume some user-choice model behind private vehicle decisions. To do so, we use the *User-Centric* (UC) routing as in the Traffic Assignment Problem (TAP) [16]. Given OD demands, this model finds the flows in the network which achieve a Wardrop equilibrium [46]. This is equivalent to each private user deciding to take the route that minimizes their own travel time. Moreover, we impose that private vehicles can travel exclusively through the road network  $\mathcal{G}_R$  as opposed to traveling in the full network  $\mathcal{G}$ . Let  $x_{ij}^{p,w}$  be the flow on link  $(i, j)$  induced by private vehicle demand  $d_w^p$  of OD pair  $w$ . Then, we assume private vehicles decide their routes by using the UC approach. This is equivalent to solving the following (find more details on the

derivation of this model in [16])

$$\min_{\mathbf{x}^p} \sum_{(i,j) \in \mathcal{A}_R} \int_{x_{ij}^u + x_{ij}^r}^{x_{ij}} t_{ij}(s) ds \quad (6a)$$

$$\text{s.t. } \sum_{i:(i,j) \in \mathcal{A}_R} x_{ij}^{p,w} + d_w^p \mathbb{1}_{j=w_s} = \sum_{k:(j,k) \in \mathcal{A}_R} x_{jk}^{p,w} \\ + d_w^p \mathbb{1}_{j=w_t}, \quad \forall w \in \mathcal{W}, j \in \mathcal{V}_R, \quad (6b)$$

$$\mathbf{x}^{p,w} \geq 0. \quad (6c)$$

Notice that this version of the UC TAP is slightly different to the classic one [16] since it considers the AMoD flow in its objective (see limits of the integral on (6a)). To solve this problem we assume that the AMoD flow is fixed and private vehicles plan their routes considering AMoD flows as exogenous. By assuming this, we can use the *Frank-Wolfe* algorithm [17] to solve (6). Let us use the shorthand notation of  $\text{TAP}(\mathbf{g}, \mathbf{x}^e)$  to indicate the solution of (6) with  $\mathbf{x}^e$  equal to any generic exogenous flow. Hence  $\mathbf{x}^p \in \text{TAP}(\mathbf{g}^p, \mathbf{x}^u + \mathbf{x}^r)$ .

### C. AMoD in Mixed Traffic

Critically, AMoD flows react to the decisions made by private vehicles and these, in turn, react to AMoDs' flows. Hence, whenever private vehicles make their routing decisions, the AMoD fleet adjusts theirs, and vice versa. This creates a nested optimization problem between these two classes of vehicles. To give a formal definition of this game-theoretical problem we use the following bilevel optimization formulation

$$\min_{\{\mathbf{x}^W\}_{w \in \mathcal{W}}, \mathbf{x}^r, \mathbf{x}^p} J(\mathbf{x}) \quad (7a)$$

$$\text{s.t. } (5b) - (5d),$$

$$\mathbf{x}^p \in \text{TAP}(\mathbf{g}^p, \mathbf{x}^u + \mathbf{x}^r), \quad (7b)$$

which has the same structure as (5) with the additional constraint (7b). The latter constraint refers to the TAP (the lower-level problem), which depends on the solution of the full problem (upper-level). Note that the upper-level problem is minimizing over the AMoD users, rebalancing, and privately-owned vehicle flows. This phenomenon has been identified and is often described as a *Stackelberg game* framework [47]. In this setting, there is a *leader* agent (in our case the AMoD manager) and a *follower* (the private vehicles). In the context of transportation networks, [47] derived sufficient conditions to solve this problem when the network has parallel links. Although these models enable a better understanding of the phenomenon, they are not applicable to general networks and one can hardly assess mixed traffic routing in realistic networks. To address this limitation, we leverage the iterative approach in [23] to compute the private vehicles' and AMoD flows. The formal convergence of this sequential method is not studied in this paper.

## III. AMoD ROUTING AND REBALANCING PROBLEM

The problem of routing and rebalancing as stated in (5) is non-convex for typical travel time functions such as the BPR.

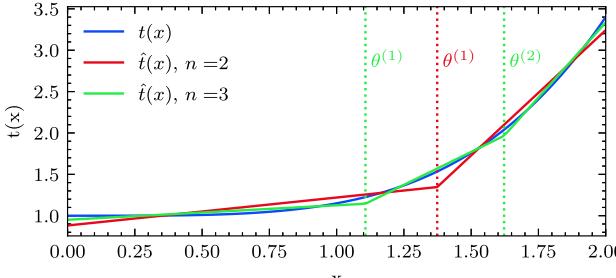


Fig. 2. Travel time function approximation.

This happens due to the term  $t(x_{ij})x_{ij}^r$  in the objective function. To address this issue, we approximate the travel time function with a piecewise-affine function.

#### A. Piecewise-Affine Approximation

Let the function approximating  $t(x)$  be of the form:

$$\hat{t}_{ij}(x) = \begin{cases} t_{ij}^0 \left( 1 + a_1 \frac{(x - \theta_{ij}^{(1)})}{m_{ij}} \right), & \text{if } \theta_{ij}^{(1)} \leq x \leq \theta_{ij}^{(2)} \\ \vdots \\ t_{ij}^0 \left( 1 + \sum_{l=1}^n \left( \frac{a_l(\theta_{ij}^{(l)}\theta_{ij}^{(l-1)})}{m_{ij}} \right) + \frac{a_n(x\theta_{ij}^{(n)})}{m_{ij}} \right), & \text{if } \theta_{ij}^{(n)} \leq x, \end{cases}$$

where  $a_l$  is the slope of segment  $l = 1, \dots, n$  of  $\hat{t}$  with  $a_1 \leq \dots \leq a_n < \infty$ , and  $\theta_{ij}^{(l)}$  is a threshold dividing segments on the travel time function for link  $(i, j)$ . In our case, we let  $\theta_{ij}^{(l)} = \theta^{(l)}m_{ij}$  where  $\theta^{(l)}$  is the normalized threshold in the travel time and capacity-normalized function depicted in Fig. 2.

To model this piecewise-affine function in the optimization problem, we introduce the following set of slack variables

$$\varepsilon_{ij}^{(n)} = \max\{0, x_{ij} - \theta_{ij}^{(n)}\}, \quad (9a)$$

$$\vdots$$

$$\varepsilon_{ij}^{(k)} = \max\{0, x_{ij} - \theta_{ij}^{(k)} - \sum_{l=k+1}^n \varepsilon_{ij}^{(l)}\}, \quad (9b)$$

$$\vdots$$

$$\varepsilon_{ij}^{(0)} = \max\{0, x_{ij} - \sum_{l=1}^n \varepsilon_{ij}^{(l)}\}, \quad (9c)$$

where each  $\varepsilon_{ij}^{(k)}$  denotes the extra flow exceeding threshold  $\theta_{ij}^{(k)}$  and up to  $\theta_{ij}^{(k+1)} - \theta_{ij}^{(k)}$ , thus,  $\varepsilon_{ij}^{(k)} \in [0, \theta_{ij}^{(k+1)} - \theta_{ij}^{(k)}]$ . We include these variables in the problem by adding the linear constraints  $\varepsilon_{ij}^{(k)} \geq 0$  and  $\varepsilon_{ij}^{(k)} \geq \theta_{ij}^{(k)} - \sum_{l=k+1}^n \varepsilon_{ij}^{(l)}$ , provided that the objective is a function of  $\varepsilon_{ij}^{(k)}$ .

Using these definitions we can generate a tractable cost function. We focus our attention on an element-wise analysis of the first term (non-convex part) of objective function (5a) using  $\hat{t}$  instead of  $t$  for which we obtain the objective function

$$\hat{J}(x_{ij}, \varepsilon_{ij}) := t_{ij}^0 \left( 1 + \sum_{l=1}^n (a_l \varepsilon_{ij}^{(l)} / m_{ij}) \right) \left( \sum_{k=1}^n \varepsilon_{ij}^{(k)} - x^p \right), \quad (10)$$

which is derived as follows:

$$\hat{t}_{ij}(x_{ij})x_{ij}^u = t_{ij}^0 \left( 1 + \sum_{l=1}^n (a_l \varepsilon_{ij}^{(l)} / m_{ij}) \right) x_{ij}^u, \quad (11a)$$

$$= t_{ij}^0 \left( 1 + \sum_{l=1}^n (a_l \varepsilon_{ij}^{(l)} / m_{ij}) \right) \left( \sum_{k=1}^n \varepsilon_{ij}^{(k)} - x^r - x^p \right), \quad (11b)$$

$$\leq t_{ij}^0 \left( 1 + \sum_{l=1}^n (a_l \varepsilon_{ij}^{(l)} / m_{ij}) \right) \left( \sum_{k=1}^n \varepsilon_{ij}^{(k)} - x^p \right). \quad (11c)$$

In (11b) we express  $x_{ij}^u$  by using a combination of (2) and (9). In the last step (11c), we add the term  $\sum_{l=1}^n t_{ij}^0 a_l \varepsilon_{ij} x_{ij}^r / m_{ij}$  to  $\hat{J}_{ij}$ . By adding this term, we consider a relaxation of the original problem (i.e., an upper bound of  $\hat{J}_{ij}$ ). This modification, which is detailed later (see Remark 1), allows our proposed objective to be a convex quadratic function. Hence, we define the AMoD problem to be

$$\min_{\mathbf{x}^W, \mathbf{x}^R, \boldsymbol{\varepsilon}} \sum_{(i,j) \in \mathcal{A}} \hat{J}(x_{ij}, \boldsymbol{\varepsilon}_{ij}) + \sum_{(i,j) \in \mathcal{A}_R} c_{ij} x_{ij}^r, \quad (12a)$$

$$\text{s.t. (5b) -- (5d)}$$

$$\varepsilon_{ij}^{(k)} \geq \theta_{ij}^{(k)} - x_{ij}, \quad \forall (i, j) \in \mathcal{A}, \quad k = 1, \dots, n, \quad (12b)$$

$$\varepsilon_{ij}^{(k)} \geq 0, \quad \forall (i, j) \in \mathcal{A}, \quad k = 1, \dots, n, \quad (12c)$$

where  $\boldsymbol{\varepsilon} = \{\varepsilon_{ij}^{(j)} \mid (i, j) \in \mathcal{A}, \quad k = 1, \dots, n\}$ .

*Theorem 1:* Problem (12) is a linearly constrained convex Quadratic Program (QP) with linear equality constraints.

*Proof:* We prove this by construction. We show that the  $\mathbf{Q}$  matrix in the QP standard form (i.e.,  $\min_{\mathbf{x}} \mathbf{x}' \mathbf{Q} \mathbf{x}$ , s.t.  $\mathbf{Ax} \leq \mathbf{b}$ ) can be modified to be positive semidefinite (PSD). Note that in (12a) the only quadratic term is of the form  $\varepsilon_{ij}^{(l)} \varepsilon_{ij}^{(k)}$  and its matrix representation (i.e.,  $\boldsymbol{\varepsilon}' \mathbf{Q} \boldsymbol{\varepsilon}$ ) does not guarantee that  $\mathbf{Q}$  is PSD. However, we observe that since we are minimizing, when  $x_{ij} \leq \theta_{ij}^{(k)}$  then  $\varepsilon_{ij}^{(k)} = 0$  and when  $x_{ij} \geq \theta_{ij}^{(k+1)}$  then  $\varepsilon_{ij}^{(k)} = (\theta_{ij}^{(k+1)} - \theta_{ij}^{(k)})$ . Therefore,

$$\varepsilon_{ij}^{(l)} \varepsilon_{ij}^{(k)} = \begin{cases} (\theta_{ij}^{(l+1)} - \theta_{ij}^{(l)}) \varepsilon_{ij}^{(k)}, & \text{if } l < k, \\ \varepsilon_{ij}^{(l)} \varepsilon_{ij}^{(l)}, & \text{if } l = k, \\ \varepsilon_{ij}^{(l)} (\theta_{ij}^{(k+1)} - \theta_{ij}^{(k)}), & \text{if } l > k, \end{cases}$$

where the first case comes from the fact that in order for  $\varepsilon_{ij}^{(k)}$  to be greater than zero, the flow  $x_{ij}$  must have exceeded  $\theta_{ij}^{(l+1)}$  for  $l < k$ . Therefore,  $\varepsilon_{ij}^{(l)}$  is at its maximum value of  $(\theta_{ij}^{(l+1)} - \theta_{ij}^{(l)})$ . The same analogy applies to the third case. Hence, the link-wise objective function of the QP without the rebalancing term is rewritten as

$$\hat{J}_{ij}^{\text{QP}}(x_{ij}^u, \boldsymbol{\varepsilon}_{ij}) = t_{ij}^0 \left( x_{ij}^u + \sum_{l=1}^n \frac{a_l}{m_{ij}} \left( \sum_{k=1}^{l-1} (\theta_{ij}^{(k+1)} - \theta_{ij}^{(k)}) \varepsilon_{ij}^{(l)} \right) + (\varepsilon_{ij}^{(l)})^2 + \sum_{k=l+1}^n (\theta_{ij}^{(k+1)} - \theta_{ij}^{(k)}) \varepsilon_{ij}^{(k)} \right).$$

Using this new formulation, we note that the  $\mathbf{Q}$  matrix is the identity matrix which is PSD and therefore  $J_{ij}^{\text{QP}}$  is convex quadratic using [48, Prop. 3.1.1].  $\square$

In contrast to our previous 3-segment method in [43], we obtain a better approximation of the original travel latency function while formulating the problem as a QP.

*Remark 1:* We observe that the effect of adding  $\sum_{l=1}^n t_{ij}^0 a_l \varepsilon_{ij} x_{ij}^r / m_{ij}$  to (11a) implies taking into account congestion-aware rebalancing routing. However, this congestion-aware routing of the rebalancing vehicles has a lower impact in  $J_{ij}^{\text{QP}}$  than the AMoD users. This is because  $a_0 = 0$  for  $\mathbf{x}^r$  (i.e., the first term in (11c) does not include  $\mathbf{x}^r$ ). Hence, the interpretation of this addition is that the rebalancing flows evaluate the travel latency function with the same structure as the AMoD flows but with  $t_{ij}^0 = 0$ .

*Remark 2:* A relevant trade-off worth noting is on the number of piecewise affine segments used to approximate the travel function. Even though a larger  $n$  will provide better approximations of  $t(\cdot)$ , and hence a better solution to the problem, this implies adding  $|\mathcal{A}|$  additional variables and linear constraints to the formulation.

### B. Linear Relaxation

Seeking a simpler formulation and faster computation performance of (12), we notice that it is possible to relax the QP to a Linear Program (LP) by modifying the only quadratic term in (12a), i.e.,  $(\varepsilon_{ij}^{(l)})^2$ . We approximate this using  $\varepsilon_{ij}^{(l)} \theta_{ij}^{(l+1)}$  and observe that when  $x_{ij} \leq \theta_{ij}^{(l)}$  or  $x_{ij} \geq \theta_{ij}^{(l+1)}$  we recover exactly  $(\varepsilon_{ij}^{(l)})^2$ . However, a gap exists when  $x_{ij} \in (\theta_{ij}^{(l)}, \theta_{ij}^{(l+1)})$  which can be diminished by adding more linear segments to  $\hat{t}(\cdot)$  and consequently decreasing the range of  $(\theta_{ij}^{(l)}, \theta_{ij}^{(l+1)})$ .

*Lemma 1:* Assuming the distance between the break points of the linear segments is uniform, i.e.,  $\theta_{ij}^{(l+1)} - \theta_{ij}^{(l)} = \theta^{(n)}/n$ , for  $l = 1, \dots, n-1$ , then the objective function of the LP formulation approximates the QP objective function by an error upper-bounded by  $a_n(\theta^{(n)})^2/4n^2 \sum_{(i,j) \in \mathcal{A}} t_{ij}^0 m_{ij}$ .

*Proof:* Notice that the maximum total error between the LP and QP is expressed by

$$\sum_{(i,j) \in \mathcal{A}} \max_{l=1,\dots,n} \left\{ \frac{a_l t_{ij}^0}{m_{ij}} ((\theta_{ij}^{(l+1)} - \theta_{ij}^{(l)}) \varepsilon_{ij}^{(l)} - (\varepsilon_{ij}^{(l)})^2) \right\} \quad (14a)$$

$$= \sum_{(i,j) \in \mathcal{A}} \max_{l=1,\dots,n} \left\{ \frac{a_l t_{ij}^0}{m_{ij}} \left( \frac{\theta_{ij}^n}{n} \varepsilon_{ij}^{(l)} - (\varepsilon_{ij}^{(l)})^2 \right) \right\} \quad (14b)$$

$$= \sum_{(i,j) \in \mathcal{A}} \max_{l=1,\dots,n} \left\{ \frac{a_l t_{ij}^0}{m_{ij}} \left( \frac{\theta_{ij}^n}{n} \frac{\theta_{ij}^n}{2n} - \left( \frac{\theta_{ij}^n}{2n} \right)^2 \right) \right\} \quad (14c)$$

$$= \sum_{(i,j) \in \mathcal{A}} \frac{a_n t_{ij}^0}{m_{ij}} \left( \frac{(\theta_{ij}^n)^2}{4n^2} \right) \quad (14d)$$

$$= \frac{a_n (\theta^{(n)})^2}{4n^2} \sum_{(i,j) \in \mathcal{A}} t_{ij}^0 m_{ij}, \quad (14e)$$

where the first equality comes from the fact that we use uniform distances for the piecewise regions, and the second equality comes from the fact that  $\varepsilon_{ij}^{(l)*} = \theta_{ij}^2/2n$  maximizes equation (14b). Finally, the last step selects the last segment  $n$  by observing that  $a_n$  has the steepest slope by assumption.  $\square$

*Theorem 2:* Let the total flow and the capacity of every link be upper-bounded and assume  $a_0 \leq a_1 \leq \dots \leq a_n < \infty$ . Then, as  $n \rightarrow \infty$ , the solution of the LP problem recovers the solution of the QP.

*Proof:* Without loss of generality, let us select the thresholds  $\boldsymbol{\theta}$  in a uniform manner as in Lemma 1. The proof follows immediately after observing in (14e) that, for a bounded  $a_n$ ,  $m_{ij}$  and  $t_{ij}^0$ , the error goes to zero as  $n \rightarrow \infty$ .  $\square$

Interestingly, these two reformulations, QP and LP, together with Theorems 1 and 2 show that a LP can be solved instead of the original convex program described in (5). This LP approximates the solution of the QP which, in turn, approximates the solution of the original problem. These two are asymptotically optimal in the number of segments used to describe the nonlinear function  $t(\cdot)$  in the objective.

### C. Origin-Based Formulation (Flow-Bundling)

So far, we have formulated the problem such that for every OD pair  $\mathbf{w} \in \mathcal{W}$  we introduce  $|\mathcal{A}|$  decision variables. The total number of variables in our QP (or LP) is then  $(n + 1 + |\mathcal{W}|)|\mathcal{A}|$ , which is typically dominated by the number of OD pairs  $|\mathcal{W}|$ . In practice, this number can be very large, sometimes up to  $|\mathcal{V}|^2$ . Hence, solving the problem using the previous formulations may require large memory capabilities.

To mitigate this issue, we leverage similar ideas to [38] which aggregate flows by origin with the objective to reduce the number of variables and constraints of the QP and LP without losing information. This flow aggregation by origin allows to reduce the number of variables to be in the order of  $(n + 1 + |\mathcal{V}|)|\mathcal{A}|$ , which makes the problem significantly faster to solve. Let us denote the set of origin (sources)  $\mathcal{S} = \{w_s \mid d_{(w_s, w_t)}^u > 0, (w_s, w_t) \in \mathcal{W}\}$  and the flow on the network with  $s$  as its source by  $\mathbf{x}^s$ ; the total user flow on a link is then  $\mathbf{x}^u = \sum_{s \in \mathcal{S}} \mathbf{x}^s$  and the set of user origin-link variables be  $\mathbf{x}^S = \{\mathbf{x}^s \mid s \in \mathcal{S}\}$ . For every origin  $s$ , let  $\psi^s(j)$  be the node *imbalance* describing the excess demand or supply at each node. This is

$$\psi^s(j) = \begin{cases} \sum_{t:(s,t) \in \mathcal{W}} -d_{(s,t)}^u, & \text{if } j = s, \\ 0, & \text{if } j \neq s, t, \\ d_{(s,t)}^u, & \text{if } j = t. \end{cases}$$

With this definition in hand, we establish the origin-based problem as follows

$$\min_{\mathbf{x}^S \geq \mathbf{0}, \mathbf{x}^r \geq \mathbf{0}} \sum_{(i,j) \in \mathcal{A}} \hat{J}_{ij}(x_{ij}, \varepsilon_{ij}) + \sum_{(i,j) \in \mathcal{A}_R} c_{ij} x_{ij}^r \quad (16a)$$

$$\text{s.t. } \sum_{i:(i,j) \in \mathcal{A}} x_{ij}^s - \sum_{k:(j,k) \in \mathcal{A}} x_{jk}^s = \psi^s(j), \quad \forall j \in \mathcal{N}, \quad \forall s \in \mathcal{S}, \quad (16b)$$

$$(5c), \quad (12b), \quad (12c), \quad (16b)$$

$$\text{where } x_{ij} = x_{ij}^u + x_{ij}^r + x_{ij}^p = \sum_{s \in \mathcal{S}} x_{ij}^s + x_{ij}^r + x_{ij}^p.$$

We proceed to show that the resulting flows of the solution of the origin-based problem (16) are the same as the OD-based problem (12). To accomplish this, we use the result below.

*Lemma 2:* Let  $\mathbf{x}^{S*}$  be the solution to the origin-based problem (16) and  $\mathbf{x}^{s*}$  the flows associated with origin  $s$ .

Then the subset of arcs  $\mathcal{A}^s = \{(i, j) \mid x_{ij}^s > 0, (i, j) \in \mathcal{A}\}$  with positive flow from origin  $s$  has no direct cycles.

*Proof:* We use contradiction. Assume that there is a cycle  $\mathcal{C}$  where  $(i_1, i_2), (i_2, i_3), \dots, (i_k, i_1) \in \mathcal{A}^s$  and let  $h_i^s$  be the marginal cost (related to the SO solution) of the cheapest path from  $s$  to  $i$ . Further, consider any  $(i, j) \in \mathcal{A}^s$ , which implies that (i) there exists a positive flow path from  $s$  to  $i$  and (ii)  $x_{ij}^s > 0$ . Since we are minimizing a function where  $t_{ij}(x)$  and  $\hat{t}_{ij}(x)$  are strictly positive and monotonically increasing for all  $(i, j) \in \mathcal{A}$ , the path connecting  $s$  to  $i$  has to be a minimum cost path. Assume this cost to be  $T_{si}$  and since  $x_{ij}^s > 0$ , the cost to  $j$  is  $T_{sj} = T_{si} + t_{ij}$ . Note that by definition  $T_{sj} \leq T_{si} + t_{ij}$ . However, if  $T_{sj} < T_{si} + t_{ij}$  then there must exist a lower-cost path to  $j$  than any of those passing through  $i$ . Hence, we must have  $T_{sj} = T_{si} + t_{ij}$  for all the links in  $\mathcal{A}^s$ . Using the fact that all travel times are strictly positive, this implies that for the cycle  $\mathcal{C}$  we have  $T_1 < T_2 < \dots < T_k < T_1$  which is logically inconsistent.  $\square$

**Lemma 3:** The link-flow solution of the origin-based problem (16) is equivalent to the solution of the OD-based problem (12). *i.e.*, for any origin  $s$ , we have  $\sum_{t:(s,t) \in \mathcal{W}} x^{w*} = x^{s*}$ .

*Proof:* Similar to [38], we prove this by construction and use the flow decomposition algorithms and results in [49, Thm. 3.5]. We begin by decomposing the origin-based solution  $x^s$  of origin  $s$  to a set of acyclic paths  $\mathcal{P}^s$  such that  $\mathbf{x}^{(w_s, w_t)} = \mathbf{x}_t^s$  where  $\mathbf{x}_t^s$  is the acyclic decomposed flow from  $\mathbf{x}^s$  going from  $s$  to  $t$ . We conclude the proof by observing that the origin-based solution has no cycle (by Lemma 2) and the fact that it is possible to decompose the problem to flows using [49, Thm. 3.5].  $\square$

Therefore, using the result of Lemma 3, we can restate the network model in terms of the origin-based flows which reduces the size of the model, memory requirements, and solution time.

#### D. Disjoint Strategy

We have discussed methods to solve the SO routing and rebalancing problem jointly. Yet another approach is to tackle these two problems separately. We initiate the method by setting  $\mathbf{x}^r = 0$  and then we repeatedly solve the routing problem followed by the load-balancing problem. Mathematically, this is to first solve

$$\min_{\mathbf{x}^W \geq 0} \sum_{(i,j) \in \mathcal{A}} t_{ij}(x_{ij})x_{ij}^u, \quad \text{s.t. (5b)}, \quad (17)$$

followed by using the optimal  $\mathbf{x}^{u*}$  to solve

$$\min_{\mathbf{x}^r \geq 0} \mathbf{c}' \mathbf{x}^r, \quad \text{s.t. (5c)}. \quad (18)$$

It is relevant to highlight that this strategy is interesting given its fast computation. Problem (17) is a constrained nonlinear program (NLP) which can be solved using any of the typical algorithms for the TAP, for example Frank-Wolfe or TAPAS; and problem (18) is a LP with  $|\mathcal{V}|$  variables.

#### IV. AMoD IN MIXED TRAFFIC

We have not yet discussed how to address the nested problem (7) which considers the interaction between the

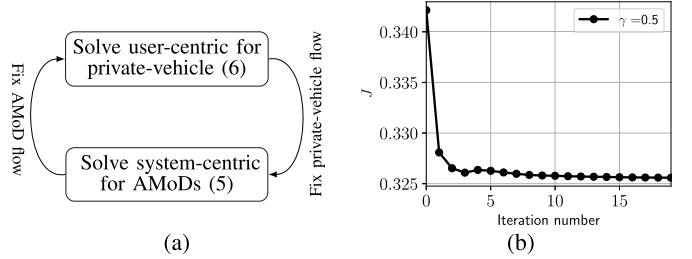


Fig. 3. (a): A sketch of the procedure for solving the bilevel problem (7). (b): An example of the total cost converging for an AMoD penetration rate of 0.5 on the NYC sub-network.

fleet of AMoDs vehicles and self-interested private vehicles. We employ the framework in [23] which applies a sequential approach (diagonalization scheme [24], [25]) to find an equilibrium between the AMoD and private flows (Fig. 3).

Rather than addressing the bilevel problem (7), we solve (6) for the private vehicles and (5) for the AMoD fleet (using any of the methods in the previous section) and iterate until convergence. Namely, for a private vehicle demand  $\mathbf{g}^u$  we solve  $\mathbf{x}^p = \text{TAP}(\mathbf{g}^p, \mathbf{0})$ . Thereafter, we solve (5) for AMoD demand  $\mathbf{g}^u$  with fixed input  $\mathbf{x}^p$  (the output of the earlier solved TAP). Since private vehicles were not aware of the AMoD flow in the system while finding their routes, we re-solve the TAP by considering a fixed AMoD flow equal to  $\mathbf{x}^u + \mathbf{x}^r$ , *i.e.*, we solve  $\mathbf{x}^p = \text{TAP}(\mathbf{g}^p, \mathbf{x}^u + \mathbf{x}^r)$ . Further, we iterate this process until it converges. An example is shown in Fig. 3b.

In this paper, we do not establish theoretical results on the stability or uniqueness of the players (AMoD fleet and private vehicles) equilibria. These results are hard to achieve due to the non-separability of the cost function regarding the players' strategies as pointed out in [24]. Still, empirically, this sequential approach always converges in a few iterations.

**Remark 3:** Notice that when employing this iterative method, some of the parameters can be updated. In particular, if one uses the disjoint strategy in Sec. III-D to solve the routing and rebalancing problem, one could update the  $\mathbf{c}$  vector at each subsequent iteration by the calculated travel times  $\mathbf{t}(\mathbf{x})$  at the current iteration. When doing this, one obtains a more precise cost function by weighting vector  $\mathbf{c}$  with the updated travel times.

#### V. ROUTE RECOVERY STRATEGIES

All methods discussed thus far solve the routing and rebalancing problem by choosing  $\mathbf{x}^u$  and  $\mathbf{x}^r$  that minimize a performance metric. Even if this flow solution allows us to assess the network deficiencies and to plan for infrastructure improvements, flows do not give explicit routes to a given vehicle. Therefore, we need to extract the routes to implement the desired flow-based solution we derive. An advantage of the proposed models in (12) and (16) in contrast to classical link-based TAP is the fact that they allow for tracing and recovering the routes (or paths). We present simple and distributed algorithms to recover the routes from the OD-based or origin-based solutions presented in Sec. III, as well as an algorithm to retrieve the rebalancing routes.

**Algorithm 1** Route-Recovery for a Specific OD Pair

```

1: procedure ROUTERECOVERY( $\mathbf{A}$ ,  $\mathbf{d}_w$ ,  $\mathbf{x}^{w*}$ ,  $\xi$ )
2:   Initialize:  $x_{ij} \leftarrow d_w \mathbf{1}_{(i,j) \in \text{shortest route for } w}$ 
3:   while  $\|\mathbf{x} - \mathbf{x}^{w*}\| > \xi$  do
4:      $\mathcal{R}_w \leftarrow \text{append next shortest path}$ 
5:      $\pi_w \leftarrow \text{solve (19)}$ 
6:   end while
7: end procedure

```

**A. AMoD User Flow**

1) *OD-Pair Model:* Let the optimal solution of the routing and rebalancing problem be  $(\mathbf{x}^{\mathcal{W}^*}, \mathbf{x}^{r*})$  and denote with  $\mathcal{R}_w$  a set of routes for OD pair  $w$ . For each  $w$ , we let  $\pi \in [0, 1]^{|\mathcal{R}_w|}$  be a vector with elements denoting the fraction of vehicle flow routed through route  $i \in \mathcal{R}_w$ . We denote with  $\mathbf{A}$  the route-link incidence matrix of  $\mathcal{R}_w$ . With these definitions, we provide a column-generation approach in which we find the routes of an OD-pair by sequentially solving the linear program

$$\min_{\pi \in [0, 1]} \|\mathbf{A}\pi d_w - \mathbf{x}^{w*}\| \quad (19a)$$

$$\text{s.t. } \pi' \mathbf{1} = 1, \quad (19b)$$

where the product  $\mathbf{A}\pi d_w$  is equal to the estimated link flow induced by routing  $d_w \pi_i$  flow through each route. The constraint ensures that the vector  $\pi$  is a probability distribution.

The problem of selecting which routes to include in  $\mathcal{R}_w$  (column selection) is yet to be addressed. We use the greedy approach of adding the next shortest route to  $\mathcal{R}_w$  and re-solving problem (19). To terminate the algorithm, we employ a user-defined parameter  $\xi$  (as shown in Alg. 1). It is worth pointing out that this procedure can run in parallel for each OD pair. For uncongested networks, we expect it to converge fast. This is because when there is little congestion, the majority of vehicles will be routed through the shortest paths, which are the first ones to be added to the set  $\mathcal{R}_w$ . Finally, note that this formulation is only available if we have information on  $\mathbf{x}^{w*}$  for all  $w \in \mathcal{W}$ .

2) *Origin-Based Model:* Let  $\mathbf{x}^{s*}$  be the solution of (16) and let  $\mathcal{T}_s = \{j \mid \psi_s(j) < 0, j \in \mathcal{V}\}$  be the set of destinations (targets) from origin  $s$ . Let  $\psi_s(j)$  be the node imbalance of node  $j$  of the origin-based flows initialized at  $s$ . For each origin  $s$ , one can decompose its OD-flow solution by solving the following LP:

$$\min_{\{\mathbf{x}^t\}_{t \in \mathcal{T}_s} \geq 0} \mathbf{t}' \mathbf{x} \quad (20a)$$

$$\text{s.t. } \sum_{i:(i,j) \in \mathcal{A}} x_{ij} - \sum_{k:(j,t) \in \mathcal{A}} x_{jt} = \psi_s(t), \quad \forall j \in \mathcal{V}, \quad (20b)$$

$$\sum_{i:(i,j) \in \mathcal{A}} x_{ij}^t - \sum_{k:(j,k) \in \mathcal{A}} x_{kj}^t \geq 0, \quad \forall j \in \mathcal{V} \setminus \{s\}, \quad (20c)$$

$$\sum_{i:(i,s) \in \mathcal{A}} x_{is}^t - \sum_{k:(s,k) \in \mathcal{A}} x_{sk}^t = \psi_s(t), \quad (20d)$$

$$\mathbf{x}^{s*} - \mathbf{x} = \mathbf{0}. \quad (20e)$$

Here  $\mathbf{x}$  is the origin-based flow (equivalent to  $\mathbf{x}^s$ ) defined as  $\mathbf{x} = \sum_{t \in \mathcal{T}_s} \mathbf{x}^t$ . The first constraint, (20c), takes care of

demand satisfaction and flow conservation. The second constraint, (20c), considers flow conservation but allows certain target nodes to have excess flow, allowing them to be a destination. Constraint (20d) ensures that the decision variables are designed for that specific origin  $s$  by ensuring that the required flow is leaving that node. (20e) then forces the solution to be equal to the origin-based flows. Finally, the objective (20a) is defined with the purpose of breaking ties in case multiple combinations of flows can satisfy the constraints (e.g. cycles).

Notice that as a result of Lemma 3, this problem is always feasible and recovers the OD-based solution. Once this is established, we could use Alg. 1 to find the path-based solution. Problem (20) is stated as a linear program that could be solved in parallel for each origin-based solution  $s$ , therefore, we expect this optimization process to be computationally efficient.

**B. Rebalancing Flows**

The problem of finding the paths of the rebalancing flows is more complex than that of finding the AMoD routes. This is because we have no information about their origin and destinations. Rather, the only information available is the aggregated link flows that the rebalancing vehicles are taking to minimize (5a) and comply with the load-balancing constraint (5c). Hence, a first step to recover the paths is to calculate the rebalancing node *imbalances*  $\phi(j)$  for every node  $j$  defined over the available rebalancing solution  $\mathbf{x}^r$ :

$$\phi(j) = \sum_{i:(i,j) \in \mathcal{A}_R} x_{ij}^r - \sum_{k:(j,k) \in \mathcal{A}_R} x_{jk}^r.$$

We define a rebalancing origin to be a deficit flow node, and its set  $\mathcal{S}_r = \{j \mid \phi(j) < 0, j \in \mathcal{A}_R\}$ ; similarly, the rebalancing destination set is defined as  $\mathcal{T}_r = \{j \mid \phi(j) > 0, j \in \mathcal{A}_R\}$ . Notice that these definitions are made in  $\mathcal{A}_R$  and not in  $\mathcal{A}$ , as the rebalancing vehicles only exist in  $G_R$ . Then we aim to recover an OD rebalancing solution by solving

$$\min_{\{\mathbf{x}^s\}_{s \in \mathcal{S}_r} \geq 0} \mathbf{t}' \mathbf{x} \quad (21a)$$

$$\text{s.t. } \sum_{i:(i,j) \in \mathcal{A}_R} x_{ij} - \sum_{k:(j,k) \in \mathcal{A}_R} x_{jk} = \phi(j), \quad \forall j \in \mathcal{V}_R, \quad (21b)$$

$$\sum_{i:(i,s) \in \mathcal{A}_R} x_{is}^s - \sum_{k:(s,k) \in \mathcal{A}_R} x_{sk}^s = \phi(s), \quad \forall s \in \mathcal{S}_r, \quad (21c)$$

$$\sum_{i:(i,j) \in \mathcal{A}_R} x_{ij}^s - \sum_{k:(j,k) \in \mathcal{A}_R} x_{kj}^s \geq 0, \quad \forall j \in \mathcal{V}_R \setminus \{s\}, \quad \forall s \in \mathcal{S}_r, \quad (21d)$$

$$\mathbf{x} - \mathbf{x}^r = \mathbf{0}, \quad (21e)$$

where we define  $\mathbf{x} = \sum_{s \in \mathcal{S}_r} \mathbf{x}^s$  and  $\mathbf{x}^r$  is the available link flow solution of (5a). Notice that the model follows the same intuition as (20). Constraint (21b) takes care of the total flow conservation of the rebalancing flow, constraint (21c) ensures that, for each origin variables  $\mathbf{x}^s$ , the outflow of node  $s$  is equal to the excess of vehicles. Constraint (21d) allows any node different than  $s$  to be a potential destination of the

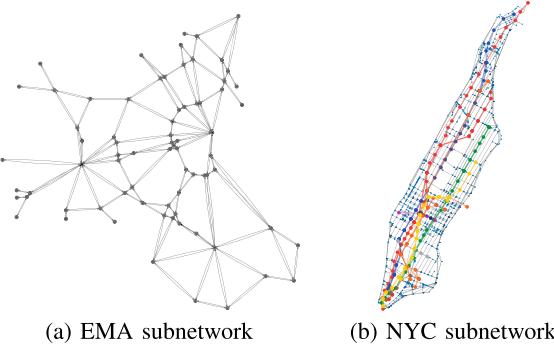


Fig. 4. Subnetworks used for the experiments. Black lines indicate road links while colored lines indicate subway lines.

rebalancing flow. Finally, (21e) ensures that the aggregated rebalancing flows by origin match the rebalancing flow obtained in the AMoD user problem.

Once we have decomposed the rebalancing flow by origins, we have for each rebalance origin  $s$  an origin-based rebalancing flow. Since now we have the flows available in an origin-based form, we can apply (20) in parallel for each  $s \in \mathcal{S}_r$  to decompose to an OD-flow solution, and finally use Alg. 1 to recover the routes.

*Remark 4:* For both of (20) and (21) it is possible to dualize the last constraint (i.e., penalize  $\|\mathbf{x} - \mathbf{x}^r\|$  on the cost function). This makes the optimization less restrictive and improves the solution time by lowering the quality of the solution. It is difficult to estimate exactly what the impact of this dualization would be in terms of efficiency. However, for low-traffic networks, we expect (20) and (21) to be faster to solve as we expect the total flow on every link will belong to fewer OD pairs. Conversely, when dealing with high-traffic scenarios, the total flow on a link might be composed of many OD pairs, making the problem harder (slower) to decompose. In practice we have observed that for low-traffic networks less than 3 routes per OD pair are enough to obtain an accurate solution, whereas for high-traffic cases, the number of routes required for good solutions are in the order of 6 to 8. Still, the problems as stated in this paper can be solved to optimality.

## VI. NUMERICAL RESULTS AND CASE STUDIES

To validate our proposed routing algorithms, we consider two data-driven case studies on sub-networks of Eastern Massachusetts interstate highways (EMA) and New York City (NYC). The EMA road network (Fig. 4a) consists of 74 nodes, 258 links, and 1113 OD pairs, and it captures the dynamics in the context of suburban/urban mobility. Complementary to EMA, the NYC network focuses on urban mobility. The NYC topology was constructed using OpenStreetMaps [50] and contains 3317 arcs, 1351 nodes. The OD demand was built using historical data taxi rides (courtesy of the New York Taxi Commission [51]) that occurred on March 1, 2012, between 18:00 and 20:00 hrs which accounts for 8658 OD pairs.

### A. Convergence of the Approximated Model

Our first experiment shows empirically our results of Theorem 1 and the observation that as  $n$  increases, the approximation of  $\hat{t}(\cdot)$  to  $t(\cdot)$  becomes tighter and therefore the

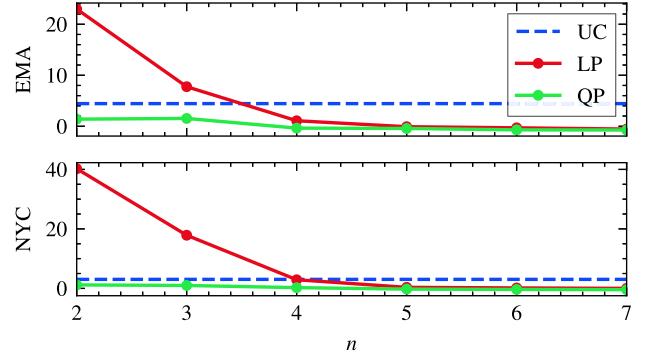


Fig. 5. Deviation in percentage terms between the approximated model and the optimal solution of the non-rebalancing SO problem (baseline). UC indicates how much the solution of the UC deviates from the SO. This gap between the UC and SO models is referred to as the *Price of Anarchy* [15].

QP and LP problems approximate the original problem more accurately. To generate this experiment, we consider a problem with no rebalancing (not including the rebalancing constraints) and with no exogenous flow (i.e.,  $\mathbf{x}^c = 0$ ). This is exactly the SO formulation of the TAP for which we use the Frank-Wolfe algorithm to find its solution (we solve the UC problem using the same method). Thereafter we solve the QP and LP versions of the CARS $n$  model for different values of  $n$  and observe that, as we increase  $n$ , the objective of CARS $n$  converges to the objective of the SO. For example, for both networks shown in Fig. 5, we observe that for  $n = 6$  the objective of the approximated models QP and LP are very close to the SO solution.

### B. Joint Vs. Disjoint Solution

This experiment aims to compare the solution of the joint and disjoint formulation of the problem. That is, solving (12) against the disjoint method in Sec. III-D. We compare this by showing the improvement (ratio between the value of the objective functions) of the joint over the disjoint approach. For EMA and NYC we take account of an improvement in the objective of 3.85% and 0.91%, respectively. Moreover, we consider the case of NYC network with a higher demand, which we simulate by multiplying the demand vector  $\mathbf{g}$  by 2. The improvement of the joint formulation over the disjoint model for this demand level is 5.85%. These results highlight the achievable benefits, especially for high demand scenarios, of jointly solving the routing and rebalancing problems, rather than separately.

### C. System-Optimal Routing and Rebalancing Trade-off

Considering the existence of selfish privately-owned vehicles and centrally-controlled AMoD vehicles, we analyze the trade-off that exists between system-optimal AMoD routing and the additional traffic due to AMoD rebalancing in terms of average travel times. We tackle the bilevel Problem (7) following the iterative methodology presented in Sec. IV. We use different *penetration rates* of AMoD customers with respect to the total demand, i.e., a penetration rate of 0.3 will indicate that 30% of the total demand uses the AMoD service while the rest use private vehicles. More specifically, we let  $\gamma \in [0, 1]$  be the penetration rate and  $\mathbf{g}$  the total OD demand.

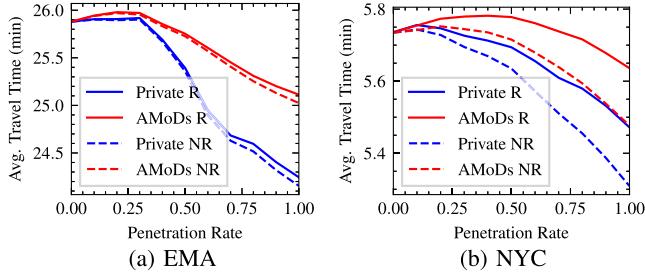


Fig. 6. Travel times for AMoD users, private vehicles and all vehicles (total) for different penetration rates of AMoDs in the network. “R” stands for an approach that considers rebalancing while “NR” does not.

We assume that  $\mathbf{g}^u = \gamma \mathbf{g}$  and  $\mathbf{g}^p = (1 - \gamma)\mathbf{g}$  are the AMoD’s and private vehicles’ demand, respectively. However, different demand separation criteria can be readily implemented in this framework.

The general trend in Fig. 6 shows that as the penetration rate of AMoD increases, its overall travel time decreases. More interestingly, not only the AMoD travel time is reduced, but also the private vehicles’ travel time. This is because the collaborative routing decisions of the AMoD fleet lessen the traffic intensity on congested roads, which consequently allow private vehicles to travel faster. For low penetration rates, the addition of AMoDs could be detrimental as the negative effect of the new rebalancing flow on travel times is higher than the positive effect generated by better routing. For EMA, the impact of rebalancing is negligible, and increasing the percentage of AMoD users in the network allows to reduce travel time by up to 3%. For NYC, we observe that rebalancing indeed is detrimental to low penetration rates, but as the percentage of SO vehicles increases, social routing improves travel times for both AMoD users and private vehicles. Yet, in general, the impact of rebalancing on the system-level performance depends on the network topology, and on the symmetry and intensity of the OD demand distribution.

#### D. Intermodal AMoD

We study the impact of intermodal SO routing against UC private vehicle routing for the NYC network. We consider high congestion levels and run the experiments by multiplying the demand distribution vector  $\mathbf{g}$  by a factor of 1.5 (see details of the demand in the online repository [52]). Similar to our last experiment, we run the analysis for different penetration rates. We assume that AMoD users are able to take public transit (subway), walk, or bike towards their destination and switch between modes in their route. In contrast to the AMoD users, we limit the flexibility of private vehicles to exclusively use the road network (no subway, biking or walking) due to parking constraints. The top row of plots on Fig. 7 display, on the left, the travel time for the two user types as the penetration rate of AMoD users increases and, on the right, the modal distribution of the total kilometers traveled. The top row shows the results when only taxi-type service is offered to AMoD users (no subway, walking or biking). We observe that the extra rebalancing flow increases the overall travel times of the system more than what SO routing can reduce. This result confirms the fact that pure vehicle-based MoD systems

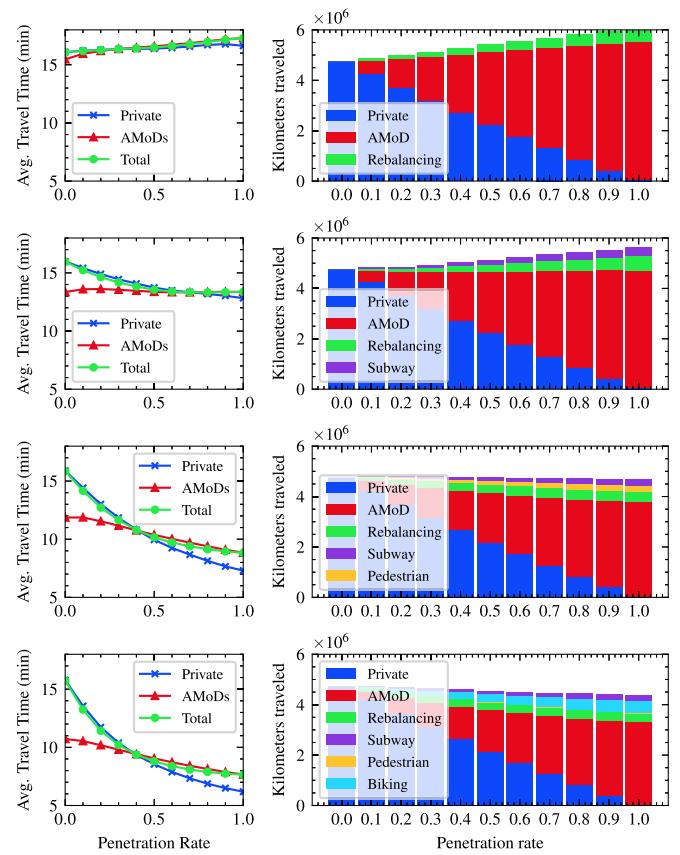


Fig. 7. System performance with alternative modes of transport for a relatively high-demand scenario in NYC (we increase demand by a factor of 1.5). The first column of plots show the average travel time for different AMoD penetration rates while the second row depicts the miles traveled per mode of transportation for each penetration rate.

can have detrimental effects on the overall travel time [53]. The subsequent plots show that by considering the flexibility of other modes of transportation, AMoD mobility can reduce traffic congestion. The second row of plots in Fig. 7 includes a public transit option, the third row adds a pedestrian option (6 km/h), and the last one also considers biking (10 km/h) as an option.<sup>2</sup> In general, we see that the more modes of transportation are offered, the lower the travel times for everyone. In addition, when new options for mobility are offered, AMoD users could reach lower travel times than private vehicles, something which is impossible to achieve when only taxi-rides are available (due to the assumption on UC routing). This happens because they are more flexible and their overall transportation capacity is larger than the available capacity for private vehicles. However, at almost 100% penetration rates, it still seems that being selfish is benefited, raising interesting questions on how to incentivize users to act in a system-centric fashion. Finally, by comparing the first three rows of plots in Fig. 7, we can make an important observation: If merely a tiny fraction of flow is accessible via subway or walking, travel times are reduced by almost 50%.

<sup>2</sup>For biking, we include a set of constraints in the same spirit as (5c) but for the bike layer. This ensures the balance between the incoming and outgoing flow of bikes at each node which goes in line with the dynamics of bike sharing systems [54].

TABLE I  
INTERMODAL AMoD RESULTS FOR DIFFERENT TRAFFIC INTENSITIES

Demand	Pen. Rate	Avg. Travel Time (min)		I-AMoD Modal Distribution				
		I-AMoD	Private Veh	Veh	Reb	Bike	Sub	Ped
1	0	5.2	5.7	80%	15%	0%	5%	0%
	0.5	5.2	5.4	81%	14%	0%	5%	0%
	1	5.0	5.0	82%	13%	0%	4%	0%
1.5	0	7.5	8.8	69%	23%	2%	6%	0%
	0.5	7.0	6.9	74%	17%	4%	5%	0%
	1	6.3	5.7	78%	13%	4%	5%	0%
2	0	10.7	15.8	52%	28%	12%	7%	1%
	0.5	9.1	8.5	68%	13%	12%	6%	0%
	1	7.7	6.2	75%	8%	11%	6%	0%



Fig. 8. Example of the SO routes connecting an OD pair. Green and red dots represent origin and destinations, respectively. Solid lines portray traveling flow in the road network while dotted lines describe flow traveling via subway.

To account for more traffic intensities, Table I presents the results for an AMoD system with taxi-type (Veh), subway (Sub), pedestrian (Ped), and biking (Bike) layers when demand is multiplied by a factor of 1, 1.5, (corresponding to the last subplots of Fig 7) and 2. The Table shows results for the overall travel times and modal distributions of the I-AMoD kilometers traveled for penetration rates equal to 0, 50%, and 100%. In general we can claim that the higher the congestion, the higher the benefit in travel times due to the enlarged capacity resulting from intermodal options. In addition, we see that subway and biking options are critical to improve travel times.

In conclusion, we observe that while pure AMoD systems might decrease the system-level performance due to the additional congestion resulting from rebalancing, intermodal centralized-routing can significantly improve the overall travel times. Especially at high levels of demand, we see that, while SO intermodal routing can significantly improve travel times, it comes with the social dilemma that, from a UC perspective, being selfish would still be optimal.

#### E. Route Recovery Example

We show the applicability of our route-recovery strategies presented in Section V. We implement the distribute version of the route-recovery algorithm described in Section V-A.2 on the

solution flows of the origin-based problem (16). We compute the routes using a commercial laptop with 8 cores for which we recover the routes in the order of 30 seconds to one minute, making it accessible for real-time implementation. Fig. 8 shows the different SO routes connecting a single OD pair. The left plot shows the recommended routes which only include taxi-type service. Furthermore, the right plot shows an intermodal route composed of taking a taxi (solid lines) and the subway (dotted line).

## VII. CONCLUSION

In this paper, we proposed a methodology to optimize the routes and rebalancing policies of a congestion-aware intermodal Autonomous Mobility-on-Demand (AMoD) system when it interacts with exogenous private traffic. To address the issue of non-convexity for this problem, we used a piecewise affine approximation of the travel latency function and proved that as the number of piecewise affine segments increases, the solution to the problem converges to the solution of the relaxed original problem. Using examples with the Eastern Massachusetts Area (EMA) and New York City (NYC) networks, (*i*) we empirically showed that the piecewise affine relaxation is asymptotically optimal, (*ii*) we captured the benefits of centrally controlling an intermodal AMoD system under mixed traffic conditions when different modes of transportation are available, (*iii*) we measured the advantage of using the approximated joint method versus a method that separately optimizes the routing and rebalancing policies, (*iv*) we revealed the existing trade-off between extra rebalancing flow and smart routing decisions, and (*v*) we tested the applicability of our proposed route-recovery algorithms in a real case study using the NYC network.

This paper opens the field for the following extensions: First, we would like to use these methodologies to solve a larger class of problems characterized as Traffic Assignment Problem with *side constraints* (TAPSC), i.e., Traffic Assignment Problems (TAPs) with arbitrary constraints such as the link-capacitated TAP [55]. Second, we are interested in leveraging our route-recovery strategies for real-time routing. Finally, we would like to devise pricing and incentive schemes to align the interests of selfish users with the system optimum and realize the full potential of smart intermodal mobility systems [56]–[58].

## ACKNOWLEDGMENT

The authors thank D. Sverdrin-Lisker and Dr. I. New for proofreading this manuscript. This article solely reflects the opinions and conclusions of its authors and not NSF, TRI, or any other entity.

## REFERENCES

- [1] J. Bates and D. Leibling, “Spaced out,” Perspect. Parking Policy, vol. 9, Jul. 2012. [Online]. Available: [https://www.racfoundation.org/assets/rac\\_foundation/content/downloadables/spaced\\_out-bates\\_leibling-jul12.pdf](https://www.racfoundation.org/assets/rac_foundation/content/downloadables/spaced_out-bates_leibling-jul12.pdf)
- [2] W. J. Mitchell, C. E. Borroni-Bird, and L. D. Burns, *Reinventing the Automobile: Personal Urban Mobility for the 21st Century*. Cambridge, MA, USA: MIT Press, 2010.
- [3] United States DOT. (Jul. 2020). *Mobility on Demand (MOD)*. [Online]. Available: <https://www.its.dot.gov/factsheets/mobilityondemand.htm>

- [4] S. Banerjee, R. Johari, and C. Riquelme, "Pricing in ride-sharing platforms: A queueing-theoretic approach," in *Proc. ACM Conf. Econ. Comput.*, 2015, p. 639.
- [5] K. Bimpikis, O. Candogan, and D. Saban, "Spatial pricing in ride-sharing networks," *Oper. Res.*, vol. 67, no. 3, pp. 744–769, May 2019.
- [6] A. J. Hawkins. (Nov. 2020). *Robotaxis Get the Green Light for Paid Rides in California*. [Online]. Available: <https://www.theverge.com/2020/11/23/21591045/california-robotaxi-paid-rides-cpuc-permits>
- [7] M. Toh. (Dec. 2020). *Self-Driving Robotaxis are Taking Off in China*. [Online]. Available: <https://edition.cnn.com/2020/12/03/tech/autonomous-robotaxi-china-intl-hnk/index.html>
- [8] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numer. Math.*, vol. 1, no. 1, pp. 269–271, Dec. 1959.
- [9] R. Bellman, "On a routing problem," *Quart. Appl. Math.*, vol. 16, no. 1, pp. 87–90, 1958.
- [10] G. Ramalingam and T. Reps, "An incremental algorithm for a generalization of the shortest-path problem," *J. Algorithms*, vol. 21, no. 2, pp. 267–305, Sep. 1996.
- [11] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Syst. Sci. Cybern.*, vol. SSC-4, no. 2, pp. 100–107, Jul. 1968.
- [12] A. Hertz, G. Laporte, and M. Mittaz, "A tabu search heuristic for the capacitated arc routing problem," *Oper. Res.*, vol. 48, no. 1, pp. 129–135, Feb. 2000.
- [13] C. W. Ahn and R. S. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations," *IEEE Trans. Evol. Comput.*, vol. 6, no. 6, pp. 566–579, Dec. 2002.
- [14] T. Roughgarden and É. Tardos, "Bounding the inefficiency of equilibria in nonatomic congestion games," *Games Econ. Behav.*, vol. 47, no. 2, pp. 389–403, May 2004.
- [15] J. Zhang, S. Pourazarm, C. G. Cassandras, and I. C. Paschalidis, "The price of anarchy in transportation networks: Data-driven evaluation and reduction strategies," *Proc. IEEE*, vol. 106, no. 4, pp. 538–553, Apr. 2018.
- [16] M. Patriksson, *The Traffic Assignment Problem: Models and Methods*, vol. 54, no. 2. New York, NY, USA: Dover, 1994.
- [17] M. Frank and P. Wolfe, "An algorithm for quadratic programming," *Naval Res. Logistics Quart.*, vol. 3, nos. 1–2, pp. 95–110, 1956.
- [18] L. J. LeBlanc, R. V. Helgason, and D. E. Boyce, "Improved efficiency of the Frank–Wolfe algorithm for convex network programs," *Transp. Sci.*, vol. 19, no. 4, pp. 445–462, 1985.
- [19] M. Florian, J. Guálat, and H. Spiess, "An efficient implementation of the 'partan' variant of the linear approximation method for the network equilibrium problem," *Networks*, vol. 17, no. 3, pp. 319–339, 1987.
- [20] H. Bar-Gera, "Traffic assignment by paired alternative segments," *Transp. Res. B, Methodol.*, vol. 44, nos. 8–9, pp. 1022–1046, Sep. 2010.
- [21] O. Jahn, R. H. Möhring, A. S. Schulz, and N. E. Stier-Moses, "System-optimal routing of traffic flows with user constraints in networks with congestion," *Oper. Res.*, vol. 53, no. 4, pp. 600–616, 2005.
- [22] D. A. Lazar, S. Coogan, and R. Pedarsani, "Capacity modeling and routing for traffic networks with mixed autonomy," in *Proc. IEEE 56th Annu. Conf. Decis. Control (CDC)*, Dec. 2017, pp. 5678–5683.
- [23] A. Houshmand, S. Wollenstein-Betech, and C. G. Cassandras, "The penetration rate effect of connected and automated vehicles in mixed traffic routing," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 1755–1760.
- [24] P. T. Harker, "Multiple equilibrium behaviors on networks," *Transp. Sci.*, vol. 22, no. 1, pp. 39–46, Feb. 1988.
- [25] H. Yang, X. Zhang, and Q. Meng, "Stackelberg games and multiple equilibrium behaviors on networks," *Transp. Res. B, Methodol.*, vol. 41, no. 8, pp. 841–861, Oct. 2007.
- [26] J. Alonso-Mora, S. Samaranayake, A. Wallar, E. Frazzoli, and D. Rus, "On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment," *Proc. Nat. Acad. Sci. USA*, vol. 114, no. 3, pp. 462–467, 2017.
- [27] R. Chen and C. G. Cassandras, "Optimal assignments in mobility-on-demand systems using event-driven receding horizon control," *IEEE Trans. Intell. Transp. Syst.*, early access, Oct. 20, 2020, doi: 10.1109/TITS.2020.3030218.
- [28] X. Bei and S. Zhang, "Algorithms for trip-vehicle assignment in ride-sharing," in *Proc. AAAI Conf. Artif. Intell.*, vol. 18, 2018, pp. 3–9.
- [29] M. Pavone, S. L. Smith, E. Frazzoli, and D. Rus, "Robotic load balancing for mobility-on-demand systems," *Int. J. Robot. Res.*, vol. 31, no. 7, pp. 839–854, Jun. 2012.
- [30] R. Zhang and M. Pavone, "Control of robotic mobility-on-demand systems: A queueing-theoretical perspective," *Int. J. Robot. Res.*, vol. 35, nos. 1–3, pp. 186–203, Jan. 2016.
- [31] K. Spieser, S. Samaranayake, W. Gruel, and E. Frazzoli, "Shared-vehicle mobility-on-demand systems: A fleet operator's guide to rebalancing empty vehicles," in *Proc. Annu. Meeting Transp. Res. Board*, 2016, pp. 1–15, Paper 16-5987.
- [32] R. M. A. Swaszek and C. G. Cassandras, "Load balancing in mobility-on-demand systems: Reallocation via parametric control using concurrent estimation," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 2148–2153.
- [33] M. W. Levin, K. M. Kockelman, S. D. Boyles, and T. Li, "A general framework for modeling shared autonomous vehicles with dynamic network-loading and dynamic ride-sharing application," *Comput., Environ. Urban Syst.*, vol. 64, pp. 373–383, Jul. 2017.
- [34] S. Hörl, C. Ruch, F. Becker, E. Frazzoli, and K. W. Axhausen, "Fleet control algorithms for automated mobility: A simulation assessment for Zurich," in *Proc. Annu. Meeting Transp. Res. Board*, 2018, pp. 1–15, Paper 18-02171.
- [35] F. Rossi, R. Zhang, Y. Hindy, and M. Pavone, "Routing autonomous vehicles in congested transportation networks: Structural properties and coordination algorithms," *Auton. Robots*, vol. 42, no. 7, pp. 1427–1442, 2018.
- [36] M. Salazar, F. Rossi, M. Schiffer, C. H. Onder, and M. Pavone, "On the interaction between autonomous mobility-on-demand and the public transportation systems," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, Nov. 2018, pp. 2262–2269.
- [37] M. Salazar, N. Lanzetti, F. Rossi, M. Schiffer, and M. Pavone, "Intermodal autonomous mobility-on-demand," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3946–3960, Sep. 2020.
- [38] F. Rossi, R. Iglesias, M. Alizadeh, and M. Pavone, "On the interaction between autonomous mobility-on-demand systems and the power network: Models and coordination algorithms," *IEEE Trans. Control Netw. Syst.*, vol. 7, no. 1, pp. 384–397, Mar. 2020.
- [39] S. Belakaria, M. Ammous, L. Smith, S. Sorour, and A. Abdel-Rahim, "Multi-class management with sub-class service for autonomous electric mobility on-demand systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 7155–7159, Jul. 2019.
- [40] United States. Bureau of Public Roads, *Traffic Assignment Manual for Application With a Large, High Speed Computer*, vol. 37. US Department of Commerce, Bureau of Public Roads, Office of Planning, Urban Planning Division, 1964.
- [41] K. Solovey, M. Salazar, and M. Pavone, "Scalable and congestion-aware routing for autonomous mobility-on-demand via Frank–Wolfe optimization," in *Proc. Robot., Sci. Syst.*, Freiburg im Breisgau, Germany, Jun. 2019.
- [42] M. Salazar, M. Tsao, I. Aguiar, M. Schiffer, and M. Pavone, "A congestion-aware routing scheme for autonomous mobility-on-demand systems," in *Proc. 18th Eur. Control Conf. (ECC)*, Jun. 2019, pp. 3040–3046.
- [43] S. Wollenstein-Betech, A. Houshmand, M. Salazar, M. Pavone, C. G. Cassandras, and I. C. Paschalidis, "Congestion-aware routing and rebalancing of autonomous mobility-on-demand systems in mixed traffic," in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst. (ITSC)*, Sep. 2020, pp. 2293–2299.
- [44] M. J. Beckmann, C. B. McGuire, and C. B. Winsten, *Studies in the Economics of Transportation*. New Haven, CT, USA: Yale Univ. Press, 1955.
- [45] S. Wollenstein-Betech, C. Sun, J. Zhang, and I. C. Paschalidis, "Joint estimation of OD demands and cost functions in transportation networks from data," in *Proc. IEEE 58th Conf. Decis. Control (CDC)*, Dec. 2019, pp. 5113–5118.
- [46] J. G. Wardrop, "Road paper. Some theoretical aspects of road traffic research," *Proc. Inst. Civil Eng.*, vol. 1, no. 3, pp. 325–362, 1952.
- [47] Y. A. Korilis, A. A. Lazar, and A. Orda, "Achieving network optima using Stackelberg routing strategies," *IEEE/ACM Trans. Netw.*, vol. 5, no. 1, pp. 161–173, Feb. 1997.
- [48] D. Bertsekas, *Nonlinear Programming*, 3rd ed. Belmont, MA, USA: Athena Scientific, 2016.
- [49] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms and Applications*. Upper Saddle River, NJ, USA: Prentice-Hall, 1993.
- [50] OpenStreetMap. (2017). *Planet Dump Retrieved From*. [Online]. Available: <https://planet.osm.org> and <https://www.openstreetmap.org>
- [51] NYC Taxi and Limousine Commission. (2020). *Taxi and Limousine Commission Trip Record Data*. [Online]. Available: <https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page>
- [52] S. Wollenstein-Betech. (Jan. 2020). *Mixed-Traffic-AMod-Route-Rebalance*. [Online]. Available: <https://github.com/salomonw/mixed-traffic-amod-route-rebalance>

- [53] E. Fitzsimmons and W. Hu, "The downside of ridehailing: More New York City gridlock," *New York Times*, Mar. 6, 2017.
- [54] R. M. A. Swaszek and C. G. Cassandras, "Receding horizon control for station inventory management in a bike-sharing system," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 1, pp. 407–417, Jan. 2020.
- [55] T. Larsson and M. Patriksson, "An augmented Lagrangean dual algorithm for link capacity side constrained traffic assignment problems," *Transp. Res. B, Methodol.*, vol. 29, no. 6, pp. 433–455, Dec. 1995.
- [56] S. Wollenstein-Betech, I. C. Paschalidis, and C. G. Cassandras, "Joint pricing and rebalancing of autonomous mobility-on-demand systems," in *Proc. 59th IEEE Conf. Decis. Control (CDC)*, Dec. 2020, pp. 2573–2578.
- [57] S. Wollenstein-Betech, I. C. Paschalidis, and C. G. Cassandras, "Optimal operations management of mobility-on-demand systems," *Frontiers Sustain. Cities*, vol. 3, pp. 1–13, Jul. 2021.
- [58] M. Salazar, D. Paccagnan, A. Agazzi, and W. P. M. H. Heemels, "Urgency-aware optimal routing in repeated games through artificial currencies," *Eur. J. Control*, Jul. 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0947358021000832>



**Salomón Wollenstein-Betech** (Student Member, IEEE) received the B.S. degree in industrial engineering from the Tecnológico de Monterrey in 2015. He is currently pursuing the Ph.D. degree with the Division of Systems Engineering, Boston University (BU). He is a member of the CODES Lab and the NOC Lab. His research focuses on data-driven assessment, optimization, and control of networks with a focus on intelligent transportation systems. He was a recipient of the Dean's Fellowship awarded by BU.



mobility systems. Both his master's and Ph.D. theses were recognized with the ETH Medal. He was granted the Best Student Paper Award at the 2018 Intelligent Transportation Systems Conference.



**Arian Houshmand** received the B.S. degree in mechanical engineering from Sharif University of Technology, Tehran, Iran, in 2014, the M.S. degree in mechanical engineering from the University of Cincinnati, Cincinnati, OH, USA, in 2016, and the Ph.D. degree in systems engineering from Boston University, Boston, MA, USA, in 2020. He is currently working as a Motion Planning and Control Software Engineer at Zoox, Foster City, CA, USA. His research interests include motion planning and collaborative teleguidance of autonomous vehicles.



**Marco Pavone** (Member, IEEE) received the Ph.D. degree in aeronautics and astronautics from Massachusetts Institute of Technology in 2010. He is currently an Associate Professor in aeronautics and astronautics with Stanford University and the Director of autonomous vehicle research at NVIDIA. Prior to this, he was a Research Technologist with the Robotics Section, NASA Jet Propulsion Laboratory. His main research interests are in the development of methodologies for the analysis, design, and control of autonomous systems, with an emphasis on self-driving cars, autonomous aerospace vehicles, and future mobility systems. He was a recipient of a number of awards, including the Presidential Early Career Award for Scientists and Engineers from President Barack Obama, the ONR YIP Award, the NSF CAREER Award, and the NASA Early Career Faculty Award. He was identified by the American Society for Engineering Education (ASEE) as one of America's 20 most promising investigators under the age of 40. He is serving as an Associate Editor for *IEEE Control Systems Magazine*.



**Ioannis Ch. Paschalidis** (Fellow, IEEE) received the Diploma degree in electrical and computer engineering from the National Technical University of Athens in 1991 and the M.S. and Ph.D. degrees in electrical engineering and computer science from Massachusetts Institute of Technology in 1993 and 1996, respectively. Since September 1996, he has been with Boston University, Boston, MA, where he is currently a Professor and a Data Science Fellow, having appointments with the Department of Electrical and Computer Engineering, the Division of Systems Engineering, the Department of Biomedical Engineering, and the Faculty of Computing and Data Sciences. He is the Director of the Center for Information and Systems Engineering. He has held visiting appointments with MIT and Columbia University. His current research interests include the fields of systems and control, networks, machine learning, optimization, computational biology, and medical informatics. He was a recipient of the National Science Foundation CAREER Award in 2000, several best paper and best algorithmic performance awards, and the 2014 IBM/IEEE Smarter Planet Challenge Award. He was an Invited Participant at the 2002 Frontiers of Engineering Symposium, organized by U.S. National Academy of Engineering and the 2014 U.S. National Academies Keck Futures Initiative Conference. From 2013 to 2019, he was the Founding Editor-in-Chief for the *IEEE TRANSACTIONS ON CONTROL OF NETWORK SYSTEMS*.



**Christos G. Cassandras** (Life Fellow, IEEE) received the B.S. degree from Yale University, New Haven, CT, USA, in 1977, the M.S.E.E. degree from Stanford University, Stanford, CA, USA, in 1978, and the M.S. and Ph.D. degrees from Harvard University, Cambridge, MA, USA, in 1979 and 1982, respectively. He was with ITP Boston, Inc., Cambridge, from 1982 to 1984, where he was involved in the design of automated manufacturing systems. From 1984 to 1996, he was a Faculty Member with the Department of Electrical and Computer Engineering, University of Massachusetts Amherst, Amherst, MA, USA. He is currently a Distinguished Professor of engineering with Boston University, Brookline, MA, USA, the Head of the Division of Systems Engineering, and a Professor of electrical and computer engineering. He specializes in the areas of discrete event and hybrid systems, cooperative control, stochastic optimization, and computer simulation, with applications to computer and sensor networks, manufacturing systems, and transportation systems. He has authored over 450 refereed articles in these areas and six books. He was the President of the IEEE Control Systems Society in 2012. He is a fellow of the International Federation of Automatic Control (IFAC). He is a member of Phi Beta Kappa and Tau Beta Pi. He was a recipient of several awards, including the 2011 IEEE Control Systems Technology Award, the 2006 Distinguished Member Award of the IEEE Control Systems Society, the 1999 Harold Chestnut Prize (IFAC Best Control Engineering Textbook), the 2011 Prize and the 2014 Prize for the IBM/IEEE Smarter Planet Challenge Competition, the 2014 Engineering Distinguished Scholar Award at Boston University, several honorary professorships, the 1991 Lilly Fellowship, and the 2012 Kern Fellowship. From 1998 to 2009, he was the Editor-in-Chief of the *IEEE TRANSACTIONS ON AUTOMATIC CONTROL*. He serves on several editorial boards and has been a guest editor for various journals.