ON THE INTERACTION BETWEEN AUTONOMOUS MOBILITY-ON-DEMAND SYSTEMS AND THE BUILT ENVIRONMENT: MODELS AND LARGE SCALE COORDINATION ALGORITHMS

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF
AERONAUTICS AND ASTRONAUTICS
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Federico Rossi
March 2018

Abstract

Autonomous Mobility-on-Demand systems (that is, fleets of self-driving cars offering on-demand transportation) hold promise to reshape urban transportation by offering high quality of service at lower cost compared to private vehicles. However, the impact of such systems on the infrastructure of our cities (and in particular on traffic congestion and the electric power network) is an active area of research. In particular, Autonomous Mobility-on-Demand (AMoD) systems could greatly increase traffic congestion due to additional "rebalancing" trips required to re-align the distribution of available vehicles with customer demand; furthermore, charging of large fleets of electric vehicles can induce significantly stress in the electric power network, leading to high electricity prices and potential network instability.

In this thesis, we build analytical tools and algorithms to *model* and *control* the interaction between AMoD systems and our cities. We open our work by exploring the interaction between AMoD systems and urban congestion. Leveraging the theory of network flows, we devise models for AMoD systems that capture endogenous traffic congestion and are amenable to efficient optimization. These models allow us to show the key theoretical result that, under mild assumptions that are substantially verified for U.S. cities, AMoD systems *do not* increase congestion compared to privately-owned vehicles for a given level of customer demand if empty-traveling vehicles are properly routed. We leverage this insight to design a real-time congestion-aware routing algorithm for empty vehicles; microscopic agent-based simulations with New York City taxi data show that the algorithm significantly reduces congestion compared to a state-of-the-art congestion-agnostic rebalancing algorithm, resulting in 22% lower wait times for AMoD customers. We then devise a randomized congestion-aware routing algorithm for *customer-carrying* vehicles and prove rigorous analytical bounds on its performance. Preliminary results based on New York City taxi data show that the algorithm could yield a further reduction in congestion and, as a result, 5% lower service times for AMoD customers.

We then turn our attention to the interaction between AMoD fleets with electric vehicles and the power network. We extend the network flow model developed in the first part of the thesis to capture the vehicles' state-of-charge and their interaction with the power network (including charging

and the ability to inject power in the network in exchange for a payment, denoted as "vehicle-to-grid"). We devise an algorithmic procedure to losslessly reduce the size of the resulting model, making it amenable to efficient optimization, and test our models and optimization algorithms on a hypothetical deployment of an AMoD system in Dallas-Fort Worth, TX with the goal of maximizing social welfare. Simulation results show that coordination between the AMoD system and the power network can reduce electricity prices by over \$180M/year, with savings of \$120M/year for local power network customers and \$35M/year for the AMoD operator. In order to realize such benefits, the transportation operator must cooperate with the power network: we prove that a pricing scheme can be used to enforce the socially optimal solution as a general equilibrium, aligning the interests of a self-interested transportation operator and self-interested power generators with the goal of maximizing social welfare. We then design privacy-preserving algorithms to compute such coordination-promoting prices in a distributed fashion. Finally, we propose a receding-horizon implementation that trades off optimality for speed and demonstrate that it can be deployed in real-time with microscopic simulations in Dallas-Fort Worth.

Collectively, these results lay the foundations for congestion-aware and power-aware control of AMoD systems; in particular, the models and algorithms in this thesis provide tools that will enable transportation network operators and urban planners to foster the positive externalities of AMoD and avoid the negative ones, thus fully realizing the benefits of AMoD systems in our cities.

To Valentina.

Acknowledgments

First and foremost, I wish to thank my advisor, Prof. Marco Pavone. Marco instilled in me an appreciation for ambitious research, long-term vision, and good writing. I thank him for giving me the opportunity to work with him and the Autonomous Systems Lab and for being a great advisor, mentor, and teacher.

I thank Professor Mykel Kochenderfer and Professor Mac Schwager for serving on the reading committee for this thesis. Their feedback on the content and presentation of this work was invaluable. I also acknowledge Professor Andrea Goldsmith and Professor Ram Rajagopal for serving on my defense committee, and thank them for their thoughtful feedback.

I am grateful to my co-authors, in particular Prof. Mahnoosh Alizadeh, Mauro Salazar, Ramon Iglesias, Dr. Rick Zhang, Dr. Saptarshi Bandyopadhyay, and Yousef Hindy. Research is a team effort: working with them has been a privilege, and I have learned much from each of our discussions. In particular, Saptarshi was a wonderful mentor during my internship at the Jet Propulsion Laboratory in 2017: I do look forward to collaborating with him and the rest of the fabulous team at JPL in the coming years.

At the personal level, I thank each and every member, past and present, of the Autonomous Systems Laboratory. I have truly enjoyed working alongside each of them, and I cherish the memory of our boisterous conversations in the lab.

I thank Rachel and Steve of Hog Island for teaching me about perseverance, season after season.

An enormous thank you goes to my *mellyn*, my friends from away, and in particular Flavio, Federica, Marzia, and Selene, for granting me their friendship through the years and the distance. Even though we only meet in person a few times a year, every time I am with them I feel at home.

My parents, Tere and Giuseppe, taught me the perseverance and grit that allowed me to take on the challenge of a Ph.D.: for that I am truly grateful.

Finally, I wish to thank my wife Valentina. Vale was at my side throughout these years; she cheered with me when things were good, and unfailingly cheered me up when things were not going so well. She supported me throughout this path and always offered good advice, relentless encouragement, and unconditional love. For that I owe her immense gratitude.

I acknowledge the support of the National Science Foundation (NSF) under CAREER Award CMMI-1454737, the Office of Naval Research, Science of Autonomy Program under Contract N00014-15-1-2673, and the Toyota Research Institute (TRI). This thesis solely reflects my opinions and conclusions and not those of NSF, ONR, TRI or any other entity.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This thesis studies the interaction between Autonomous Mobility-on-Demand systems and the built environment, that is, our cities. Autonomous Mobility-on-Demand (AMoD) is a novel proposed mode of urban transportation enabled by the two emerging technologies of autonomous driving and one-way car-sharing. In an AMoD system, a fleet of centrally routed self-driving vehicles services on-demand transportation requests in an urban environment. The key difference between current mobility-on-demand systems (e.g. Uber, Lyft, or DiDi) and AMoD is that a self-driving fleet is amenable to *centralized* control, enabling deployment of fleet-wide policies to match customers with available vehicles, compute routes, and (in the case of electric vehicles) schedule charging; in addition, AMoD systems offer lower operational costs compared to human-driven MoD systems.

AMoD systems hold promise to deliver a number of benefits to our cities, including increased access to mobility for those unable or unwilling to drive, lower pollution (due both to massive electrification and to shifting of pollution from the tailpipe to the chimney stack), a smaller overall number of vehicles (with positive effects on the cost and pollution resulting from vehicle manufacturing and disposal), and reduced need for parking spaces in city centers [Mitchell et al., 2010]. The potential impact of AMoD systems on the electric power network is especially notable: fleets of electric vehicles, in coordination with the power network, could significantly increase adoption of distributed generation (e.g. rooftop solar) by synchronizing their charging activity with peaks in renewable generation and, potentially, returning power to the network with vehicle-to-grid (V2G) schemes at times of low demand for transportation and high demand for power [Mitchell et al., 2010].

Such benefits require *coordination* between AMoD systems and existing urban infrastructure. Indeed, in absence of coordination, AMoD systems could impose significant negative externalities on our cities! Studies have shown that mass adoption of electric vehicles (such as would be enabled by AMoD systems) and uncoordinated charging could lead to massive increases in electricity prices and power network instability [Hadley and Tsvetkova, 2009], which could only be mitigated with very significant infrastructure investments. In addition, the shift from private mobility to AMoD could cause

significant increases in traffic congestion due to "rebalancing" trips necessary to realign empty vehicles with the distribution of customer demand [Levin et al., 2017, Maciejewski and Bischoff, 2017].

In this thesis, we lay the groundwork to address these issues by building models and optimization algorithms that allow for the efficient, large-scale control of AMoD systems and account for their interactions with the built environment. We focus our attention on two problems: (i) urban congestion and (ii) the interaction between electric-powered AMoD systems and the electric power network.

1.1 AMoD systems and the built environment

In this section, we formally define the problem of controlling an AMoD system and its interaction with the built environment.

In the AMoD problem, a fleet of single-occupancy self-driving vehicles services transportation requests on a road network. Vehicles travel along road links, modeled as edges, which are subject to traffic congestion. Transportation requests arrive in the system according to a stochastic process; each request originates and ends at a given location in the road network. The goal of the AMoD operator is to service each request by (i) matching it with an available vehicle, (ii) routing the vehicle to the request's origin location for pickup, and (iii) routing the customer-carrying vehicle to the request's destination. Additionally, the AMoD operator preemptively *rebalances* empty vehicles from customers' destinations to areas of high demand in order to ensure good vehicle availability and short wait times for future requests. If the autonomous vehicles are electric-powered, the AMoD operator must also compute a charging schedule for the vehicles to ensure that they have sufficient battery charge to perform trips.

Optimization objectives can include availability of vehicles upon customer arrival, average request wait time, average request service time (i.e. the sum of the wait time and travel time), overall vehicle-miles traveled, and fuel and energy cost.

In general, control of AMoD systems *in isolation* belongs to the class of networked, heterogeneous, stochastic decision problems with uncertain information [Pavone, 2015]. The interaction of these systems with the built environment further increases the complexity of the problem by introducing couplings between AMoD systems, urban congestion, and the operations of the electric power network.

AMoD and urban congestion Large fleets of autonomous vehicles operating on-demand transportation services can cause significant congestion due to additional vehicle-miles traveled [Levin et al., 2016, Maciejewski and Bischoff, 2017]. Crucially, congestion is an *endogenous* phenomenon for AMoD systems: centralized routing policies for large autonomous fleets can significantly affect congestion patterns, which, in turn, affects travel times and therefore routing choices. Thus,

on the one hand, routing policies that do not account for endogenous congestion effects can result in poor performance both for the AMoD system and for all other users of the road network; on the other hand, the AMoD operator can actively *control* and *mitigate* congestion with congestion-aware routing (e.g. by adopting longer and less congested routes for empty vehicles), resulting in benefits for both AMoD customers and other road users.

AMoD and the electric power network AMoD systems are especially well-suited for electric vehicles (EVs). With individually-owned EVs, each vehicle must have an appropriate state of charge for their owner's desired travels; conversely, in AMoD systems, the fleet operator can match transportation requests with any vehicle with an adequate state of charge. This additional signficant degree of freedom can be leveraged to design fleet-wide smart charging policies that benefit both the AMoD operator (in terms of high quality of service and lower electricity expenditure) and the electric power network. At the distribution level, smart charging can reduce congestion in the power network and thus enable increased penetration of distributed generation, e.g. rooftop solar; at the transmission level, shifting charging loads in time and space can flatten the demand for power, reducing use of expensive "peaker" power plants and enabling increased adoption of renewable generation.

EVs equipped with inverters can also return power to the power network (a mode of operation known as vehicle-to-grid, or V2G): this can further contribute to the control of the power network and provide the AMoD operator with an additional revenue source.

1.2 Contribution

Currently unavailable control-theoretical tools are required in order to foster the positive externalities that AMoD systems impose on the built environment and mitigate the negative ones. In this thesis, we propose such tools, with particular attention to the problems of traffic congestion and the interaction with the power network.

Our contribution is threefold. First, we propose models for AMoD systems that capture the interaction between fleets of self-driving cars and the built environment. In particular, we focus on the impact of AMoD systems on the road network (that is, the congestion problem), and on their interaction with the power network (denoted as Power-in-the-loop AMoD, or P-AMoD). Second, we propose control algorithms for these systems. In particular, we leverage optimization and model-predictive control techniques to synthesise control algorithms that optimise the operations of the AMoD system and account for their externalities; we also design pricing schemes to enforce socially-optimal strategies in presence of self-interested transportation and power network operators, and we provide privacy-preserving algorithms to compute such prices in a distributed fashion. Third, we validate these models and algorithms with case studies. In particular, we study the impact of an AMoD system on congestion in NYC, and the impact of a fleet of an electric AMoD fleet in Dallas

Fort Worth on the Texas power network.

In detail, in Chapters 3 and 4, we explore the interaction between AMoD systems and traffic congestion. In Chapter 3, we propose a network flow model that captures the operations of AMoD systems and the effect of endogenous congestion, and is amenable to efficient optimization. We study the structural properties of the model: our key theoretical result shows that, under mild structural assumptions that are substantially verified for major U.S. cities, there always exists a way of routing empty vehicles without increasing congestion compared to the case where only passenger-carrying trips are performed . That is, in stark contrast with common belief, AMoD systems *do not increase* congestion compared to private cars if properly routed, assuming that demand for transportation remains constant. We leverage this insight to propose efficient algorithmic tools for receding-horizon congestion-aware control of AMoD fleets (and in particular for congestion-aware routing of empty vehicles), and we demonstrate the effectiveness of this approach with microscopic agent-based simulations based on real-world taxi data in Manhattan. In Chapter 4, we further leverage the analytical insights from the congestion-aware AMoD model to design efficient randomized algorithms for congestion-aware routing of *customer-carrying* vehicles. We provide rigorous bounds on the expected performance of the proposed algorithm, and we assess its effectiveness with a case study in Manhattan.

In Chapter 5, we turn our attention to the interaction between AMoD systems and the electric power network. We propose a network flow model that captures the operations of the AMoD system, the vehicles' charge level and the effect of their charging schedule on the electric power network (modeled through a DC model), the time-varying customer demand for transportation and power, and traffic congestion. We propose algorithms that leverage the structural properties of the model to losslessly reduce its size and make it amenable to efficient optimization. Jointly optimizing the operations of a Power-in-the-loop AMoD system requires cooperation between the AMoD system operator and the power network operator: we prove that any optimal solution to the P-AMoD problem can also be enforced as a general equilibrium through pricing, and we show that a dual decomposition algorithm can be used to compute the market-clearing prices in a distributed fashion without requiring the transportation operator to disclose private information (e.g., transportation requests). We then show, through large-scale numerical experiments based on real-world data in Dallas-Fort Worth, TX, that cooperation between the AMoD system and the power network results in considerable benefits (in terms of power prices) both for the transportation operator and for the power network operator, with savings in the order of \$180M/year shared between local power network customers (\$120M/year) and the AMoD operator (\$35M/year), and no negative impact on AMoD customers. Finally, we propose a receding-horizon P-AMoD controller and show that it is able to realize many of the benefits of AMoD in a real-time setting through agent-based microscopic simulations in Dallas-Fort Worth.

1.3 Structure

The rest of this thesis is structured as follows.

In Chapter 2, we outline some tools that will be used in the rest of the thesis; in particular, we focus on the theory of network flow algorithms and its application to modeling and control of AMoD fleets.

In Chapters 3 and 4, we explore the interaction between AMoD systems and traffic congestion. In Chapter 3, we propose a network flow model for congestion-aware control of AMoD systems, study its fundamental properties, and propose a congestion-aware routing algorithm for rebalancing vehicles; in Chapter 4, we further leverage the model to propose a congestion-aware routing algorithm for customer-carrying as well as rebalancing vehicles.

In Chapter 5, we turn our attention to the interaction between AMoD systems and the electric power network. We extend the network flow model to capture the interaction between AMoD systems and the power network, propose a technique to losslessly reduce the size of the resulting model, and devise pricing schemes to enforce the social optimum as a general equilibrium; we then assess the impact of P-AMoD on a case study in Dallas-Fort Worth, TX, and propose a real-time implementation.

Finally, in Chapter 6, we draw our conclusions and outline directions for future research.

The full text of this thesis along with supplementary material and errata is available online at `https://www.federico.io/dissertation`.

Chapter 2

Preliminaries: network flow models

In this thesis we leverage network flow models to model the operations of AMoD systems and their interaction with the built environment. In this chapter, we give a brief overview of elements of network flow theory and provide a high-level description of their use in modeling AMoD systems; we also outline some of the disadvantages of such models (namely, their deterministic structure and continuum approximation) and discuss possible tools to overcome them.

2.1 Network flow and multi-commodity flow problems

Consider a network $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the node set and $\mathcal{E}$ is the edge set. Each edge $(u, v) \in \mathcal{E}$ has a capacity $c(u, v)$ denoting the maximum amount of flow that can traverse each edge.

A commodity must be transported from an origin locations to a destination location in the network. The commodity is described by the tuple (s, t, λ), where $s \in \mathcal{V}$ is the origin node, $t \in \mathcal{V}$ is the destination node, and $\lambda \in \mathcal{R}^+$ is the intensity (or demand) for the commodity, that is, the amount of the commodity that must be transported.

A network flow is a function $f : \mathcal{E} \mapsto \mathbb{R}_{\geq 0}$ that satisfies the following equations [Ahuja et al., 1993]

$$\sum_{u:(u,v)\in\mathcal{E}} f(u,v) + 1_{v=s}\lambda = \sum_{w:(v,w)\in\mathcal{E}} f(v,w) + 1_{v=t}\lambda, \qquad \text{for all } v \in \mathcal{V} \quad \text{(continuity)}, \tag{2.1a}$$

$$f(u,v) \leq c(u,v), \qquad \text{for all } (u,v) \in \mathcal{E} \quad \text{(capacity)}, \tag{2.1b}$$

where 1_x denotes the indicator function of the Boolean variable $x = \{\text{true, false}\}$ (1_x equals one if x is true).

Equation (2.1a) ensures that the network flow originates at the commodity's origin location, arrives at the commodity's destination, and is conserved at all other nodes; Equation (2.1b) ensures

that no edge carries an amount of flow higher than its capacity.

The definition of network flows can readily be generalized to flows with multiple sources and sinks. Consider a commodity with multiple origins $\{s_i\}_i$ with intensities $\{\lambda_i\}_i$, and multiple destinations $\{t_j\}_j$ with intensities $\{\lambda_j\}_j$. Then Equation (2.1a) is replaced by

$$\sum_{u:(u,v)\in\mathcal{E}} f(u,v) + \sum_i 1_{v=s_i}\lambda_i = \sum_{w:(v,w)\in\mathcal{E}} f(v,w) + \sum_j 1_{v=t_j}\lambda_j, \qquad \text{for all } v \in \mathcal{V}. \tag{2.2}$$

Note that Equation 2.2 can be satisfied at all nodes only if $\sum_i \lambda_i = \sum_j \lambda_j$, that is if the sum of the intensities of all origins equals the sum of intensities of all destinations.

In the multi-commodity flow problem, a set of M commodities $\{(s_m, t_m, \lambda_m)\}_{m\in\mathcal{M}}$ must be delivered through the network. Each commodity is associated with a flow $\{f_m(u,v)\}_{(u,v)}$, and flows satisfy the constraints:

$$(2.2) \qquad \text{for all } m \in \mathcal{M} \tag{2.3a}$$

$$\sum_{m\in\mathcal{M}} f_m(u,v) \le c(u,v), \qquad \text{for all } (u,v) \in \mathcal{E} \tag{2.3b}$$

In a minimum cost multicommodity flow problem, each edge $(u,v) \in \mathcal{E}$ is associated with a cost $a(u,v)$, which denotes the expense required to send one unit of flow across the edge. The minimum cost multicommodity flow problem entails finding a feasible flow that satisfies:

$$\begin{aligned} &\underset{f_m(\cdot,\cdot)}{\text{minimize}} && \sum_{(u,v)\in\mathcal{E}} a(u,v) \sum_{m\in\mathcal{M}} f_m(u,v) \\ &\text{subject to} && (2.3). \end{aligned} \tag{2.4}$$

2.1.1 Optimal solution of minimum-cost multi-commodity flow problems

Network flow models (and, in particular, minimum-cost multi-commodity flow problems in the form of (2.4)) have linear structure: that is, Problem (2.4) can be cast as a linear program with $O(M|\mathcal{E}|)$ variables. Interior point algorithms can solve the problem in $O((M|\mathcal{E}|)^{3.5})$ time [Karmarkar, 1984]; specialized combinatorial algorithms (e.g. [Goldberg et al., 1990, Leighton et al., 1995], [Goldberg et al., 1998]) can also leverage the structure of the Min-MCF problem, often offering better performance. For single-commodity flows, network simplex algorithms [Orlin, 1997, Tarjan, 1997] can be employed to further reduce the computational complexity of the problem. In practical applications, multi-commodity flow problems with millions of variables can be solved efficiently on commodity hardware [Mittelmann, 2016].

2.2 Network flow problems for AMoD systems

In this thesis, we model AMoD systems in the framework of multi-commodity network flow problems. In this section, we give a high-level description of the simple time-invariant model that will be used in Chapters 3 and 4. A more detailed descriptions is presented in Section 3.2.2 (congestion-aware, time-invariant model) and 5.2.1 (power-in-the-loop, congestion-aware, time-varying model).

We model a road network as a capacitated graph $G(\mathcal{V}, \mathcal{E})$. Nodes $v \in \mathcal{V}$ represent intersections and locations for customer trip origins and destinations. Edges $(u, v) \in \mathcal{E}$ represent road links.

Congestion is modeled as a constraint on the vehicle flow that a road link can accommodate: specifically, a function $c(u, v) : \mathcal{E} \mapsto N_{\geq 0}$ denotes the capacity of each link (in vehicles per unit time). All vehicles on a link are assumed to travel at the free-flow speed if the congestion constraint is satisfied. The free-flow travel time across the road link is denoted by $t(u, v) : \mathcal{E} \mapsto \mathbb{R}_{>0}$. This threshold congestion model represents a tradeoff between accuracy and amenability to efficient optimization. In the rest of the thesis, we show that the model is very well-suited for *synthesis* of congestion-aware control policies; detailed analysis models (including static models [Wardrop, 1952], queuing-based models [Osorio and Bierlaire, 2009], and simulation-based models [Treiber et al., 2000]) are available to validate the performance of the resulting policies.

Customer requests are denoted by the collection of tuples $\{(s_m, t_m, \lambda_m)\}$, where $s_m \in \mathcal{V}$ is the origin of the request, $t_m \in \mathcal{V}$ is the destination of the request, and $\lambda_m \in \mathbb{N}_{>0}$ is the number of passengers wishing to travel from s_m to t_m in one unit of time, henceforth called the intensity of the request. Transportation requests are assumed to be stationary and deterministic, that is, λ_m is constant in time. The set of transportation requests is denoted as $\mathcal{M} = \{(s_m, t_m, \lambda_m)\}_m$ with $M = |\mathcal{M}|$.

We model customer routes and rebalancing routes as *flows* of customers and vehicles on the graph $G(\mathcal{V}, \mathcal{E})$. Specifically, the set of customer routes for customer request m is modeled by a network flow $\{f_m(u, v)\}_{(u,v)\in\mathcal{E}}$.

Rebalancing routes realign the vehicle distribution with the distribution of passenger departures by moving empty vehicles to passengers' origins; accordingly, the rebalancing routes for empty vehicles are modeled by a network flow $\{f_R(u, v)\}_{(u,v)\in\mathcal{E}}$ with origins corresponding to the destinations of customer flows and destinations corresponding to the origins of customer flows.

2.2.1 Network flows and customer routes

Our goal is to provide tools for AMoD operators to control fleets of self-driving vehicles: to this end, network flows must be converted to routes that individual vehicles can follow. A customer route is an ordered list of edges $\{(s_m, u), (u, v), \ldots, (w, t_m)\}$ that forms a path connecting a customer origin s_m with a customer destination t_m. Each customer route can be associated with an intensity λ corresponding to the rate of customers (or, equivalently, vehicles) traveling on the route. Analogously,

a rebalancing route is a path connecting a customer destination t_m with a customer origin s_l (for rebalancing paths, the origin and destination may belong to different customers), associated with a rate of vehicles traveling on that route. Vehicles follow customer routes to transport customers from their respective origins to their destinations; rebalancing routes realign the vehicle distribution with the distribution of passenger departures by moving empty vehicles to passengers' origins.

For a given origin s, destination t, and intensity λ, we define a *path flow* as a function $f^p(u,v) : \mathcal{E} \mapsto \mathbb{R}_{\geq 0}$ that satisfies Equation 2.3a and only assigns positive flow to edges belonging to a path p going from s to t; in other words, there exists $p = \{(s,u),(u,v),\dots,(w,t)\}$ such that $f^p(u,v) > 0 \Leftrightarrow (u,v) \in p$. A customer or rebalancing route can be equivalently described as a path flow of intensity $\lambda = 1$ by assigning value $f(u,v) = 1$ to the edges contained in the route and $f(u,v) = 0$ otherwise. Hence, path flows *equivalently* represent customer and rebalancing routes, a fact we will extensively leverage in this thesis.

Conversely, any network flow can be decomposed in a collection of path flows. Specifically, the flow decomposition algorithm ([Ahuja et al., 1993], reported below as Algorithm 1) can be used to decompose *any* network flow with no cycles (and, in particular, any network flow that is the solution to a minimum-cost multi-commodity flow problem in the form of (2.4)) in a collection of path flows such that the sum of the path flows' intensities equals the intensity of the network flow.

This establishes a correspondence between network flows and vehicle paths: any vehicle path can be represented as a network flow (specifically, a path flow), and any network flow can be represented as a collection of vehicle paths, each associated with a vehicle rate.

Algorithm 1: Flow decomposition algorithm [Ahuja et al., 1993, Sec. 3.5].

Input: A network flow $\{f_m(u,v)\}_{(u,v)}$
A set $\mathcal{O}$ of origin nodes $s \in \mathcal{O}$
A set $\mathcal{D}$ of destination nodes $t \in \mathcal{D}$
Output: A list of paths $\{Path_i\}_i$. Each entry $Path_i$ is a collection of consecutive edges $\{(s,u),(u,v),\dots(w,t)\}$.
A list of path intensities $\{\lambda_i\}_i$
procedure FLOWDECOMPOSITION($\{f_m(u,v)\}_{(u,v)}$)
i=1;
while $\sum_{s\in\mathcal{O}} \sum_{v\in V} f_m(s,v) > 0$ **do**
$Path_i \leftarrow$ a path from some $s \in \mathcal{O}$ to some $t \in \mathcal{D}$ containing only edges (u,v) with $f_m(u,v) > 0$
$\lambda_i = \min_{(u,v)\in Path_i} f_m(u,v)$
for all $(u,v) \in Path_i$ **do**
$f_m(u,v) = f_m(u,v) - \lambda_i$
$i = i + 1$
return the path flows $\{\{Path_i\},\{\lambda_i\}\}_i$

2.2.2 Inclusion of additional constraints

The linear structure of network flow models makes them amenable to inclusion of additional constraints. In particular, these models can be extended to capture the interaction with other systems with linear or convex structure (e.g. the power network). Furthermore, multi-origin and multi-destination flows can be used to encode degrees of freedom such as variable departure times and variable charge levels both for customer-carrying and empty trips. We extensively explore such extensions in the context of the interaction with the power network in Section 5.2 .

2.3 Limitations of network flow models

2.3.1 Network flow models and stochasticity

Network flow models are fully deterministic: in particular, the demand for transportation services is assumed to be known and deterministic. This is at odds with the mode of operation of AMoD systems, where customer requests are stochastic and not known in advance. To accommodate stochastic and unknown customer demand, we employ model-predictive control [Borrelli et al., 2017]: a finite-horizon optimization problem is periodically solved based on (i) currently available information and (ii) expected estimate future demand, and the first time step of the proposed solution is implemented.

In addition, network flow models can often be shown to track the *expectation* of an underlying stochastic process. In particular, in [Iglesias et al., 2016, Iglesias et al., 2019], the authors propose a BCMP queuing-theoretical model of an AMoD system that captures the stochasticity of the customer arrival process and of vehicle travel times; the authors shows that, if the AMoD system enjoys high vehicle availability, optimizing the expectation of the BCMP model is *equivalent* to optimizing the deterministic network flow model proposed in this chapter.

2.3.2 Continuum approximation

Network flow models approximate customers and vehicles as *continuous* commodities, akin to a fluidic model. This property can make such model impractical for real-time control of AMoD systems, where discrete control actions must be issued to the vehicles. While integral versions of the multi-commodity flow problem exist, such problems are generally NP-hard (as discussed in Section 4.1.2). Three approaches are available to overcome this difficulty:

- The single-commodity minimum-cost network flow problem is known to have *totally unimodular* structure: that is, if the capacity of all road links and the intensities of all origins and destinations are integral, then there exists an integral optimal solution to the problem. Such an integral optimal solution can then be decomposed in integral vehicle routes through the flow decomposition algorithm. We exploit this property to design an efficient controller for congestion-aware routing of rebalancing vehicles in Chapter 3.

- Randomized routing algorithms [Srinivasan, 1999] decompose and sample the solution of a network flow problem to produce integral routes for the vehicles that recover the cost of the continuous solution in expectation and offer probabilistic bounds on the probability of violating the congestion constraints. The use of randomized routing algorithms is typically restricted to multi-commodity flows where each commodity has a single origin and destination; in Chapter 4, we leverage the structural properties of the AMoD network flow model to employ these algorithms in presence of a multi-origin, multi-destination rebalancing flow.

- *Receding-horizon sampling* can be used to obtain discrete control actions for the vehicles by sampling the first time step of the optimal solution to a time-varying network flow problem. While this approach is not guaranteed to recover the performance of the optimum solution in expectation, it often exhibits good performance in practical applications. In Chapter 5, we leverage a receding-horizon sampling approach to achieve real-time control of AMoD systems in coordination with the power network.

Chapter 3

Routing autonomous vehicles in congested transportation networks: structural properties and coordination algorithms

This chapter considers the problem of routing and rebalancing a shared fleet of autonomous (i.e., self-driving) vehicles providing on-demand mobility within a *capacitated* transportation network, where congestion might disrupt throughput. We model the problem within a network flow framework and show that under relatively mild assumptions the rebalancing vehicles, if properly coordinated, do not lead to an increase in congestion (in stark contrast to common belief). From an algorithmic standpoint, such theoretical insight suggests that the problem of routing customers and rebalancing vehicles can be *decoupled*, which leads to a computationally-efficient routing and rebalancing algorithm for the autonomous vehicles. Numerical experiments and case studies corroborate our theoretical insights and show that the proposed algorithm outperforms state-of-the-art point-to-point methods by avoiding excess congestion on the road. Collectively, this work provides a rigorous approach to the problem of congestion-aware, system-wide coordination of autonomously driving vehicles, and to the characterization of the sustainability of such robotic systems.

3.1 Introduction

Autonomous (i.e., robotic, self-driving) vehicles are rapidly becoming a reality and hold great promise for increasing safety and enabling access to mobility for those unable or unwilling to drive

[Mitchell et al., 2010, Urmson, 2014]. A particularly attractive operational paradigm involves coordinating a fleet of autonomous vehicles to provide on-demand service to customers, also called autonomous mobility-on-demand (AMoD). An AMoD system may reduce the cost of travel [Spieser et al., 2014] as well as provide additional sustainability benefits such as increased overall vehicle utilization, reduced demand for urban parking infrastructure, and reduced pollution (with electric vehicles) [Mitchell et al., 2010]. The key benefits of AMoD are realized through vehicle sharing, where each vehicle, after servicing a customer, drives itself to the location of the next customer or *rebalances* itself throughout the city in anticipation of future customer demand [Pavone et al., 2012].

In terms of traffic congestion, however, there has been no consensus on whether autonomous vehicles in general, and AMoD systems in particular, will ultimately be beneficial or detrimental. It has been argued that by having faster reaction times, autonomous vehicles may be able to drive faster and follow other vehicles at closer distances without compromising safety, thereby effectively increasing the capacity of a road and reducing congestion [Maciejewski and Bischoff, 2017]. They may also be able to interact with traffic lights to reduce full stops at intersections [Pérez et al., 2010]. On the downside, the process of vehicle rebalancing (empty vehicle trips) increases the total number of vehicles on the road (assuming the number of vehicles with customers stays the same). Indeed, it has been argued that the presence of many rebalancing vehicles may contribute to an *increase* in congestion [Templeton, 2010, Barnard, 2016]. These statements, however, do not take into account that in an AMoD system the operator has control over the actions (destination and routes) of the vehicles, and may route vehicles intelligently to avoid increasing congestion or perhaps even decrease it.

Accordingly, the goal of this chapter is twofold. First, on an engineering level, we aim to devise routing and rebalancing algorithms for an autonomous vehicle fleet that seek to minimize congestion. Second, on a socio-economic level, we aim to rigorously address the concern that autonomous cars may lead to increased congestion and thus disrupt currently congested transportation infrastructures.

Literature review: In this section of the thesis, we investigate the problem of controlling an AMoD system within a road network in the presence of congestion effects. Previous work on control of AMoD systems have primarily concentrated on the rebalancing problem [Pavone et al., 2012, Spieser et al., 2014], whereby one strives to allocate empty vehicles throughout a city while minimizing fuel costs or customer wait times. The rebalancing problem has been studied in [Pavone et al., 2012] using a fluidic model and in [Zhang and Pavone, 2016] using a queueing network model. An alternative formulation is the one-to-one pickup and delivery problem [Berbeglia et al., 2010], where a fleet of vehicles service pickup and delivery requests within a given region. Combinatorial asymptotically optimal algorithms for pickup and delivery problems were presented in [Treleaven et al., 2011, Treleaven et al., 2013], and generalized to road networks in [Treleaven et al., 2012]. Almost all current approaches assume point-to-point travel between origins and destinations (no road network), and even routing problems on road networks (e.g.

[Treleaven et al., 2012]) do not take into account vehicle-to-vehicle interactions that would cause congestion and reduce system throughput.

On the other hand, traffic congestion has been studied in economics and transportation for nearly a century. The first congestion models [Wardrop, 1952, Lighthill and Whitham, 1955, Daganzo, 1994] sought to formalize the relationship between vehicle speed, density, and flow. Since then, approaches to modeling congestion have included empirical [Kerner, 2009], simulation-based [Treiber et al., 2000, Yang and Koutsopoulos, 1996, Balmer et al., 2009, Fagnant and Kockelman, 2014, Levin et al., 2017], queueing-theoretical [Osorio and Bierlaire, 2009], and optimization [Peeta and Mahmassani, 1995, Janson, 1991]. While there have been many high fidelity congestion models that can accurately predict traffic patterns, the primary goal of congestion modeling has been the *analysis* of traffic behavior. Efforts to *control* traffic have been limited to the control of intersections [Le et al., 2015, Xiao et al., 2015] and freeway on-ramps [Papageorgiou et al., 1991] because human drivers behave non-cooperatively. The problem of cooperative, system-wide routing (a key benefit of AMoD systems) is similar to the dynamic traffic assignment problem (DTA) [Janson, 1991] and to [Wilkie et al., 2011, Wilkie et al., 2014] in the case of online routing. The key difference is that these approaches only optimize routes for passenger vehicles while we seek to optimize the routes of *both* passenger vehicles *and* empty rebalancing vehicles.

Statement of contributions: The contribution of this chapter is threefold. First, we model an AMoD system within a network flow framework, whereby customer-carrying and empty rebalancing vehicles are represented as flows over a *capacitated* road network (in such model, when the flow of vehicles along a road reaches a critical capacity value, congestion effects occur). Within this model, we provide a cut condition for the road graph that needs to be satisfied for congestion-free customer and rebalancing flows to exist. Most importantly, under the assumption of a *symmetric* road network, we investigate an existential result that leads to two key conclusions: (1) rebalancing does not increase congestion, and (2) for certain cost functions, the problems of finding customer and rebalancing flows can be decoupled. Second, leveraging the theoretical insights, we propose a computationally-efficient algorithm for congestion-aware routing and rebalancing of an AMoD system that is broadly applicable to time-varying, possibly asymmetric road networks. Third, through numerical studies on real-world traffic data, we validate our assumptions and show that the proposed real-time routing and rebalancing algorithm outperforms state-of-the-art point-to-point rebalancing algorithms in terms of lower customer wait times by avoiding excess congestion on the road.

Organization: The remainder of this chapter is organized as follows: in Section 3.2 we present a network flow model of an AMoD system on a capacitated road network and formulate the routing and rebalancing problem. In Section 3.3 we present key structural properties of the model including fundamental limitations of performance and conditions for the existence of feasible (in particular, congestion-free) solutions. The insights from Section 3.3 are used to develop a practical real-time routing and rebalancing algorithm in Section 3.4. Numerical studies and simulation results are

presented in Section 3.5, and in Section 3.6 we draw conclusions and discuss directions for future work.

3.2 Model Description and Problem Formulation

In this section we formulate a network flow model for an AMoD system operating over a capacitated road network. The model allows us to derive key structural insights into the vehicle routing and rebalancing problem, and motivates the design of real-time, congestion-aware algorithms for coordinating the robotic vehicles. We start in Section 3.2.1 with a discussion of our congestion model; then, in Section 3.2.2 we provide a detailed description of the overall AMoD system model.

3.2.1 Congestion Model

We use a simplified congestion model consistent with classical traffic flow theory [Wardrop, 1952]. In classical traffic flow theory, at low vehicle densities on a road link, vehicles travel at the free flow speed of the road (imposed by the speed limit). This is referred to as the free flow phase of traffic. In this phase, the free flow speed is approximately constant [Kerner, 2009]. The flow, or flow rate, is the number of vehicles passing through the link per unit time, and is given by the product of the speed and density of vehicles. When the flow of vehicles reaches an empirically observed critical value, the flow reaches its maximum. Beyond the critical flow rate, vehicle speeds are dramatically reduced and the flow decreases, signaling the beginning of traffic congestion. The maximum stationary flow rate is called the *capacity* of the road link in the literature. In our approach, road capacities are modeled as *constraints on the flow of vehicles*. In this way, the model captures the behavior of vehicles up to the onset of congestion.

This simplified congestion model is adequate for our purposes because the goal is not to analyze the behavior of vehicles in congested networks, but to control vehicles in order to avoid the onset of congestion. We also do not explicitly model delays at intersections, spillback behavior due to congestion, or bottleneck behavior due to the reduction of the number of lanes on a road link. An extension to our model that accommodates (limited) congestion on links is presented in Section 3.5.2.

3.2.2 Network Flow Model of AMoD system

We consider a road network modeled as a directed graph $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the node set and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the edge set. Figure 3.1 shows one such network. The nodes v in $\mathcal{V}$ represent intersections and locations for trip origins/destinations, and the edges (u, v) in $\mathcal{E}$ represent road links. As discussed in Section 3.2.1, congestion is modeled by imposing capacity constraints on the road links: each constraint represents the capacity of the road upon the onset of congestion.

Specifically, for each road link $(u, v) \in \mathcal{E}$, we denote by $c(u, v) : \mathcal{E} \mapsto \mathbb{N}_{>0}$ the capacity of that link. When the flow rate on a road link is less than the capacity of the link, all vehicles are assumed to travel at the free flow speed, or the speed limit of the link. For each road link $(u, v) \in \mathcal{E}$, we denote by $t(u, v) : \mathcal{E} \mapsto \mathbb{R}_{\geq 0}$ the corresponding free flow time required to traverse road link (u, v). Conversely, when the flow rate on a road link is larger than the capacity of the link, the traversal time is assumed equal to ∞ (we reiterate that our focus in this section is on avoiding the onset of congestion).

We assume that the road network is *capacity-symmetric* (or symmetric for short): for any cut[1] $(\mathcal{S}, \bar{\mathcal{S}})$ of $G(\mathcal{V}, \mathcal{E})$, the overall capacity of the edges connecting nodes in $\mathcal{S}$ to nodes in $\bar{\mathcal{S}}$ equals the overall capacity of the edges connecting nodes in $\bar{\mathcal{S}}$ to nodes in $\mathcal{S}$, that is

$$\sum_{(u,v)\in\mathcal{E}:\ u\in\mathcal{S},\, v\in\bar{\mathcal{S}}} c(u, v) = \sum_{(v,u)\in\mathcal{E}:\ u\in\mathcal{S},\, v\in\bar{\mathcal{S}}} c(v, u)$$

It is easy to verify that a network is capacity-symmetric if and only if the overall capacity entering each *node* equals the capacity exiting each node., i.e.

$$\sum_{u\in\mathcal{V}:(u,v)\in\mathcal{E}} c(u, v) = \sum_{w\in\mathcal{V}:(v,w)\in\mathcal{E}} c(v, w) \quad \forall v \in \mathcal{V}$$

If all *edges* have symmetrical capacity, i.e., for all $(u, v) \in \mathcal{E}$, $c(u, v) = c(v, u)$, then the network is capacity-symmetric. The converse statement, however, is not true in general.

Transportation requests are described by the tuple (s, t, λ), where $s \in \mathcal{V}$ is the origin of the requests, $t \in \mathcal{V}$ is the destination, and $\lambda \in \mathbb{R}_{>0}$ is the rate of requests, in customers per unit time. Transportation requests are assumed to be stationary and deterministic, i.e., the rate of requests does not change with time and is a deterministic quantity. The set of transportation requests is denoted by $\mathcal{M} = \{(s_m, t_m, \lambda_m)\}_m$, and its cardinality is denoted by M.

Single-occupancy vehicles travel within the network while servicing the transportation requests. We denote $f_m(u, v) : \mathcal{E} \mapsto \mathbb{R}_{\geq 0}$, $m = \{1, \ldots, M\}$, as the *customer flow* for requests m on edge (u, v), i.e., the amount of flow from origin s_m to destination t_m that uses link (u, v). We also denote $f_R(u, v) : \mathcal{E} \mapsto \mathbb{R}_{\geq 0}$ as the *rebalancing flow* on edge (u, v), i.e., the amount of rebalancing flow traversing edge (u, v) needed to realign the vehicles with the asymmetric distribution of transportation requests.

3.2.3 The Routing Problem

The goal is to compute flows for the autonomous vehicles that (i) transfer customers to their desired destinations in minimum time (customer-carrying trips) and (ii) rebalance vehicles throughout the

[1] For any subset of nodes $\mathcal{S} \subseteq \mathcal{V}$, we define a *cut* $(\mathcal{S}, \bar{\mathcal{S}}) \subseteq \mathcal{E}$ as the set of edges whose origin lies in $\mathcal{S}$ and whose destination lies in $\bar{\mathcal{S}} = \{\mathcal{V} \setminus \mathcal{S}\}$. Formally, $(\mathcal{S}, \bar{\mathcal{S}}) := \{(u, v) \in \mathcal{E} : u \in \mathcal{S}, v \in \bar{\mathcal{S}}\}$.

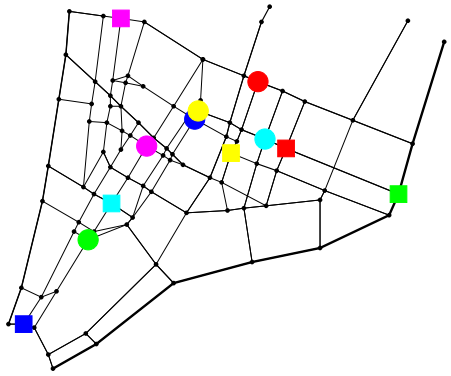

Figure 3.1: A road network modeling Lower Manhattan and the Financial District. Nodes (denoted by small black dots) model intersections; select nodes, denoted by colored circular and square markers, model passenger trips' origins and destinations. Different trip requests are denoted by different colors. Roads are modeled as edges; line thickness is proportional to road capacity.

network to realign the vehicle fleet with transportation demand (customer-empty trips). Specifically, the *Congestion-free Routing and Rebalancing Problem (CRRP)* is formally defined as follows. Given a capacitated, symmetric network $G(\mathcal{V}, \mathcal{E})$, a set of transportation requests $\mathcal{M} = \{(s_m, t_m, \lambda_m)\}_m$, and a weight factor $\rho > 0$, solve

$$\underset{f_m(\cdot,\cdot), f_R(\cdot,\cdot)}{\text{minimize}} \quad \sum_{m\in\mathcal{M}} \sum_{(u,v)\in\mathcal{E}} t(u,v) f_m(u,v) + \rho \sum_{(u,v)\in\mathcal{E}} t(u,v) f_R(u,v) \tag{3.1a}$$

$$\text{subject to} \quad \sum_{u\in\mathcal{V}} f_m(u, s_m) + \lambda_m = \sum_{w\in\mathcal{V}} f_m(s_m, w) \;\; \forall m \in \mathcal{M} \tag{3.1b}$$

$$\sum_{u\in\mathcal{V}} f_m(u, t_m) = \lambda_m + \sum_{w\in\mathcal{V}} f_m(t_m, w) \;\; \forall m \in \mathcal{M} \tag{3.1c}$$

$$\sum_{u\in\mathcal{V}} f_m(u, v) = \sum_{w\in\mathcal{V}} f_m(v, w) \qquad \forall m \in \mathcal{M}, v \in \mathcal{V} \setminus \{s_m, t_m\} \tag{3.1d}$$

$$\sum_{u\in\mathcal{V}} f_R(u, v) + \sum_{m\in\mathcal{M}} 1_{v=t_m} \lambda_m = \sum_{w\in\mathcal{V}} f_R(v, w) + \sum_{m\in\mathcal{M}} 1_{v=s_m} \lambda_m \quad \forall v \in \mathcal{V} \tag{3.1e}$$

$$f_R(u, v) + \sum_{m\in\mathcal{M}} f_m(u, v) \le c(u, v) \;\; \forall (u, v) \in \mathcal{E} \tag{3.1f}$$

The cost function (3.1a) is a weighted sum (with weight ρ) of the overall duration of all passenger trips and the duration of rebalancing trips. The weight ρ denotes the ratio between the cost per unit time incurred by the customer-carrying vehicles (i.e., the sum of the customers' value of time and of the vehicles' operating cost) and the cost per unit time incurred by the rebalancing vehicles (which only captures the vehicles' operating cost). Constraints (3.1b), (3.1c) and (3.1d) enforce continuity of each trip (i.e., flow conservation) across nodes. Constraint (3.1e) ensures that vehicles are rebalanced

throughout the road network to re-align vehicle distribution with transportation requests, i.e. to ensure that every outbound customer flow is matched by an inbound flow of rebalancing vehicles and vice versa. Finally, constraint (3.1f) enforces the capacity constraint on each link (function 1_x denotes the indicator function of the Boolean variable $x = \{\text{true, false}\}$, that is 1_x equals one if x is true, and equals zero if x is false). Note that the CRRP is a linear program and, in particular, a special instance of the fractional multi-commodity flow problem [Ahuja et al., 1993].

We denote a customer flow $\{f_m(u,v)\}_{(u,v),m}$ that satisfies Equations (3.1b), (3.1c), (3.1d) and (3.1f) as a *feasible customer flow*. For a given set of feasible customer flows $\{f_m(u,v)\}_{(u,v),m}$, we denote a flow $\{f_R(u,v)\}_{(u,v)}$ that satisfies Equation (3.1e) and such that the combined flows $\{f_m(u,v), f_R(u,v)\}_{(u,v),m}$ satisfy Equation (3.1f) as a *feasible rebalancing flow*. We remark that a rebalancing flow that is feasible with respect to a set of customer flows may be infeasible for a different collection of customer flows.

For a given set of optimal flows $\{f^*_m(u,v)\}_{(u,v),m}$ and $\{f^*_R(u,v)\}_{(u,v)}$, the minimum number of vehicles needed to implement them is given by

$$V_{\min} = \left\lceil \sum_{(u,v)\in\mathcal{E}} t(u,v) \left(f^*_R(u,v) + \sum_{m\in\mathcal{M}} f^*_m(u,v) \right) \right\rceil .$$

This follows from a similar analysis done in [Pavone et al., 2012] for point-to-point networks. Hence, the cost function (3.1a) is aligned with the desire of minimizing the number of vehicles needed to operate an AMoD system.

3.2.4 Discussion

A few comments are in order. First, we assume that transportation requests are time invariant. This assumption is valid when transportation requests change slowly with respect to the average duration of a customer's trip, which is often the case in dense urban environments [Neuburger, 1971]. Additionally, in Section 3.4 we will present algorithmic tools that allow one to extend the insights gained from the time-invariant case to the time-varying counterpart. Second, the assumption of single-occupancy for the vehicles models most of the existing (human) one-way vehicle sharing systems (where the driver is considered "part" of the vehicle), and chiefly disallows the provision of ride-sharing or carpooling service (this is an aspect left for future research). Third, as also discussed in Section 3.2.1, our congestion model is simpler and less accurate than typical congestion models used in the transportation community. However, our model lends itself to efficient real-time optimization and thus it is well-suited to the *control* of fleets of autonomous vehicles. Existing high-fidelity congestion models should be regarded as complementary and could be used offline to identify the congestion thresholds used in our model. Fourth, while we have defined the CRRP in terms of fractional flows, an integer-valued counterpart can be defined and (approximately) solved to find optimal routes for each *individual* customer and vehicle. Algorithmic aspects will be investigated in

depth in Section 3.4, with the goal of devising practical, real-time routing and rebalancing algorithms. Fifth, trip requests are assumed to be known. In practice, trip requests can be reserved in advance, estimated from historical data, or estimated in real time. Finally, the assumption of capacity-symmetric road networks indeed appears reasonable for a number of major U.S. metropolitan areas (note that this assumption is much less restrictive than assuming every *individual* road is capacity-symmetric). In Section 3.5.1, by using OpenStreetMap data [Haklay and Weber, 2008], we provide a rigorous characterization in terms of capacity symmetry of the road networks of New York City, Chicago, Los Angeles and other major U.S. cities. The results consistently show that urban road networks are usually symmetric to a *very high* degree. Additionally, several of our theoretical and algorithmic results extend to the case where this assumption is lifted, as it will be highlighted throughout the chapter.

3.3 Structural Properties of the Network Flow Model

In this section we provide two key structural results for the network flow model presented in Section 3.2.2. First, we provide a cut condition that needs to be satisfied for feasible customer and rebalancing flows to exist. In other words, this condition provides a fundamental limitation of performance for congestion-free AMoD service in a given road network. Second, we investigate an existential result (our main theoretical result) that is germane to two key conclusions: (1) rebalancing does not increase congestion in symmetric road networks, and (2) for certain cost functions, the problems of finding customer and rebalancing flows can be *decoupled* – an insight that will be heavily exploited in subsequent sections.

3.3.1 Fundamental Limitations

We start with a few definitions. For a given set of feasible customer flows $\{f_m(u,v)\}_{(u,v),m}$, we denote by $F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}})$ the overall flow exiting a cut $(\mathcal{S},\bar{\mathcal{S}})$, i.e.,

$$F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) := \sum_{m\in\mathcal{M}} \sum_{u\in\mathcal{S},v\in\bar{\mathcal{S}}} f_m(u,v).$$

Similarly, we denote by $C_{\text{out}}(\mathcal{S},\bar{\mathcal{S}})$ the capacity of the network exiting $\mathcal{S}$, i.e., $C_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) = \sum_{u\in S,v\in\bar{\mathcal{S}}} c(u,v)$. Analogously, $F_{\text{in}}(\mathcal{S},\bar{\mathcal{S}})$ denotes the overall flow entering $\mathcal{S}$ from $\bar{\mathcal{S}}$, i.e., $F_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) := F_{\text{out}}(\bar{\mathcal{S}},\mathcal{S})$, and $C_{\text{in}}(\mathcal{S},\bar{\mathcal{S}})$ denotes the capacity entering $\mathcal{S}$ from $\bar{\mathcal{S}}$, i.e., $C_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) := C_{\text{out}}(\bar{\mathcal{S}},\mathcal{S})$. We highlight that the arguments leading to the main result of this subsection (Theorem 3.3.4) do not require the assumption of capacity symmetry; hence, Theorem 3.3.4 holds for *asymmetric* road networks as well.

The next technical lemma shows that the net flow leaving set $\mathcal{S}$ equals the difference between the flow originating from the origins s_m in $\mathcal{S}$ and the flow exiting through the destinations t_m in $\mathcal{S}$,

that is,

Lemma 3.3.1 (Net flow across a cut)**.** *Consider a set of feasible customer flows* $\{f_m(u,v)\}_{(u,v),m}$*. Then, for every cut* $(\mathcal{S},\bar{\mathcal{S}})$*, the net flow leaving set* $\mathcal{S}$ *satisfies*

$$F_{out}(\mathcal{S},\bar{\mathcal{S}}) - F_{in}(\mathcal{S},\bar{\mathcal{S}}) = \sum_{m\in\mathcal{M}} 1_{s_m\in\mathcal{S}}\lambda_m - \sum_{m\in\mathcal{M}} 1_{t_m\in\mathcal{S}}\lambda_m.$$

Proof. We compute the sum over all customer flows $m \in \mathcal{M}$ and over all nodes $v \in \mathcal{V}$ of the node balance equation for flow m at node v (Equation (3.1c) if node v is the source of m, Equation (3.1d) if node v is the sink of m, or Equation (3.1b) otherwise). We obtain

$$\sum_{v\in\mathcal{S}}\sum_{m\in\mathcal{M}}\left(\sum_{u\in\mathcal{V}} f_m(u,v) + 1_{v=s_m}\lambda_m\right) = \sum_{v\in\mathcal{S}}\sum_{m\in\mathcal{M}}\left(\sum_{w\in\mathcal{V}} f_m(v,w) + 1_{v=t_m}\lambda_m\right).$$

For any edge (u,v) such that $u,v \in \mathcal{S}$, the customer flow $f_m(u,v)$ appears on both sides of the equation. Thus the equation above simplifies to

$$\sum_{m\in\mathcal{M}}\sum_{v\in\mathcal{S}}\left(\sum_{u\in\bar{\mathcal{S}}} f_m(u,v) + 1_{v=s_m}\lambda_m\right) = \sum_{m\in\mathcal{M}}\sum_{v\in\mathcal{S}}\left(\sum_{w\in\bar{\mathcal{S}}} f_m(v,w) + 1_{v=t_m}\lambda_m\right),$$

which leads to the claim of the lemma

$$F_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) + \sum_{m\in\mathcal{M}} 1_{s_m\in\mathcal{S}}\lambda_m = F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) + \sum_{m\in\mathcal{M}} 1_{t_m\in\mathcal{S}}\lambda_m.$$

□

We now state two additional lemmas providing, respectively, lower and upper bounds for the outflows $F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}})$.

Lemma 3.3.2 (Lower bound for outflow)**.** *Consider a set of feasible customer flows* $\{f_m(u,v)\}_{(u,v),m}$*. Then, for any cut* $(\mathcal{S},\bar{\mathcal{S}})$*, the overall flow* $F_{out}(\mathcal{S},\bar{\mathcal{S}})$ *exiting cut* $(\mathcal{S},\bar{\mathcal{S}})$ *is lower bounded according to*

$$\sum_{m\in\mathcal{M}} 1_{s_m\in\mathcal{S},t_m\in\bar{\mathcal{S}}}\lambda_m \le F_{out}(\mathcal{S},\bar{\mathcal{S}}).$$

Proof. Adding Equations (3.1b), (3.1c) and (3.1d) over all nodes in $\mathcal{S}$ and over all flows whose origin is in $\mathcal{S}$ and whose destination is in $\bar{\mathcal{S}}$, one obtains

$$\sum_{m:s_m\in\mathcal{S},t_m\in\bar{\mathcal{S}}}\sum_{v\in\mathcal{S}}\left(\sum_{u\in\mathcal{V}} f_m(u,v) + 1_{v=s_m}\lambda_m\right) = \sum_{m:s_m\in\mathcal{S},t_m\in\bar{\mathcal{S}}}\sum_{v\in\mathcal{S}}\left(\sum_{w\in\mathcal{V}} f_m(v,w)\right).$$

Flows $f_m(u,v)$ such that both u and v are in $\mathcal{S}$ appear on both sides of the equation. Simplifying, one obtains

$$\sum_{m\in\mathcal{M}} 1_{s_m\in\mathcal{S},t_m\in\bar{\mathcal{S}}}\lambda_m = \sum_{m:s_m\in\mathcal{S},t_m\in\bar{\mathcal{S}}}\left(\sum_{v\in\mathcal{S},w\in\bar{\mathcal{S}}} f_m(v,w) - \sum_{v\in\mathcal{S},u\in\bar{\mathcal{S}}} f_m(u,v)\right)$$

The first term on the right-hand side represents a lower bound for $F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}})$, since

$$F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) = \sum_{m\in\mathcal{M}}\sum_{v\in\mathcal{S},w\in\bar{\mathcal{S}}} f_m(v,w) \geq \sum_{m:s_m\in\mathcal{S},t_m\in\bar{\mathcal{S}}}\sum_{v\in\mathcal{S},w\in\bar{\mathcal{S}}} f_m(v,w).$$

Furthermore, the second term on the right-hand side is upper-bounded by zero. The lemma follows. □

Lemma 3.3.3 (Upper bound for outflow)**.** *Assume there exists a set of* feasible *customer and rebalancing flows* $\{f_m(u,v),\, f_R(u,v)\}_{(u,v),m}$*. Then, for every cut* $(\mathcal{S},\bar{\mathcal{S}})$*,*

1. $F_{out}(\mathcal{S},\bar{\mathcal{S}}) \leq C_{out}(\mathcal{S},\bar{\mathcal{S}})$*, and*
2. $F_{out}(\mathcal{S},\bar{\mathcal{S}}) \leq C_{in}(\mathcal{S},\bar{\mathcal{S}})$.

Proof. The first condition follows trivially from equation (3.1f). As for the second condition, consider a cut $(\mathcal{S},\bar{\mathcal{S}})$. Analogously as for the definitions of $F_{\text{in}}(\mathcal{S},\bar{\mathcal{S}})$ and $F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}})$, let $F^{\text{reb}}_{\text{in}}(\mathcal{S},\bar{\mathcal{S}})$ and $F^{\text{reb}}_{\text{out}}(\mathcal{S},\bar{\mathcal{S}})$ denote, respectively, the overall rebalancing flow entering (exiting) cut $(\mathcal{S},\bar{\mathcal{S}})$. Summing equation (3.1e) over all nodes in S, one easily obtains

$$F^{\text{reb}}_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) - F^{\text{reb}}_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) = \sum_{m\in\mathcal{M}} 1_{s_m\in\mathcal{S}}\lambda_m - \sum_{m\in\mathcal{M}} 1_{t_m\in\mathcal{S}}\lambda_m.$$

Combining the above equation with Lemma 3.3.1, one obtains

$$F^{\text{reb}}_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) - F^{\text{reb}}_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) = F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) - F_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}),$$

in other words, rebalancing flows should make up the difference between the customer inflows and outflows across cut $(\mathcal{S},\bar{\mathcal{S}})$. Accordingly, the total inflow of vehicles across $(\mathcal{S},\bar{\mathcal{S}})$, $F^{\text{tot}}_{\text{in}}(\mathcal{S},\bar{\mathcal{S}})$, satisfies the inequality

$$\begin{aligned} F^{\text{tot}}_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) :=& F_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) + F^{\text{reb}}_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) \\ =& F_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) + F^{\text{reb}}_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) + F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) - F_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) \\ \geq& F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}). \end{aligned}$$

Since the customer and rebalancing flows $\{f_m(u,v),\, f_R(u,v)\}_{(u,v),m}$ are feasible, then, by equation

(3.1f), $F_{\text{in}}^{\text{tot}}(S, \bar{S}) \leq C_{\text{in}}(S, \bar{S})$. Collecting the results, one obtains the second condition. □

We are now in a position to present a *structural* (i.e., flow-independent) necessary condition for the existence of feasible customer and rebalancing flows.

Theorem 3.3.4 (Necessary condition for feasible flows)**.** *A necessary condition for the existence of a set of* feasible *customer and rebalancing flows* $\{f_m(u,v),\ f_R(u,v)\}_{(u,v),m}$*, is that, for every cut* $(\mathcal{S}, \bar{\mathcal{S}})$*,*

1. $\sum_{m\in\mathcal{M}} 1_{s_m\in\mathcal{S}, t_m\in\bar{\mathcal{S}}}\lambda_m \leq C_{out}(\mathcal{S}, \bar{\mathcal{S}})$*, and*
2. $\sum_{m\in\mathcal{M}} 1_{s_m\in\mathcal{S}, t_m\in\bar{\mathcal{S}}}\lambda_m \leq C_{in}(\mathcal{S}, \bar{\mathcal{S}})$.

Proof. The theorem is a trivial consequence of Lemmas 3.3.2 and 3.3.3. □

Theorem 3.3.4 essentially provides a structural fundamental limitation of performance for a given road network: if the cut conditions in Theorem 3.3.4 are not met, then there is no hope of finding congestion-free customer and rebalancing flows. We reiterate that Theorem 3.3.4 holds for both symmetric and asymmetric networks (for a symmetric network, claim 2) in Lemma 3.3.3 and condition 2) in Theorem 3.3.4 are redundant).

3.3.2 Existence of Congestion-Free Flows

In this section we address the following question: assuming there exists a feasible customer flow, is it always possible to find a feasible rebalancing flow? As we will see, the answer to this question is affirmative and has both conceptual and algorithmic implications.

Theorem 3.3.5 (Feasible rebalancing)**.** *Assume there exists a set of feasible customer flows* $\{f_m(u,v)\}_{(u,v),m}$*. Then, it is* always *possible to find a set of feasible rebalancing flows* $\{f_R(u,v)\}_{(u,v)}$.

Proof. We prove the theorem for the special case where no node $v \in \mathcal{V}$ is associated with both an origin and a destination for the transportation requests in $\mathcal{M}$. This is without loss of generality, as the general case where a node v has both an origin and a destination assigned can be reduced to this special case, by associating with node v a "shadow" node so that (i) all destinations are assigned to the shadow node and (ii) node v and its shadow node are mutually connected via an infinite-capacity, zero-travel-time edge.

We start the proof by defining the concepts of *partial rebalancing flows* and *defective origins and destinations*. Specifically, a partial rebalancing flow, denoted as $\{\hat{f}_R(u,v)\}_{(u,v)}$, is a set of mappings from $\mathcal{E}$ to $\mathbb{R}_{\geq 0}$ obeying the following properties:

1. It satisfies constraint (3.1e) at every node that is not an origin nor a destination, that is $\forall\, v \in \{\mathcal{V} \setminus \{\{s_m\}_m \cup \{t_m\}_m\}\}$,

$$\sum_{u \in \mathcal{V}} \hat{f}_R(u,v) = \sum_{w \in \mathcal{V}} \hat{f}_R(v,w).$$

2. It may not satisfy constraint (3.1e) in the "$\leq$ direction" at every node that is an origin, that is $\forall\, v \in \mathcal{V}$ such that $\exists m \in \mathcal{M} : v = s_m$,

$$\sum_{u \in \mathcal{V}} \hat{f}_R(u,v) \leq \sum_{w \in \mathcal{V}} \hat{f}_R(v,w) + \sum_{m \in \mathcal{M}} 1_{v=s_m} \lambda_m.$$

3. It may not satisfy constraint (3.1e) in the "$\geq$ direction" at every node that is a destination, that is $\forall\, v \in \mathcal{V}$ such that $\exists m \in \mathcal{M} : v = t_m$,

$$\sum_{u \in \mathcal{V}} \hat{f}_R(u,v) + \sum_{m \in \mathcal{M}} 1_{v=t_m} \lambda_m \geq \sum_{w \in \mathcal{V}} \hat{f}_R(v,w).$$

4. The combined customer and partial rebalancing flows $\{f_m(u,v), \hat{f}_R(u,v)\}_{(u,v),m}$ satisfy Equation (3.1f) for every edge $(u,v) \in \mathcal{E}$.

Note that the trivial zero flow, that is $\hat{f}_R(u,v) = 0$ for all $(u,v) \in \mathcal{E}$, is a partial rebalancing flow (in other words, the set of partial rebalancing flows in not empty). Clearly a feasible rebalancing flow is also a partial rebalancing flow, but the opposite is not necessarily true.

For a given partial rebalancing flow, we denote an origin node, that is a node $v \in \mathcal{V}$ such that $v = s_m$ for some $m = 1, \ldots, M$, as a *defective* origin if Equation (3.1e) is not satisfied at $v = s_m$ (in other words, the strict inequality $<$ holds). Analogously, we denote a destination node, that is a node $v \in \mathcal{V}$ such that $v = t_m$ for some $m = 1, \ldots, M$, as a *defective* destination if Equation (3.1e) is not satisfied at $v = t_m$ (in other words, the strict inequality $>$ holds). The next lemma links the concepts of partial rebalancing flows and defective origins/destinations.

Lemma 3.3.6 (Co-existence of defective origins/destinations)**.** *For every partial rebalancing flow that is not a feasible rebalancing flow, there exists at least one node $u \in \mathcal{V}$ that is a defective origin,* and *one node $v \in \mathcal{V}$ that is a defective destination.*

Proof. By contradiction. Since the flow $\{\hat{f}_R(u,v)\}_{(u,v)}$ is not a feasible rebalancing flow, there exists at least one defective origin or a defective destination. Assume that there exists at least one defective destination, say a node $\hat{t}_j$ where Equation (3.1e) is violated:

$$\sum_{u \in \mathcal{V}} \hat{f}_R(u,\hat{t}_j) + \sum_{m \in \mathcal{M}} 1_{\hat{t}_j = t_m} \lambda_m > \sum_{w \in \mathcal{V}} \hat{f}_R(\hat{t}_j, w),$$

Now, assume that there does not exist any defective origin. By summing Equation (3.1e) over all nodes $v \in \mathcal{V}$ and simplifying all flows $\hat{f}_R(u,v)$ (as they appear on both sides of the resulting equation), one obtains

$$\sum_{v\in\mathcal{V}}\sum_{m\in\mathcal{M}} 1_{v=t_m}\lambda_m > \sum_{v\in\mathcal{V}}\sum_{m\in\mathcal{M}} 1_{v=s_m}\lambda_m,$$

that is $\sum_{m\in\mathcal{M}}\lambda_m > \sum_{m\in\mathcal{M}}\lambda_m$, which is a contradiction. Noticing that the symmetric case where we assume that there exists at least one defective destination leads to an analogous contradiction, the lemma follows. □

For a given set of customer flows $\{f_m(u,v)\}_{(u,v),m}$ and partial rebalancing flows $\{\hat{f}_R(u,v)\}_{(u,v)}$, we call an edge $(u,v) \in \mathcal{E}$ *saturated* if equation (3.1f) holds with equality for that edge. We call a path *saturated* if at least one of the edges along the path is saturated. We now prove the existence of a special partial rebalancing flow where defective destinations and defective origins are separated by a graph cut formed exclusively by saturated edges (this result, and its consequences, are illustrated in Figure 3.2).

Lemma 3.3.7 (Existence of partial rebalancing flows)**.** *Assume there exists a set of* feasible *customer flows $\{f_m(u,v)\}_{(u,v),m}$, but there does not exist a set of feasible rebalancing flows $\{f_R(u,v)\}_{(u,v)}$. Then, there exists a partial rebalancing flow $\{\hat{f}_R(u,v)\}_{(u,v)}$ that induces a graph cut $(\mathcal{S},\bar{\mathcal{S}})$ with the following properties: (i) all defective destinations are in $\mathcal{S}$, (ii) all defective origins are in $\bar{\mathcal{S}}$, and (iii) all edges in $(S,\bar{\mathcal{S}})$ are saturated.*

Proof. The proof is constructive and constructs the desired partial rebalancing flow by starting with the trivial zero flow $\hat{f}_R(u,v) = 0$ for all $(u,v) \in \mathcal{E}$. Let $\mathcal{V}_{\text{or, def}} := \{\hat{s}_1, \ldots, \hat{s}_{|\mathcal{V}_{\text{or, def}}|}\}$ and $\mathcal{V}_{\text{dest, def}} := \{\hat{t}_1, \ldots, \hat{t}_{|\mathcal{V}_{\text{dest, def}}|}\}$ be the set of defective origins and destinations, respectively, under such flow. Then, the zero flow is iteratively updated according to the following procedure:

1. Look for a path between a node in $\mathcal{V}_{\text{dest, def}}$ and a node in $\mathcal{V}_{\text{or, def}}$ that is not saturated (note that for rebalancing flows, paths go from destinations to origins). If no such path exists, quit. Otherwise, go to Step 2.

2. Add the same amount of flow on all edges along the path until either (i) one of the edges becomes saturated or (ii) constraint (3.1e) is fulfilled either at the defective origin or at the defective destination. Note that the resulting flow remains a partial rebalancing flow.

3. Update sets $\mathcal{V}_{\text{or, def}}$ and $\mathcal{V}_{\text{dest, def}}$ for the new partial rebalancing flow and go to Step 1.

The algorithm terminates. To show this, we prove the invariant that if a node is no longer defective for the updated partial rebalancing flow (in other words, Step 2 ends due to condition (ii)), it will not become defective at a later stage. Consider a defective destination node v that

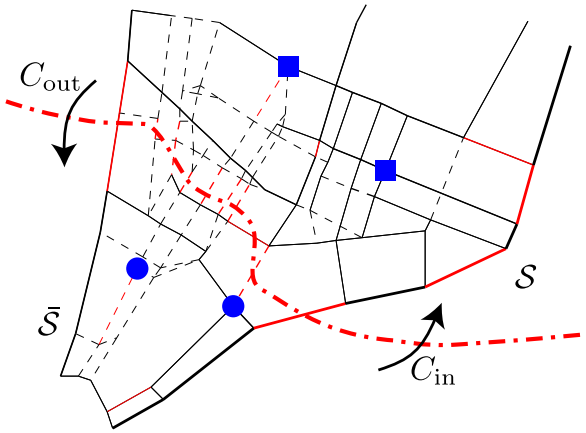

Figure 3.2: A graphical representation of Lemma 3.3.7. If there exists a set of feasible customer flows but there does not exist a set of feasible rebalancing flows, one can find a partial rebalancing flow where all the defective origins, represented as blue circles, are separated from all the defective destinations, represented as blue squares, by a cut of saturated edges (shown in red). Note that not all saturated edges necessarily belong to the cut. In the proof of Theorem 3.3.5 we show that the capacity of such a cut $(\mathcal{S}, \bar{\mathcal{S}})$ is asymmetric, i.e., $C_{\text{out}} < C_{\text{in}}$ – a contradiction that leads to the claim of Theorem 3.3.5.

becomes non-defective under the updated partial rebalancing flow (the proof for defective origins is analogous). Then, at the subsequent stage it cannot be considered as a destination in Step 1 (as it is no longer in set $\mathcal{V}_{\text{dest, def}}$). If a path that does not contain v is selected, then v stays non-defective. Otherwise, if a path that contains v is selected, then, after Step 2, both the inbound flow (that is the flow into v) and the outbound flow (that is the flow out of v) will be increased by the same quantity, and the node will stay non-defective. An induction on the stages then proves the claim. As the number of paths is finite, and sets $\mathcal{V}_{\text{or, def}}$ and $\mathcal{V}_{\text{dest, def}}$ cannot have any nodes added, the algorithm terminates after a finite number of stages.

The output of the algorithm (denoted, with a slight abuse of notation, as $\{\hat{f}_R(u,v)\}_{(u,v)}$) is a partial rebalancing flow that is not feasible (as, by assumption, there does not exist a set of feasible rebalancing flows). Therefore, by Lemma 3.3.6, such partial rebalancing flow has at least one defective origin and at least one defective destination. Let us define $\mathcal{E}_{ns} := \mathcal{E} \setminus \{(u,v) : (u,v) \text{ is saturated}\}$ as the collection of non-saturated edges under the flows $\{f_m(u,v)\}_{(u,v),m}$ and $\{\hat{f}_R(u,v)\}_{(u,v)}$. For any defective destination and any defective origin, all paths connecting them contain at least one saturated edge (due to the exit condition in Step 1). Therefore, the graph $G_{ns}(\mathcal{V}, \mathcal{E}_{ns})$ has two properties: (i) it is disconnected (that is, it is not possible to find a direct path between every pair of nodes in $\mathcal{V}$ by using edges in $\mathcal{E}_{ns}$), and (ii) a defective origin and a defective destination can not be in the same strongly connected component (hence, graph $G_{ns}(\mathcal{V}, \mathcal{E}_{ns})$ can be partitioned into at least two strongly connected components).

We now find the cut $(\mathcal{S}, \bar{\mathcal{S}})$ as follows. If a strongly connected component of G_{ns} contains defective destinations, we assign its nodes to set $\mathcal{S}$. If a strongly connected component contains defective origins, we assign its nodes to set $\bar{\mathcal{S}}$. If a strongly connected component contains neither defective origins nor destinations, we assign its nodes to $\mathcal{S}$ (one could also assign its nodes to $\bar{\mathcal{S}}$, but such choice is immaterial for our purposes). By construction, $(\mathcal{S}, \bar{\mathcal{S}})$ is a cut, and its edges are all saturated. Furthermore, set $\mathcal{S}$ only contains destination nodes, and set $\bar{\mathcal{S}}$ only contains origin nodes, which concludes the proof. □

We are now in a position to prove Theorem 3.3.5. The proof is by contradiction. Assume that a set of feasible rebalancing flows $\{f_R(u,v)\}_{(u,v)}$ does not exist. Then Lemma 3.3.7 shows that there exists a partial rebalancing flow $\{\hat{f}_R(u,v)\}_{(u,v)}$ and a cut $(\mathcal{S}, \bar{\mathcal{S}})$ such that all defective destinations under $\{\hat{f}_R(u,v)\}_{(u,v)}$ belong to $\mathcal{S}$ and all defective origins belong to $\bar{\mathcal{S}}$. Let us denote the sum of all partial rebalancing flows across cut $(\mathcal{S}, \bar{\mathcal{S}})$ as

$$\hat{F}^{\text{reb}}_{\text{out}}(\mathcal{S}, \bar{\mathcal{S}}) := \sum_{u \in \mathcal{S}, v \in \bar{\mathcal{S}}} \hat{f}_R(u,v),$$

and, analogously, define $\hat{F}^{\text{reb}}_{\text{in}}(\mathcal{S}, \bar{\mathcal{S}}) := \hat{F}^{\text{reb}}_{\text{out}}(\bar{\mathcal{S}}, \mathcal{S})$. Since all edges in the cut $(\mathcal{S}, \bar{\mathcal{S}})$ are saturated under $\{\hat{f}_R(u,v)\}_{(u,v)}$, one has, due to equation (3.1f), the equality

$$C_{\text{out}}(\mathcal{S}, \bar{\mathcal{S}}) = F_{\text{out}}(\mathcal{S}, \bar{\mathcal{S}}) + \hat{F}^{\text{reb}}_{\text{out}}(\mathcal{S}, \bar{\mathcal{S}}).$$

Additionally, again due to equation (3.1f), one has the inequality

$$F_{\text{in}}(\mathcal{S}, \bar{\mathcal{S}}) + \hat{F}^{\text{reb}}_{\text{in}}(\mathcal{S}, \bar{\mathcal{S}}) \leq C_{\text{in}}(\mathcal{S}, \bar{\mathcal{S}}).$$

Combining the above equations, one obtains

$$F_{\text{in}}(S, \bar{S}) + \hat{F}^{\text{reb}}_{\text{in}}(\mathcal{S}, \bar{\mathcal{S}}) - F_{\text{out}}(\mathcal{S}, \bar{\mathcal{S}}) - \hat{F}^{\text{reb}}_{\text{out}}(\mathcal{S}, \bar{\mathcal{S}}) \leq C_{\text{in}}(\mathcal{S}, \bar{\mathcal{S}}) - C_{\text{out}}(\mathcal{S}, \bar{\mathcal{S}}).$$

To compute $\hat{F}^{\text{reb}}_{\text{in}}(\mathcal{S}, \bar{\mathcal{S}}) - \hat{F}^{\text{reb}}_{\text{out}}(\mathcal{S}, \bar{\mathcal{S}})$, we follow a procedure similar to the one used in Lemma 3.3.1. Summing equation (3.1e) over all nodes in $\mathcal{S}$, one obtains,

$$\sum_{v \in \mathcal{S}} \left[\sum_{u \in \mathcal{V}} \hat{f}_R(u,v) + \sum_{m \in \mathcal{M}} 1_{v = t_m} \lambda_m \right] > \sum_{v \in \mathcal{S}} \left[\sum_{w \in \mathcal{V}} \hat{f}_R(v,w) + \sum_{m \in \mathcal{M}} 1_{v = s_m} \lambda_m \right].$$

The strict inequality is due to the fact that for a partial rebalancing flow that is not feasible there exists at least one defective destination (Lemma 3.3.6), which, by construction, must belong to $\mathcal{S}$. Simplifying those flows $\hat{f}_R(u,v)$ for which both u and v are in $\mathcal{S}$ (as such flows appear on both sides

of the above inequality), one obtains

$$\hat{F}^{\text{reb}}_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) - \hat{F}^{\text{reb}}_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) > \sum_{m\in\mathcal{M}} (1_{s_m\in\mathcal{S}} - 1_{t_m\in\mathcal{S}})\,\lambda_m.$$

Also, by Lemma 3.3.1,

$$F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) - F_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) = \sum_{m\in\mathcal{M}} (1_{s_m\in\mathcal{S}} - 1_{t_m\in\mathcal{S}})\,\lambda_m.$$

Collecting all the results so far, we conclude that

$$\begin{aligned} 0 &< F_{\text{in}}(S,\bar{S}) + \hat{F}^{\text{reb}}_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) - F_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) - \hat{F}^{\text{reb}}_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) \\ &= C_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) - C_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}). \end{aligned}$$

Hence, we reached the conclusion that $C_{\text{in}}(\mathcal{S},\bar{\mathcal{S}}) - C_{\text{out}}(\mathcal{S},\bar{\mathcal{S}}) > 0$, or, in other words, the capacity of graph $G(\mathcal{V},\mathcal{E})$ across cut $(\mathcal{S},\bar{\mathcal{S}})$ is *not* symmetric. This contradicts the assumption that graph $G(\mathcal{V},\mathcal{E})$ is capacity-symmetric, and the claim follows. □

The importance of Theorem 3.3.5 is twofold. First, perhaps surprisingly, it shows that for symmetric road networks it is *always* possible to rebalance the autonomous vehicles *without* increasing congestion – in other words, the rebalancing of autonomous vehicles in a symmetric road network does *not* lead to an increase in congestion. Second, from an algorithmic standpoint, if the cost function in the CRRP only depends on the customer flows (that is, $\rho = 0$ and the goal is to minimize the customers' travel times), then the CRRP problem can be *decoupled* and the customers and rebalancing flows can be solved separately without loss of optimality. This insight will be instrumental in Section 3.4 to the design of real-time algorithms for routing and rebalancing.

We conclude this section by noticing that the CRRP, from a computational standpoint, can be reduced to an instance of the Minimum-Cost Multi-Commodity Flow problem (Min-MCF), a classic problem in network flow theory [Ahuja et al., 1993]. The problem can be efficiently solved via linear programming: the number of variables of the linear program is $|\mathcal{E}|(M+1)$, and an interior point algorithm can solve the problem in $O((|\mathcal{E}|(M+1))^{3.5})$ time [Karmarkar, 1984]. The number of customer requests M admits an upper bound of $|\mathcal{V}|^2$, since each request is associated with an origin node and a destination node; the complexity of the problem does *not* depend on the number of customers or the number of autonomous vehicles. Specialized combinatorial algorithms [Goldberg et al., 1990, Leighton et al., 1995, Goldberg et al., 1998] are also available to solve the Min-MCF problem.

However, the solution to the CRRP provides *static fractional* flows, which are not directly implementable for the operation of actual AMoD systems. Practical algorithms (inspired by the theoretical CRRP model) are presented in the next section.

3.4 Real-time Congestion-Aware Routing and Rebalancing

A natural approach to routing and rebalancing would be to periodically resolve the CRRP within a receding-horizon, batch-processing scheme (a common scheme for the control of transportation networks [Seow et al., 2010, Pavone et al., 2012, Zhang et al., 2016]). This approach, however, is not directly implementable as the solution to the CRRP provides *fractional* flows (as opposed to routes for the *individual* vehicles). This shortcoming can be addressed by considering an integral version of the CRRP (dubbed integral CRRP), whereby the flows are *integer*-valued and can be thus easily translated into routes for the individual vehicles, e.g. through a flow decomposition algorithm [Ford and Fulkerson, 1962]. The integral CRRP, however, is an instance of the integral Minimum-Cost Multi-Commodity Flow problem, which is known to be NP-hard [Karp, 1975, Even et al., 1976]. Naïve rounding techniques are inapplicable: rounding a solution for the (non-integral) CRRP does not yield, in general, feasible integral flows, and hence feasible routes. For example, continuity of vehicles and customers can not be guaranteed, and vehicles may appear and disappear along a route. In general, to the best of our knowledge, there are no polynomial-time approximation schemes for the integral Minimum-Cost Multi-Commodity Flow problem.

On the positive side, the integral CRRP admits a decoupling result akin to Theorem 3.3.5: given a set of feasible, *integral* customer flows, one can always find a set of feasible, *integral* rebalancing flows. (In fact, the proof of Theorem 3.3.5 does not exploit anywhere the property that the flows are fractional, and thus the proof extends virtually unchanged to the case where the flows are integer-valued). Our approach is to leverage this insight (and more in general the theoretical results from Section 3.3) to design a heuristic, yet efficient approximation to the integral CRRP that (i) scales to large-scale systems, and (ii) is general, in the sense that can be broadly applied to time-varying, asymmetric networks.

Specifically, we consider as objective the minimization of the customers' travel times, which, from Section 3.3 and the aforementioned discussion about the generalization of Theorem 3.3.5 to integral flows, *suggests* that customer routing can be decoupled from vehicle rebalancing (strictly speaking, this statement is only valid for static and symmetric networks – its generalization beyond these assumptions will be addressed numerically in Section 3.5). Accordingly, to emulate the real-world operation of an AMoD system, we divide a given city into geographic regions (also referred to as "stations" in some formulations) [Pavone et al., 2012, Zhang and Pavone, 2016]. Each arriving customer is assigned the closest vehicle *within that region* (vehicle imbalance across regions is handled separately by the vehicle rebalancing algorithm, discussed below). The problem of determining the partition of the city in regions is a special case of the stochastic facility location problem [Cornuejols et al., 1990]; in our experiments, we select regions with k-means clustering of past demand, as discussed in Section 3.5. Leveraging the decoupling result in Theorem 3.3.5, we focus on optimizing the paths of the *rebalancing* vehicles in order to minimize congestion in the network: accordingly, we apply a greedy, computationally efficient event-based approach for customer routing

where customers are routed to their destinations using the shortest-time path as computed by an A^* algorithm [Hart et al., 1968]. The travel time along each edge is computed using a heuristic delay function that is related to the current volume of traffic on each edge. A popular heuristic is the simple Bureau of Public Roads (BPR) delay model [Bureau of Public Roads, 1964], which computes the travel time on each edge $(u, v) \in \mathcal{E}$ as

$$t_d(u, v) := t(u, v) \left(1 + \alpha \left(\frac{f(u, v)}{c(u, v)} \right)^{\beta} \right),$$

where $f(u, v) := \sum_{m=1}^{M} f_m(u, v) + f_R(u, v)$ is the total flow on edge (u, v), and α and β are usually set to 0.15 and 4 respectively.

Separately from customer routing, vehicle rebalancing from one region to another is performed every $t_{\text{hor}} > 0$ time units as a batch process (unlike customer routing, which is an event-based process). The rebalancing frequency t_{hor} is selected so as to smooth short-term fluctuations in customer demand and, secondarily, to ensure that a rebalancing solution can be numerically computed within one batch period. Denote by $v_i(t)$ the number of vehicles in region i at time t, and by $v_{ji}(t)$ the number of vehicles traveling from region j to i. Let $v_i^{\text{own}}(t) := v_i(t) + \sum_j v_{ji}(t)$ be the number of vehicles currently "owned" by region i (i.e., at or enroute to region i). Denote by $v_i^e(t)$ the number of excess vehicles in region i, or the number of vehicles left after servicing the customers waiting within region i. From its definition, $v_i^e(t)$ is given by $v_i^e(t) = v_i^{\text{own}}(t) - c_i(t)$, where $c_i(t)$ is the number of customers within region i. Finally, denote by $v_i^d(t)$ the desired number of vehicles within region i. For example, for an even distribution of excess vehicles, $v_i^d(t) \propto \sum_i v_i^e(t)/N$, where N is the number of regions. Note that the $v_i^d(t)$'s are rounded so they assume integer values. The set of origin regions (i.e., regions that should send out vehicles), S_R, and destination regions (i.e., regions that should receive vehicles), T_R, for the rebalancing vehicles are then determined by comparing $v_i^e(t)$ and $v_i^d(t)$, specifically,

$$\begin{aligned} &\text{if } v_i^e(t) > v_i^d(t), \quad \text{region } i \in S_R \\ &\text{if } v_i^e(t) < v_i^d(t), \quad \text{region } i \in T_R. \end{aligned}$$

We assume the residual capacity $c_R(u, v)$ of an edge (u, v), defined as the difference between its overall capacity $c(u, v)$ and the current number of vehicles along that edge, is known and remains approximately constant over the rebalancing time horizon. In case the overall rebalancing problem is not feasible (i.e. it is not possible to move all excess vehicles to regions that have a deficit of vehicles while satisfying the congestion constraints), we define slack variables with cost C that allow the optimizer to select a subset of vehicles and rebalancing routes of maximum cardinality such that each link does not become congested. The slack variables are denoted as ds_i for each $i \in S_R$, and dt_j for each $j \in T_R$.

Every t_{hor} time units, the rebalancing vehicle routes are computed by solving the following integer linear program

$$\underset{\substack{f_R(\cdot,\cdot),\\ \{ds_i\},\{dt_j\}}}{\text{minimize}} \quad \sum_{(u,v)\in\mathcal{E}} t(u,v)\, f_R(u,v) + \sum_{i\in S_R} C ds_i + \sum_{i\in T_R} C dt_i \tag{3.2a}$$

$$\text{subject to} \quad \sum_{u\in\mathcal{V}} f_R(u,v) + 1_{v\in S_R}(v_v^e(t) - v_v^d(t) - ds_v) = \sum_{w\in V} f_R(v,w) + 1_{v\in T_R}(v_v^d(t) - v_v^e(t) - dt_v),$$

$$\text{for all } v \in \mathcal{V} \tag{3.2b}$$

$$f_R(u,v) \le c_R(u,v), \qquad \text{for all } (u,v)\in\mathcal{E} \tag{3.2c}$$

$$f_R(u,v) \in \mathbb{N}, \qquad \text{for all } (u,v)\in\mathcal{E} \tag{3.2d}$$

$$ds_i, dt_j \in \mathbb{N}, \qquad \text{for all } i\in S_R, j\in T_R \tag{3.2e}$$

The set of (integral) rebalancing flows $\{f_R(u,v)\}_{(u,v)}$ is then decomposed into a set of rebalancing paths via a flow decomposition algorithm [Ford and Fulkerson, 1962]. Each rebalancing path connects one origin region with one destination region: thus, rebalancing paths represent the set of routes that excess vehicles should follow to rebalance to regions with a deficit of vehicles.

The rebalancing optimization problem is an instance of the Minimum Cost Flow problem. If all edge capacities are integral, the linear relaxation of the Minimum Cost Flow problem enjoys a totally unimodular constraint matrix [Ahuja et al., 1993]. Hence, the linear relaxation will necessarily have an integer optimal solution, which will be a fortiori an optimal solution to the original Minimum Cost Flow problem. It follows that an integer-valued solution to the rebalancing optimization problem can be computed efficiently, namely in polynomial time, e.g., via linear programming.

The number of variables of Problem (3.2) is $|\mathcal{E}| + 2|M|$: the problem size depends linearly on the number of road links and on the number of transportation requests (which admits an upper bound corresponding to the square of the number of regions). Remarkably, the problem size does not depend on the number of customers or the number of vehicles present in the system. Thus, the problem can be solved in $O((|\mathcal{E}|+2|M|)^{3.5})$ with an interior point algorithm [Karmarkar, 1984]. Several efficient combinatorial algorithms [Tardos, 1985, Ahuja et al., 1993] are also available, whose computational performance is typically significantly better.

The favorable computational properties of the routing and rebalancing algorithm presented in this section enable application to large-scale systems, as described next.

3.5 Numerical Experiments

In this section, we evaluate the validity of the capacity-symmetry assumption for several major U.S. cities (Section 3.5.1), characterize the effect of rebalancing on congestion in asymmetric networks (Section 3.5.2), and explore the performance of the algorithm presented in Section 3.4 on real-world

Urban center	Avg. frac. cap. disp.	Std. dev.
Chicago, IL	$1.2972 \cdot 10^{-4}$	$1.003 \cdot 10^{-4}$
New York, NY	$1.6556 \cdot 10^{-4}$	$1.304 \cdot 10^{-4}$
Colorado Springs, CO	$3.1772 \cdot 10^{-4}$	$2.308 \cdot 10^{-4}$
Los Angeles, CA	$0.9233 \cdot 10^{-4}$	$0.676 \cdot 10^{-4}$
Mobile, AL	$1.9368 \cdot 10^{-4}$	$1.452 \cdot 10^{-4}$
Portland, OR	$1.0769 \cdot 10^{-4}$	$0.778 \cdot 10^{-4}$

Table 3.1: Average fractional capacity disparity for several major urban centers in the United States.

road topologies with real customer demands (Section 3.5.3).

3.5.1 Capacity Symmetry within Urban Centers in the US

The existential result in Section 3.3, namely Theorem 3.3.5, relies on the assumption that the road network is capacity-symmetric, i.e., for every cut $(\mathcal{S}, \bar{\mathcal{S}}), C_{\text{out}}(\mathcal{S}, \bar{\mathcal{S}}) = C_{\text{in}}(\mathcal{S}, \bar{\mathcal{S}})$. One may wonder whether this assumption is (approximately) met in practice. From an intuitive standpoint, one might argue that transportation networks within urban centers are indeed *designed* to be capacity symmetric, so as to avoid accumulation of traffic flow in some directions. We corroborate this intuition by computing the imbalance between the outbound capacity (i.e., C_{out}) and the inbound capacity (i.e., C_{in}) for 1000 randomly-selected cuts within several urban centers in the United States. For each edge $(u, v) \in \mathcal{E}$, we approximate its capacity as proportional to the product of the speed limit $v_{\max}(u, v)$ on that edge and the number of lanes $L(u, v)$, that is, $c(u, v) \propto v_{\max}(u, v) \cdot L(u, v)$. The road graph $G(\mathcal{V}, \mathcal{E})$, the speed limits, and the number of lanes are obtained from OpenStreetMap data [Haklay and Weber, 2008].

For a cut $(\mathcal{S}, \bar{\mathcal{S}})$, we define its fractional capacity disparity $D(\mathcal{S}, \bar{\mathcal{S}})$ as

$$D(\mathcal{S}, \bar{\mathcal{S}}) := 2 \, \frac{\left| C_{\text{out}}(\mathcal{S}, \bar{\mathcal{S}}) - C_{\text{in}}(\mathcal{S}, \bar{\mathcal{S}}) \right|}{C_{\text{out}}(\mathcal{S}, \bar{\mathcal{S}}) + C_{\text{in}}(\mathcal{S}, \bar{\mathcal{S}})}.$$

Table 3.1 shows the average fractional capacity disparity (over 1000 samples) for several US urban centers. As expected, the road networks for such cities appear to posses a very high degree of capacity-symmetry, which validates the symmetry assumption made in Section 3.3.

3.5.2 Characterization of Congestion due to Rebalancing in Asymmetric Networks

The theoretical results in Section 3.3 are proven for capacity-symmetric networks, which are in general a reasonable model for typical urban road networks (as shown in the previous section). Nevertheless, it is of interest to characterize the applicability of our theoretical results (chiefly, the

existential result in Theorem 3.3.5) to road networks that significantly violate the capacity-symmetry property. In other words, we investigate to what degree rebalancing might lead to an increase in congestion if the network is asymmetric.

To this purpose, we compute solutions to the CRRP for road networks with varying degrees of capacity asymmetry and compare the corresponding travel times to those obtained by computing optimal routes in the absence of rebalancing (as would be the case, e.g., if the vehicles were privately owned). We focus on the road network portrayed in Figure 3.3, derived from OpenStreetMap data [Haklay and Weber, 2008]. With 1351 nodes and 3137 edges, the road network captures all major streets and avenues in Manhattan. Transportation requests are based on actual taxi rides in New York City on March 1, 2012 from 6 to 8 p.m. (courtesy of the New York Taxi and Limousine Commission). We clustered all departures and arrivals into 100 stations and considered only origin-destination pairs with more than 5 customers per hour on average (27,571 or 51.1% of all trips). As in Section 3.5.1, we approximated the capacity of each road as proportional to the product of the speed limit $v_{\max}(u, v)$ and the number of lanes $L(u, v)$. To ensure that the flow induced by the trips would induce a small amount of congestion *before* introducing any asymmetry, we scaled down the capacities of all roads uniformly. Empirically, a scaling factor of 0.041 (or 25× reduction) introduced sufficient congestion, which is consistent with the observations that (i) we only consider 51.1% of true customer flow due to network filtering and (ii) taxis only constitute a fraction of the vehicles in Manhattan.

To investigate the effects of network asymmetry, we introduce an *artificial capacity asymmetry* into the baseline Manhattan road network by progressively reducing the capacity of all northbound avenues. In order to *quantify* the effect rebalancing has on congestion and travel times, we assign slack variables $\delta_C(u, v)$, associated with a cost $c_c(u, v)$, to each congestion constraint (3.1f). The cost $c_c(u, v)$ is selected such that the optimization algorithm selects a congestion-free solution whenever one is available. Once a solution is found, the actual travel time on each (possibly congested) link is computed using the heuristic BPR delay model [Bureau of Public Roads, 1964] presented in Section 3.4. This approach maintains feasibility even in the congested traffic regime, and hence it allows us to assess the impact of rebalancing on congestion in asymmetric networks.

Table 3.2 summarizes the results of our simulations. In the baseline case, no artificial capacity asymmetry is introduced, i.e., the fractional capacity reduction of northbound avenues is equal to 0%. Overall, the difference between the travel times in the two cases is very small (approximately 1.16%), which is consistent with the fact that New York City's road graph has largely symmetric capacity, as shown in Section 3.5.1. Interestingly, even with a massive (60%) reduction in northbound capacity, travel times with and without rebalancing vehicles are practically equivalent (within 0.12%). Collectively, these results show that the existential result in Theorem 3.3.5, proven under the assumption of a symmetric network, appears to extend (albeit approximately) to asymmetric networks. In particular, it appears that vehicle rebalancing does not lead to an appreciable increase

in congestion under very general conditions.

	Average travel time [s]		
Cap. reduction	Without reb.	With reb.	Travel time increase
0%	58.00	58.67	1.16 %
10%	58.12	59.15	1.76 %
20%	58.49	59.67	2.02 %
30%	59.26	60.56	2.20 %
40%	60.65	61.78	1.86 %
50%	63.66	64.55	1.40 %
60%	72.04	72.13	0.12 %

Table 3.2: Customer travel times with and without rebalancing for different levels of network asymmetry.

3.5.3 Congestion-Aware Real-time Rebalancing

In this section we evaluate the performance of the real-time routing and rebalancing algorithm presented in Section 3.4, and compare it to a baseline approach that does not explicitly take congestion into account. We simulate 8,000 vehicles providing service to approximately 480,000 actual taxi requests over 24 hours on March 1, 2012, using the same Manhattan road network as in the previous section (shown in Figure 3.3).

We use the MATSim agent-based traffic simulator [Horni et al., 2016] and modify its DVRP extension [Maciejewski et al., 2017] to accommodate station-based dispatching and rebalancing of idle vehicles[2]. MATSim uses an agent-based, microscopic traffic model where each road is abstracted as a capacitated FIFO queue. Vehicles can enter a road link only if that link has not reached its maximum capacity. Once a vehicle enters a link, it can leave it after (i) the free-flow travel time on the link has elapsed and (ii) it has reached the head of the queue. Other delay factors such as traffic signals, turning times, and pedestrian blocking are not simulated.

Taxi requests are clustered into 100 stations corresponding to subsets of the nodes in the network. The 481,989 trip requests from the same New York Taxi and Limousine Commission data set used in Section 3.5.2 are simulated using a time step of 1 second.

Three algorithms are evaluated, namely (i) a nearest-neighbor dispatching algorithm that performs no rebalancing of idle vehicles, (ii) the congestion-aware routing and rebalancing algorithm presented in Section 3.4, and (iii) a baseline rebalancing algorithm. The baseline approach is derived from the real-time rebalancing algorithm presented in [Zhang and Pavone, 2016], which is a point-to-point algorithm that computes rebalancing origins and destinations without considering the underlying road network. In the baseline approach, customer routes are computed in the same way as in Section 3.4. For rebalancing, the origins and destinations are first solved using the algorithm

[2]The source code for the modified taxi extension is available at `http://dx.doi.org/10.5281/zenodo.1048415`

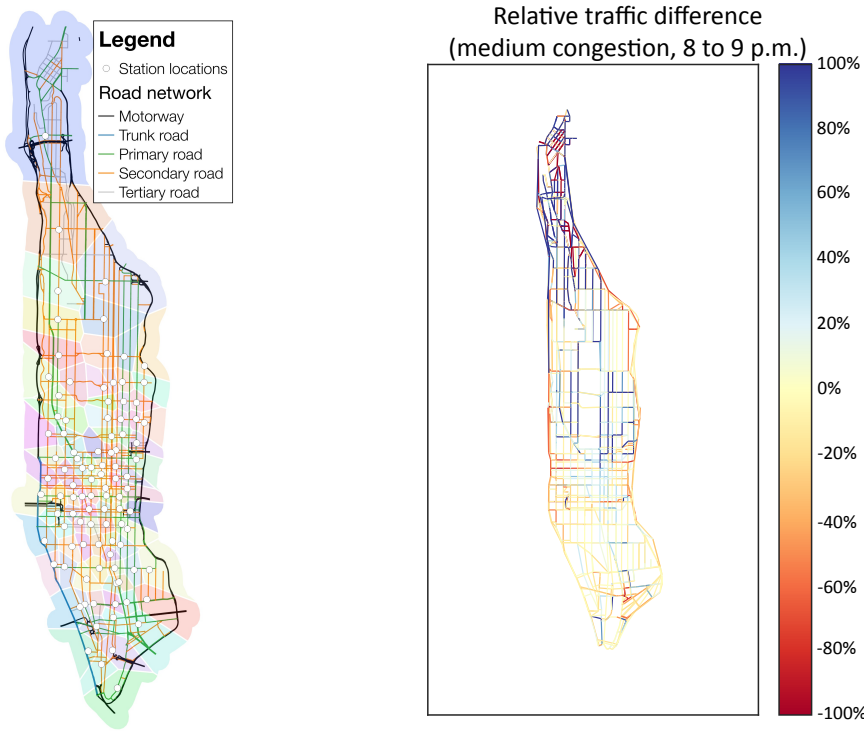

Figure 3.3: *Left:* Manhattan road network and partition of the city in regions. The roads' speed limit is determined by their type; the capacity of each road link is proportional to the speed limit and to the number of lanes. Station locations are computed with k-means clustering of historical travel demand; regions (shown in the background as colored areas) are a Voronoi partition with stations as the seeds. *Right:* Performance of the "real-time congestion-aware rebalancing algorithm" as compared to the baseline algorithm in [Zhang and Pavone, 2016]. The color of each road corresponds to the percent *difference* in the number of vehicles traversing it between the congestion-aware and baseline rebalancing algorithms–blue indicating a reduction in congestion using the congestion-aware algorithm.

provided in [Zhang and Pavone, 2016], then the routes are computed using the A^* algorithm much like the customer routes.

In [Zhang et al., 2016], the authors show that, in the low-congestion regime, the baseline algorithm offers near-optimal performance and outperforms several state-of-the-art dispatching and rebalancing algorithms. However, the baseline algorithm ignores the potential for additional congestion induced by rebalancing vehicles, and thus, performs poorly for highly congested networks. On the other hand, this performance penalty is reflected in the nearest-neighbor rebalancing algorithm, causing it to perform much better in high congestion cases where this penalty outweighs the benefit of pre-positioning empty vehicles.

IBM ILOG CPLEX was used to implement the congestion-aware and baseline rebalancing algorithms. The computation time was (on average) under 0.5 s on commodity hardware (Intel Core

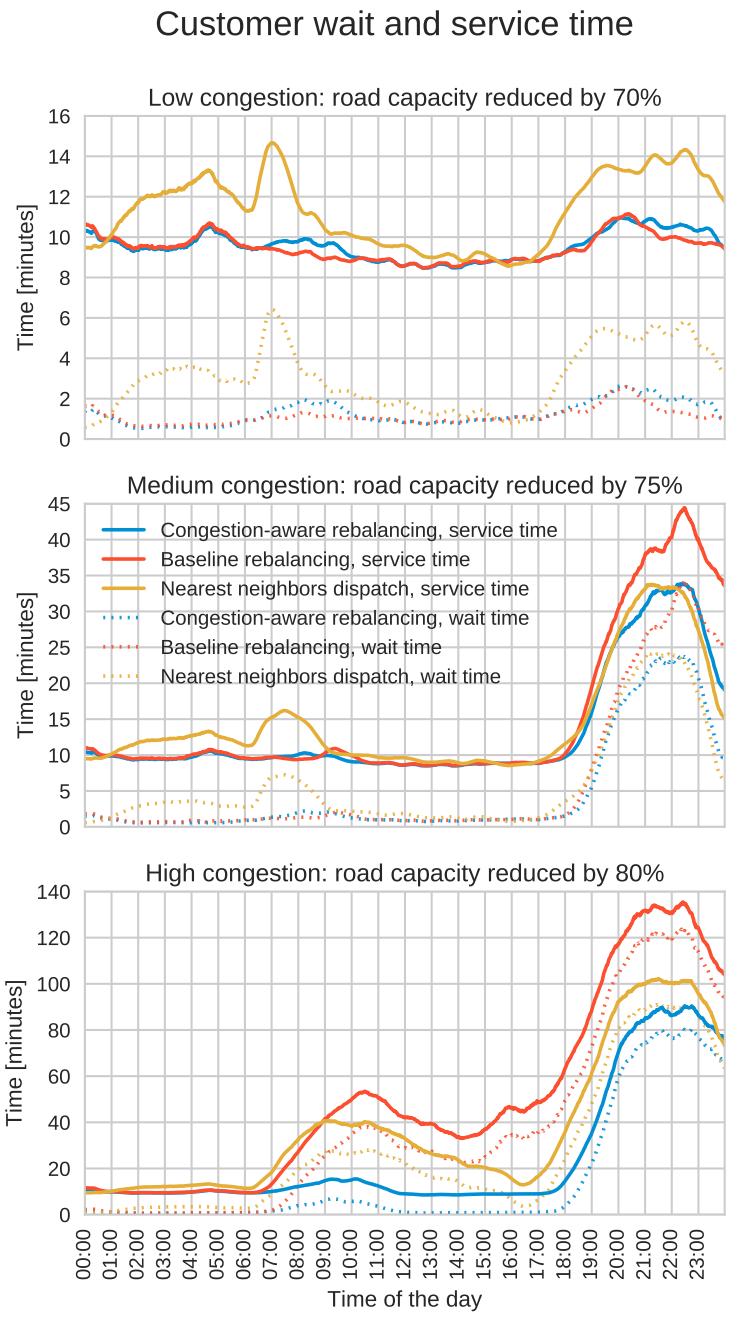

Figure 3.4: Comparison of customer wait and service times from different rebalancing and dispatching algorithms for low, medium, and high levels of congestion. The congestion-aware algorithm recovers the asymptotic behavior of the baseline rebalancing algorithm for low levels of congestion, and it outperforms both the baseline rebalancing algorithm and the nearest-neighbor dispatch algorithm for high levels of congestion.

i7-5960, 64 GB RAM); the maximum computation time was 4.52 s. To account for the computation time, release of the rebalancing routes was delayed by 5 s in the simulation framework.

For each algorithm, we simulated three scenarios corresponding to low, medium and very high levels of congestion (corresponding to a reduction of the road network's nominal capacity of 70%, 75% and 80% respectively). Figure 3.4 presents a summary of the performance results. Note that the service time represents the total time a customer spends in the system (waiting time plus traveling time).

For low levels of congestion, the performance of the congestion-aware algorithm closely tracks the performance of the baseline rebalancing algorithm. The nearest-neighbor dispatch algorithm performs significantly worse than either rebalancing algorithm in this regime. For medium and high levels of congestion, performance of the baseline algorithm is significantly degraded: as expected, rebalancing trips cause significant congestion in the network, as exemplified in Figure 3.3. The nearest-neighbor dispatch algorithm offers better performance than the baseline rebalancing algorithm in this regime: when the road network is congested, not rebalancing at all appears to be preferable to excessive rebalancing.

The congestion-aware algorithm recovers the performance of the baseline rebalancing algorithm in the low-congestion regime and the performance of the nearest-neighbor dispatcher in the medium-congestion regime; in the high-congestion regime, it outperforms both. By selectively rebalancing vehicles where and when congestion is low, and by selecting rebalancing routes that do not increase congestion, the algorithm is able to mediate between the behavior of the baseline rebalancing algorithm and the behavior of the nearest-neighbor algorithm depending on the level of congestion: this results in good performance both in terms of network congestion and in terms of customer service times across a wide range of congestion regimes.

3.6 Conclusions and Future Work

In this chapter we presented a network flow model of an autonomous mobility-on-demand system on a capacitated road network. We formulated the routing and rebalancing problem and showed that on capacity-symmetric road networks, it is always possible to route rebalancing vehicles in a coordinated way that does not increase traffic congestion. Using a model road network of Manhattan, we showed that rebalancing did not increase congestion even for moderate degrees of network asymmetry. We leveraged the theoretical insights to develop a computationally efficient real-time congestion-aware routing and rebalancing algorithm and demonstrated its superior performance compared to state-of-the-art point-to-point rebalancing algorithms through simulation. This highlighted the importance of congestion awareness in the design and implementation of control strategies for a fleet of self-driving vehicles.

This work opens the field to many future avenues of research. First, it is of interest to directly use

the solution to the integral CRRP as a practical real-time routing algorithm to compute congestion-free routes for customer vehicles and rebalancing vehicles alike. While the integral CRRP is in general intractable, in Chapter 4 we leverage randomized algorithms [Raghavan and Tompson, 1987, Srinivasan, 1999] to compute high-quality approximate solutions for large-scale systems. Second, from a modeling perspective, we would like to study the inclusion of stochastic information (e.g., demand prediction, travel time uncertainty) for the routing and rebalancing problem, as well as a richer set of performance metrics and constraints (e.g., time windows to pick up customers). Third, it is worthwhile to study how our results give intuition into business models for autonomous urban mobility (e.g. fleet sizes). Fourth, it is of interest to explore other approaches that may reduce congestion, including ride-sharing, demand staggering, and integration with public transit to create an intermodal transportation network. Fifth, we plan to study the deployment of large *electric* AMoD fleets and, in particular, to characterize the interaction between such fleets and the electric power network. Sixth, we would like to explore decentralized architectures for cooperative routing and rebalancing. Finally, we would like to demonstrate the real-world performance of the algorithms by implementing them on real fleets of self-driving vehicles.

Chapter 4

Congestion-Aware Randomized Routing in Autonomous Mobility-on-Demand Systems

In this chapter we study the routing and rebalancing problem for a fleet of autonomous (i.e., self-driving) vehicles providing on-demand transportation within a capacitated urban road network. In Chapter 3, we designed a model for congestion-aware control of AMoD systems. However, the real-time congestion-aware routing algorithm proposed in Section 3.4 only computes routes for rebalancing (customer-empty) vehicles, whereas customer-carrying vehicles are routed according to shortest-path heuristics. In this chapter, we extend real-time congestion-aware routing techniques to *customer-carrying* as well as customer-empty vehicles.

We leverage a network flow model with integral flows, where unit flows represent individual vehicle routes. We show that finding a *congestion-free* integral solution to the routing and rebalancing problem is NP-hard. We thus provide a polynomial-time randomized algorithm which finds a *low-congestion* solution that approximately minimizes the travel time of passengers. The algorithm enjoys correctness guarantees in terms of (1) theoretical bounds on the probability of violating the congestion constraints, and (2) approximation factor with respect to the minimum expected passenger travel time that can be achieved with a congestion-free solution for road networks with symmetric edges. We evaluate the performance of the algorithm on a high-fidelity model of the Manhattan road network. We show that the proposed algorithm introduces a very small amount of congestion, while offering travel times that are very close to (and sometimes better than) what is achievable with a congestion-free solution.

Literature review: The problem of vehicle rebalancing has been studied for both carsharing [Barth and Todd, 1999, Smith et al., 2013, Zhang and Pavone, 2015] and AMoD systems

[Pavone et al., 2011, Zhang and Pavone, 2016] where the underlying network is a complete graph (point-to-point routing). These approaches (i) seek to only optimize rebalancing routes, and (ii) do not consider congestion effects caused by routing customers and rebalancing vehicles on a road network. The routing and rebalancing problem we study in this chapter is similar to the one-to-one pickup and delivery problem on a Euclidean plane [Treleaven et al., 2013] or on road networks [Treleaven et al., 2012], for which polynomial-time, asymptotically-optimal algorithms exist. However, these formulations do not take into account congestion. In particular, for the case of AMoD, the presence of empty-traveling rebalancing vehicles on the road was believed to have a negative impact on congestion [Templeton, 2010, Levin et al., 2016]. Recent work, however, suggests that with optimized routing this need not be the case, as shown in Chapter 3. The work in Chapter 3, however, is limited to a *macroscopic* analysis of system behavior, and does not directly allow the computation of individual vehicle routes. In contrast, in this chapter we focus on the computation of *individual routes* for the vehicles. In a nutshell, this is achieved by modifying the model in Chapter 3 to accommodate integral flows.

In the field of transportation science, our problem is similar to the dynamic traffic assignment (DTA) problem [Janson, 1991]. The two major differences between our problem and DTA approaches are that (i) DTA only optimizes customer routes, not rebalancing routes, and (ii) most DTA methods optimize for user equilibrium, where a decision by any vehicle to change routes would necessarily lead to an increase in travel time. A key advantage of AMoD systems is that vehicles can be centrally coordinated and optimally routed: accordingly, in this chapter we seek a *system-optimal* solution, as opposed to a user equilibrium.

Finally, traffic congestion is a well-studied topic in transportation science. Existing congestion models vary in degree of fidelity: from basic models establishing the relationship among speed, density, and flow [Lighthill and Whitham, 1955], to simulation-based microscopic car-following models [Treiber et al., 2000]. However, for the most part, the purpose of traffic modeling has been the *analysis* of traffic patterns rather than the *active coordination* and *control* of traffic. In this chapter, we leverage simple, yet efective traffic models that are amenable to tractable analysis and control.

Statement of contributions: Our goal is to efficiently find optimal customer-carrying and vehicle rebalancing routes that minimize the overall travel times (or, equivalently, the overall number of vehicles) in the presence of congestion effects. Our strategy is to design a polynomial-time approximation algorithm to the problem of *congestion-free* routing and rebalancing with minimum number of vehicles (whereby congestion follows a simple threshold model based on road capacities). This latter problem serves as a *proxy* for solving the general problem of minimizing travel times in the presence of congestion effects, and allows us to avoid the use of sophisticated congestion models for the computation of travel times (that would further compound the complexity of the routing problem).

Specifically, our contribution is threefold: First, we propose an approximate randomized algorithm for the solution to the congestion-free routing and rebalancing problem. The algorithm produces solutions that violate the capacity constraints by small amounts – a probabilistic characterization of the degree of constraint violation is analytically derived for symmetric road networks. Second, we provide a semi-analytical characterization of the algorithm's approximation factor for symmetric road networks. The approximation factor represents the ratio between the total customer travel time needed with randomized routing under an empirical congestion model, and the optimal *congestion-free* total customer travel time. Third, we validate the performance of the randomized routing algorithm on a realistic road network with real-world customer demand. The randomized technique provides a solution that compares very favorably with the optimal congestion-free one, and, in general, allows one to route thousands of vehicles with minimal impact on the transportation network.

Organization: The rest of the chapter is organized as follows: in Section 4.1 we introduce notation and rigorously define the integer congestion-free routing and rebalancing problem. A randomized approximation algorithm for such a problem is presented in Section 4.2, together with correctness guarantees. Numerical results corroborating our analysis are provided in Section 4.3, while conclusions and future directions are summarized in Section 4.4.

4.1 Model Description and Problem Formulation

We define the routing and rebalancing problem as a network flow problem on a capacitated road network. The model we adopt is largely similar to the one presented in the previous chapter. The key difference is our assumption that customer demands, network flows, and road capacities are integral. This assumption is in line with our goal of solving for individual vehicle routes, which can then be used as part of a practical real-time control algorithm for large vehicle fleets.

4.1.1 Congestion model

Two congestion models are adopted in this chapter. A simpler *synthesis* model is used to compute vehicle routes in Sections 4.2.2 and 4.2.3, whereas a higher-fidelity *analysis* model is used to (a) analytically characterize the performance of the approximate routing algorithm presented in Section 4.2.6 and (b) numerically evaluate algorithm's performance with real-world customer demand in Section 4.3.

Both models are consistent with classical traffic flow theory [Wardrop, 1952], [Lighthill and Whitham, 1955]. In classical traffic flow theory, for a given road, vehicle speeds tend to remain relatively constant at low vehicle densities (called the "free flow" speed) [Wardrop, 1952]. The flow rate (i.e., the number of vehicles traversing a road per unit time) grows with vehicle density up to a critical value (referred to in the literature as the *capacity* of the road), at which point vehicle

speeds and flow rate decrease significantly, signaling the onset of congestion. The capacity of the road is reached when the flow rate is maximized.

The synthesis congestion model adopted in this chapter is a threshold model. The vehicle density on each road is constrained to be no larger than the critical road density, which corresponds to the road capacity. Every vehicle travels at the free flow speed. This model captures the behavior of traffic up to the onset of congestion: furthermore, any set of vehicle routes that respects the capacity constraints on every road is guaranteed to be *congestion-free*.

The analysis congestion model offers a characterization of the congested behavior of a road. We assume that the travel time $\tilde{t}$ is strictly increasing in the flow rate f traversing the link; we place no further assumptions on its shape. One such congestion model is the widely used Bureau of Public Roads (BPR) link delay model [Bureau of Public Roads, 1964], which models the travel time on a link as

$$\tilde{t}(f) = t_0 \left(1 + 0.15 \left(f/c\right)^4\right) \tag{4.1}$$

where $\tilde{t}$ is the travel time associated with flow rate f, t_0 is the free-flow travel time and c is the capacity of the road.

4.1.2 Integral Congestion-free Routing and Rebalancing

The network flow model adopted in this section is very similar to the one presented in Chapter 2 and Section 3.2.2, but requires customer and rebalancing flows to be integral.

In this chapter, we assume that the road network is *symmetric*: $(u, v) \in \mathcal{E} \Leftrightarrow (v, u) \in \mathcal{E}$ and $c(u, v) = c(v, u)\ \forall (u, v) \in \mathcal{E}$. We will relax this assumption in the numerical experiments presented in Section 4.3.

The integral Congestion-free Routing and Rebalancing problem (i-CRRP) is defined as follows,

Definition 4.1.1 (i-CRRP)**.** *Given a network flow model of an AMoD system, compute a set of routes that*

(i) transfers customers to their desired destinations (customer-carrying trips),

(ii) rebalances vehicles throughout the network to realign the vehicle fleet with the customers' demands (customer-empty, or rebalancing trips),

(iii) does not cause congestion on any road link, and

(iv) minimizes the overall travel time of customer-carrying vehicles.

The i-CRRP can be cast as a mixed-integer linear program. We represent passenger routes for passengers of class $m \in \mathcal{M}$ with the integral network flow $\{f_m(u, v)\}_{(u,v)}$. Similarly, we represent rebalancing routes with the integral network flow $\{f_{\mathrm{R}}(u, v)\}_{(u,v)}$. Given a capacitated network $G(\mathcal{V}, \mathcal{E})$ and a set of transportation requests $\mathcal{M} = \{(s_m, t_m, \lambda_m)\}$, the i-CRRP entails solving

$$\underset{f_m(\cdot,\cdot),f_{\mathrm{R}}(\cdot,\cdot)}{\text{minimize}} \quad \sum_{m\in\mathcal{M}}\sum_{(u,v)\in\mathcal{E}} t(u,v)f_m(u,v) \tag{4.2a}$$

$$\text{subject to} \quad \sum_{u\in\mathcal{V}} f_m(u,v) + 1_{v=s_m}\lambda_m = \sum_{w\in\mathcal{V}} f_m(v,w) + 1_{v=t_m}\lambda_m \qquad \forall m\in\mathcal{M}, v\in\mathcal{V} \tag{4.2b}$$

$$\sum_{u\in\mathcal{V}} f_{\mathrm{R}}(u,v) + \sum_{m\in\mathcal{M}} 1_{v=t_m}\lambda_m = \sum_{w\in\mathcal{V}} f_{\mathrm{R}}(v,w) + \sum_{m\in\mathcal{M}} 1_{v=s_m}\lambda_m \qquad \forall v\in\mathcal{V} \tag{4.2c}$$

$$f_{\mathrm{R}}(u,v) + \sum_{m\in\mathcal{M}} f_m(u,v) \le c(u,v) \qquad \forall (u,v)\in\mathcal{E} \tag{4.2d}$$

$$f_m(u,v) \in \mathbb{N}_{\ge 0} \qquad \forall m\in\mathcal{M}, (u,v)\in\mathcal{E} \tag{4.2e}$$

$$f_{\mathrm{R}}(u,v) \in \mathbb{N}_{\ge 0} \qquad \forall (u,v)\in\mathcal{E}. \tag{4.2f}$$

Equation 4.2a captures the goal of minimizing the number of vehicles on the road. Equations 4.2b and 4.2e ensure that each customer-carrying flow $\{f_m(u,v)\}_{(u,v),m}$ is an integral network flow. Equations 4.2c and 4.2f ensure that the rebalancing flow $\{f_{\mathrm{R}}(u,v)\}_{(u,v)}$ is an integral network flow. Finally, Equation 4.2d enforces the capacity constraint for every link. One can easily recognize that the i-CRRP is an instance of an integral minimum-cost multicommodity flow problem (Min-MCF) where customers and rebalancing vehicles are interpreted as commodities. We leverage this observation to characterize its complexity.

Theorem 4.1.2 (Complexity of the i-CRRP)**.** *The decision version of the i-CRRP is NP-complete.*

Proof. We show that any instance of SAT can be reduced to an instance of two-customer i-CRRP (the extension to $k \ge 3$ customers is trivial).

The i-CRRP is an instance of the integral Minimum-Cost Multi-Commodity Flow (Min-MCF) problem. However, in the i-CRRP, one of the commodities is always a a *rebalancing* commodity, i.e. a commodity whose origin nodes and corresponding intensities coincide with the set of destination nodes and intensities of the other commodities, and vice versa.

The decision version of the integral Min-MCF is known to be NP-complete [Karp, 1975], [Even et al., 1976] in the general case. Theorem 3 in [Even et al., 1976] can be modified to also hold for the special case of MCF problems with a rebalancing commodity. In [Even et al., 1976], the authors show that any instance of k-SAT (with $k \ge 3$) can be cast as an instance of integral two-commodity flow: the first commodity (with intensity 1) ensures that every variable is either true or false but not both, whereas the second commodity (with intensity k) models clause satisfaction. Figure 4.1 shows a graphical depiction of the proof.

Note that, in the construction in [Even et al., 1976], no directed path exists from the destination of commodities 1 and 2 to their origins. We modify the construction by introducing a directed edge with capacity 1 from the destination of commodity 1 to its origin and a directed edge with capacity k from the destination of commodity 2 to its origin. In order for a rebalancing network flow to exist, both edges must be saturated; the rest of the graph is identical to the graph in [Even et al., 1976].

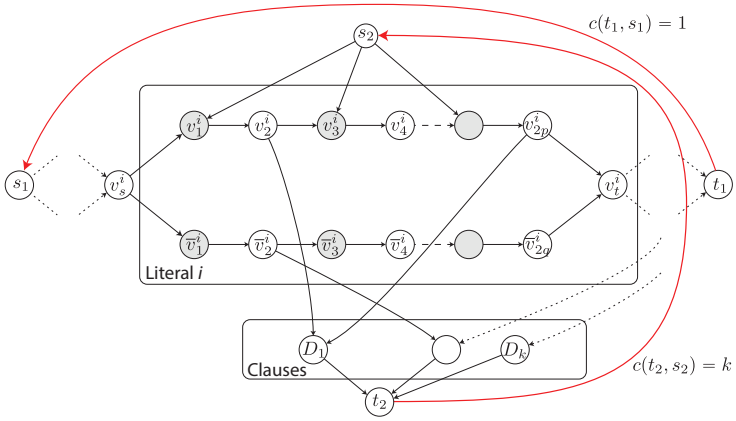

Figure 4.1: Graphical depiction of the proof of Theorem 4.1.2. A network flow of intensity k from s_1 to t_1 models clause satisfaction. A network flow of intensity 1 from s_2 to t_2 ensures that every literal can be true or false, but not both. We introduce two directed edges (shown in red) from t_1 to s_1 (with capacity k) and from t_2 to s_2 (with unit capacity).

Thus, any instance of SAT can be reduced to an instance of (the decision version of) two-customer i-CRRP. It follows that the decision version of i-CRRP is NP-hard. Since the i-CRRP problem can be verified in polynomial time, the decision version of i-CRRP is NP-complete. □

Any candidate solution to the i-CRRP can be verified in polynomial time: this proves that the i-CRRP is NP-complete.

We re-emphasize that the i-CRRP serves as a proxy for solving the general problem of minimizing travel times in the presence of congestion effects (without the complexity of dealing with sophisticated congestion models).

4.2 A Randomized Routing algorithm

Theorem 4.1.2 shows that the i-CRRP can not be solved efficiently for large problem instances unless P=NP. Furthermore, to the best of our knowledge, no polynomial-time approximation schemes are known for the integral Min-MCF problem (of which, as mentioned before, i-CRRP is an instance). Our strategy is then to focus on an approximate version of the i-CRRP, whereby capacity constraints can be slightly violated, that is,

Definition 4.2.1 (Approximate i-CRRP)**.** *Given a network model of an AMoD system, compute a set of routes that*

(i) transfers customers to their desired destinations (customer-carrying trips),

(ii) rebalances vehicles throughout the network to realign the vehicle fleet with the customers' demands (customer-empty, or rebalancing, trips),

(iii) with high probability, violates the capacity constraints on all road links by at most a small value δ,

(iv) has bounded suboptimality (in terms of customer travel times) under the analysis congestion model.

There exist randomized techniques for finding approximately optimal integral solutions for multi-commodity flow problems, e.g., randomized routing [Srinivasan, 1999, Raghavan and Tompson, 1987]. However, these require that each commodity has a single source and a single sink. Thus, they cannot be directly applied for the i-CRRP, since the rebalancing flow has multiple sources and sinks.

In Chapter 3, we show that the problems of routing passenger vehicles and routing rebalancing vehicles in the i-CRRP can be *decoupled* on *capacity*-symmetric networks (symmetric networks considered in this chapter fall under the class of capacity-symmetric networks). Specifically, for any given set of feasible customer routes, there exists a feasible set of rebalancing routes. Following this intuition, we exploit randomized techniques to find low-congestion routes for the customer-carrying vehicles. The customer routes may violate some of the congestion constraints: thus, we modify the road network so as to guarantee that a feasible rebalancing solution exists.

The procedure can be summarized as follows. First, a solution to the linear programing (LP) relaxation of the i-CRRP is computed, which produces a set of customers' network flows and a rebalancing network flow (Section 4.2.1). Second, each customer's network flow is "decomposed" into a collection of path flows (Section 4.2.2), which fulfill the capacity constraint (4.2d) but may violate the other constraints, namely (4.2b), (4.2c), (4.2e), and (4.2f). Third, a sampling procedure adjusts the path flow intensities to yield customers' path flows that satisfy Equations (4.2b), (4.2c), (4.2e), and (4.2f) but may (slightly) violate some of the capacity constraints (Section 4.2.3) – such path flows have intensity equal to 1 and are equivalent to customer routes (due to the equivalency between path flows of unit intensity and routes). Fourth, the residual capacity of edges in the road network are adjusted so as to guarantee that a feasible rebalancing flow exists; a rebalancing flow is then found by solving a linear program (Section 4.2.4). Finally, the rebalancing flow is decomposed into a collection of rebalancing flow with unit intensity, which are an equivalent representation for rebalancing routes (Section 4.2.5). In Section (4.2.6) we characterize the probability of violating the capacity constraints and the expected travel time of all customers under an empirical congestion model.

4.2.1 Step One: Linear Relaxation of the i-CRRP

The first step entails solving an LP relaxation of the i-CRRP, denoted as the *fractional* CRRP, whereby Equations 4.2e and 4.2f are replaced by

$$f_m(u,v) \in \mathbb{R}_{\geq 0} \qquad \forall m \in \mathcal{M}, (u,v) \in \mathcal{E}, \tag{4.3a}$$

$$f_{\mathrm{R}}(u,v) \in \mathbb{R}_{\geq 0} \qquad \forall (u,v) \in \mathcal{E}. \tag{4.3b}$$

Additionally, in order to reduce the size of the linear program, we *bundle* customer requests departing from the same origin into a single network flow. Specifically, we replace the set of customer requests leaving from a node s, $\{\{s_m, t_m, \lambda_m\}_m : s_m = s\}$, with a single customer request with origin s (with intensity $\sum_{m:s_m=s} \lambda_m$) and destinations $\{t_m\}_{m:s_m=s}$, each with intensity λ_m.

The resulting fractional CRRP is an instance of the fractional Min-MCF problem, which can be efficiently solved as a linear program of size $|\mathcal{E}|(S+1)$ (where S is the number of distinct origin nodes appearing in the customer requests) or via specialized combinatorial algorithms, e.g., [Goldberg et al., 1998].

4.2.2 Step Two: Flow Decomposition

The second step decomposes each customer network flow resulting from the relaxed LP into a collection of path flows, which can be later sampled and "adjusted" (in terms of their intensities) to ensure that constraints (4.2b) and (4.2e) are satisfied. Specifically, we utilize the decomposition algorithm in [Ahuja et al., 1993, Sec. 3.5] (also reported in Chapter 2 as Algorithm 1) – this algorithm decomposes general network flows into a collection of path flows and cycles with complexity $O(|V||\mathcal{E}|)$ [Ahuja et al., 1993], and its output is a collection of path flows whose sum equals the input network flow. Each path flow has a single origin and a single destination, corresponding to the origin and one of the destinations in the corresponding network flow. Thus, each path flow corresponds to one of the original customer requests. Importantly, in our case this decomposition step does *not* yield any cycles: if a cycle was present, then removing it would result in a new solution with lower cost.

4.2.3 Step Three: Path Sampling

Step Two yields, for every customer $m \in \mathcal{M}$, a set $\mathcal{F}_m$ of path flows (the cardinality of each set $\mathcal{F}_m$ is at most $|\mathcal{E}|$). Step Three entails sampling path flows from each set $\mathcal{F}_m$ to obtain a set of customer path flows that satisfy constraints (4.2b), (4.2c), (4.2e), and (4.2f). Specifically, we randomly and independently sample one path flow from each set $\mathcal{F}_m$, by using path flow intensities as probability distribution (the intensities represent a valid probability distribution since for each customer m, $\lambda_m = 1$ by assumption, and hence the path flow intensities must sum to one). We then set the intensity of the sampled path flow, denoted as $\{f^{\mathrm{s}}_m(u,v)\}_{(u,v)}$, equal to one (i.e., intensity$(\{f^{\mathrm{s}}_m\}) =$

1). By construction, the set of path flows $\{f^{\mathrm{s}}_m\}_m$ satisfy Equations (4.2b) and (4.2e). Explicitly, for each customer m, the path flow $\{f^{\mathrm{s}}_m\}$ is a network flow with origin s_m, destination t_m, and intensity 1: thus, it satisfies Equation (4.2b). Path flows of intensity 1 have integral flow on every edge: thus, the flows $\{f^{\mathrm{s}}_m\}_m$ satisfy Equation (4.2e).

In other words, the third step re-adjusts the intensities of the sampled path flows to ensure satisfaction of continuity constraints (4.2b) and integrality constraints (4.2e), at the expense of the capacity constraint (4.2d). Specifically, the expected (with respect to the randomization procedure) customer network flow crossing every edge $(u, v) \in \mathcal{E}$ is upper-bounded by the capacity of that edge, $c(u, v)$, that is:

$$\mathbb{E}\left[\sum_m f^{\mathrm{s}}_m(u,v)\right] = \sum_m f_m(u,v) \leq \sum_m f_m(u,v) + f_{\mathrm{R}}(u,v) \leq c(u,v),$$

where $\{f_m\}_m$ and $\{f_{\mathrm{R}}\}$ are the solutions from the LP relaxation in step one – the equality follows from the fact that path flows and the rebalancing solution are sampled with probability equal to their intensity. However, while the output of the algorithm satisfies the capacity constraints *in expectation*, a given realization may violate them.

4.2.4 Step Four: Computing a Rebalancing Flow

For a given collection of customer path flows, we adjust the residual capacity of the road network to ensure feasibility of the rebalancing flow. We define the *augmented capacity* of an edge as

$$\overline{c}(u,v) = \max\left(c(u,v), \sum_m f^s_m(u,v), \sum_m f^s_m(v,u)\right).$$

We then compute the *residual augmented capacity* of an edge as

$$\overline{c}_{\mathrm{R}}(u,v) = \overline{c}(u,v) - \sum_m f^s_m(u,v).$$

By construction, the sampled rebalancing flows are feasible solutions to the i-CRRP (and, in particular, they satisfy constraint (4.2d)) for a capacitated road network $\overline{G}(\mathcal{V}, \mathcal{E})$ with capacities $\{\overline{c}(u,v)\}_{(u,v)}$. Furthermore, $\overline{c}(u,v) = \overline{c}(v,u)$, so such road network is symmetric. Therefore, in accordance with 3.3.5, there exists a feasible rebalancing flow $\{f^s_{\mathrm{R}}(u,v)\}_{(u,v)}$ for the road network $\overline{G}(\mathcal{V}, \mathcal{E})$ with capacities $\{\overline{c}(u,v)\}_{(u,v)}$ and customer flows $\{f^s_m(u,v)\}_{m,(u,v)}$ (we use the superscript s to denote the dependency of $\{f^s_{\mathrm{R}}(u,v)\}_{(u,v)}$ on the sampled customer path flows $\{\{f^s_m(u,v)\}_{(u,v)}\}$).

Such a rebalancing flow can be found by solving

$$\begin{aligned}
\underset{f^s_{\mathrm{R}}(\cdot,\cdot)}{\text{minimize}} \quad & \sum_{(u,v)\in\mathcal{E}} t(u,v) f^s_{\mathrm{R}}(u,v) && && \text{(4.4a)}\\
\text{subject to} \quad & \sum_{u\in\mathcal{V}} f^s_{\mathrm{R}}(u,v) + \sum_{m\in\mathcal{M}} 1_{v=t_m}\lambda_m = \sum_{w\in\mathcal{V}} f^s_{\mathrm{R}}(v,w) + \sum_{m\in\mathcal{M}} 1_{v=s_m}\lambda_m && \forall v\in\mathcal{V} && \text{(4.4b)}\\
& f^s_{\mathrm{R}}(u,v) \le \bar{c}_{\mathrm{R}}(u,v) && \forall (u,v)\in\mathcal{E} && \text{(4.4c)}\\
& f^s_{\mathrm{R}}(u,v) \in \mathbb{N}_{\ge 0} && \forall (u,v)\in\mathcal{E}. && \text{(4.4d)}
\end{aligned}$$

Problem (4.4) is an instance of a single-commodity flow problem and all the source flows, sink flows and edge capacities are integral – as a result, the problem enjoys a totally unimodular structure and can be exactly and efficiently solved as a linear program by replacing constraint (4.4d) with

$$f^s_{\mathrm{R}}(u,v) \in \mathbb{R}_{\ge 0}, \quad \forall (u,v)\in\mathcal{E}. \tag{4.5}$$

4.2.5 Step Five: Flow Decomposition of the Rebalancing Network Flow

In the fifth step, we decompose the rebalancing network flow into a collection of path flows of unit intensity. In analogy with the customer path flow decomposition in Section 4.2.2, we decompose the rebalancing network flow using the flow decomposition algorithm. The output of the flow decomposition algorithm is a collection of path flows connecting every rebalancing origin (i.e., every customer destination) with a rebalancing destination (i.e., customer origin). Since the rebalancing network flow is integral, the flow decomposition algorithm returns *integral* path flows, i.e., path flows with integral intensity. As before, the decomposition of the rebalancing network flow does not yield any cycles.

4.2.6 Randomized Routing: Complexity and Performance

In this section, we characterize the complexity and performance of the randomized routing algorithm in terms of probabability of capacity constraint violations and approximation factor. Algorithm 2 summarizes the procedure detailed in the previous subsections for finding a solution to the approximate i-CRRP (function FlowDecomposition represents the flow decomposition algorithm [Ahuja et al., 1993, Sec. 3.5], reported in this thesis as Algorithm 1).

Complexity

Computational complexity is given by,

Theorem 4.2.2 (Complexity of randomized routing)**.** *The computational complexity of Algorithm 2 is polynomial in the size of the i-CRRP.*

Algorithm 2: Randomized routing algorithm

1: **Input**: An instance of the i-CRRP
2: **Output**: customer and rebalancing path flows, $\{f^{\mathrm{s}}_m\}$ and $\{f^{\mathrm{s}}_{\mathrm{RS}}\}$
3: $\{f_m\}_m, \{f_{\mathrm{R}}\} \leftarrow$ solve LP relaxation of the i-CRRP
4: **for all** $m \in \mathcal{M}$ **do**
5: $\quad \{\mathcal{F}_m\} \leftarrow$ FlowDecomposition($\{f_m\}$)
6: $\{f^{\mathrm{s}}_m\} \leftarrow$SampleCustomerPaths($\{\mathcal{F}_m\}_m$)
7: $\bar{c}_{\mathrm{R}}$ = ResidualAugmentedCapacity($\{f^s_m\}$)
8: $\{f^s_{\mathrm{R}}\} \leftarrow$ solve the LP relaxation of Problem 4.4
9: $\{f^{\mathrm{s}}_{\mathrm{RS}}\} \leftarrow$FlowDecomposition($\{f^s_{\mathrm{R}}\}$)

Proof. Any linear program can be solved in time polynomial in the size of its inputs [Karmarkar, 1984]. In particular, the LP relaxation of the i-CRRP in line 3 and the linear relaxation of Problem 4.4 in line 8 can be solved in polynomial time [Goldberg et al., 1998]. The computational complexity of the flow decomposition algorithm is $O(|V||\mathcal{E}|)$ [Ahuja et al., 1993]: the decomposition algorithm is called S times in lines 4-5 and once in line 9. The sampling procedure on line 6 is carried out $|\mathcal{M}|$ times and only involves trivial network flow manipulations. Finally, the residual augmented capacity of the network is computed with $|M||E|$ operations. Hence, the overall complexity is polynomial in the size of the i-CRRP. □

Performance

In this section we characterize the performance of the randomized routing algorithm (Algorithm 2). We first study the probability that a capacity constraint is violated (recall that Algorithm 2 ensures satisfaction of the capacity constraints only in *expectation*).

Theorem 4.2.3 (Violation of capacity constraints)**.** *Assume* $\max_{(u,v)\in\mathcal{E}} \sqrt{3\log(|\mathcal{E}|)/c(u,v)} \leq 1$ *and let* $\underline{\alpha}$ *be the unique* $\alpha \in (0, 1]$ *such that*

$$\max_{(u,v)\in\mathcal{E}} \sqrt{3\log(|\mathcal{E}|/\alpha)/c(u,v)} = 1.$$

Then, for any $\alpha \in [\underline{\alpha}, 1]$*, with probability* $1-\alpha$*, Algorithm 2 finds a solution to the approximate i-CRRP such that the capacity constraint for each link* (u, v) *is violated at most by a multiplicative factor* $(1 + \delta_{u,v})$*, where*

$$\delta_{u,v} := \sqrt{3\log(|\mathcal{E}|/\alpha)/c(u,v)}, \quad \textit{for all } (u,v).$$

Proof of Theorem 4.2.3. The proof relies on a Chernoff bound for every congestion constraint and on Boole's inequality.

Congestion on a single edge: We define the random variable $X(u,v)$ as the number of customer path flows selected by Algorithm 2 that cross any one edge (u,v). The mean of $X(u,v)$ is upper-bounded by the capacity of the edge $c(u,v)$. The Bernoulli trials are not independent: multiple flows from the same customer may cross edge (u,v), and exactly one is selected. However, the trials enjoy negative association [Dubhashi and Ranjan, 1996], therefore a Chernoff bound [Mitzenmacher and Upfal, 2005] holds:

$$\mathbb{P}\left(X(u,v) \geq (1+\delta)c(u,v)\right) \leq \exp(-c(u,v)\delta^2/3) \qquad \text{if } 0 < \delta \leq 1$$

Congestion on all edges: Select $\delta_{u,v}$ on every edge such that $c(u,v)\delta_{u,v}^2 = c(w,t)\delta_{w,t}^2$ for every pair of edges$(u,v),(w,t) \in \mathcal{E}$. Then, by Boole's inequality, the probability that at least one edge violates the congestion constraint is upper-bounded by

$$\mathbb{P}\left(\exists (u,v) \in \mathcal{E} : X(u,v) \geq (1+\delta_{u,v})c(u,v)\right) \leq |\mathcal{E}| \exp(-c(u,v)\delta_{u,v}^2/3)$$

Let us call $\alpha = |\mathcal{E}| \exp(-c(u,v)\delta_{u,v}^2/3)$. Solving for $\delta_{u,v}$ proves that all customer path flows satisfy the claim.

Rebalancing path flows The rebalancing network flow satisfies the property

$$f_{\mathrm{C}}^s(u,v) + f_{\mathrm{R}}^s(u,v) \leq \max\left(c(u,v), f_{\mathrm{C}}^s(u,v), f_{\mathrm{C}}^s(v,u)\right)$$

where f_{C}^s is defined in Equation (4.6). For any realization of the customer path flows, if $f_{\mathrm{C}}^s(u,v)$ and $f_{\mathrm{C}}^s(v,u)$ satisfy $f_{\mathrm{C}}^s(u,v) \leq (1+\delta_{u,v})c(u,v)$ and $f_{\mathrm{C}}^s(v,u) \leq (1+\delta_{v,u})c(v,u)$ respectively, then $f_{\mathrm{C}}^s(u,v) + f_{\mathrm{R}}^s(u,v) \leq (1+\delta_{v,u})c(u,v)$. This proves our claim. □

Remark A similar result can be found in the general case where $\delta_{u,v} \in \mathbb{R}_{>0}$ by exploiting a more general version of the Chernoff bound.

A few comments are in order. Clarify, Theorem 4.2.3 relies on the assumption that $\max_{(u,v)\in\mathcal{E}} \sqrt{3\,\log(|\mathcal{E}|)\,/c(u,v)} \leq 1$. This assumption is quite mild: for typical routing maps, $|\mathcal{E}| \sim 10,000$ and $c(u,v) \geq 100$, which leads to $\underline{\alpha} \sim 10^{-10}$. Second, when Theorem 4.2.3's assumption is met, the violation of a capacity constraint is at most 100% (since $\delta_{u,v} \leq 1$, always) with probability $1 - \underline{\alpha}$. Third, Theorem 4.2.3 relies on a Boole's inequality argument, which may lead to significant conservatism. A numerical characterization of the capacity constraint violations is performed in Section 4.3.

We now turn our attention to the approximation factor of the randomized routing algorithm. The Chernoff bound guarantees that any sampled solution yields a sampled customer traffic flow (and therefore a level of congestion) very close to the average with high probability. The rebalancing flow may add congestion: however, the overall vehicle flow on a link (u,v) is upper-bounded by the maximum between the capacity of the link, the customer traffic flow on link (u,v), and the customer

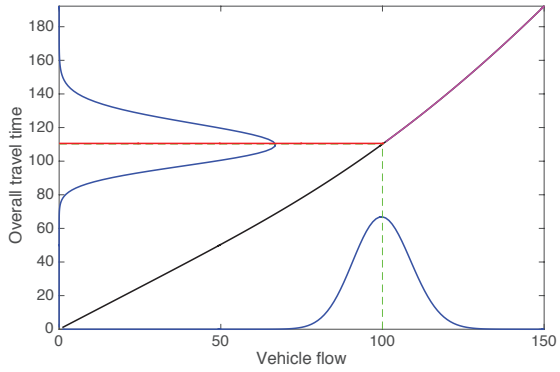

Figure 4.2: Overall customer travel time on a link: fractional solution (dashed green) and expected value of a sampled solution (red). The BPR link delay model is used.

traffic flow on link (v, u). As a result, the distribution of the customer travel time remains close to the distribution of the travel time experienced by customers on a road with infinite capacity.

Figure 4.2 gives a graphical depiction of this intuition for a link where only customer flows are present. We select $t_0 = 1$, $c = 100$, 500 path flows each crossing the link with probability 1/5. We use the BPR link delay model. In this example, the difference between the optimal fractional cost and the expected cost of a sampled solution according to the BPR model is 0.34%.

To formalize this intuition, we first provide two lemmas characterizing the distribution of the overall travel time and the per-vehicle travel time of customer-carrying vehicles on a link. We then provide a lemma characterizing the effect of the rebalancing flow on the overall travel time of customer-carrying vehicles. Finally, in Theorem 4.2.7 we provide a bound on the ratio of expected customer travel time between Algorithm 2 and the fractional CRRP solution.

Let $\{f^{\mathrm{s}}_{\mathrm{C}}(u,v)\}_{(u,v)}$ be the customer network flow induced by the output of Algorithm 2:

$$\{f^{\mathrm{s}}_{\mathrm{C}}(u,v)\}_{(u,v)} = \left\{ \sum_m f^{\mathrm{s}}_m(u,v) \right\}_{(u,v)}. \tag{4.6}$$

We denote the expectation of $f^s_{\mathrm{C}}(u,v)$ as $f_{\mathrm{C}}(u,v) \coloneqq \mathbb{E}\left[f^s_{\mathrm{C}}(u,v)\right]$. Note that $f_{\mathrm{C}}(u,v)$, is equal to the total flow $\sum_m f_m(u,v)$ of customer-carrying vehicles on link (u,v) induced by the solution to the LP relaxation of the i-CRRP (as discussed in Section 4.2.3).

Lemma 4.2.4 (Per-link approximation factor, no rebalancing)**.** *Consider a link $(u,v) \in \mathcal{E}$. Let $\tilde{t}_{(u,v)}(\cdot) \in \mathcal{T}$ be an analysis congestion model for link (u,v) and assume $\mathbb{E}\left[f^s_{\mathrm{C}}(u,v)\right] > 0$. Then, there exists a function $U : \mathcal{T} \times \mathbb{R} \mapsto \mathbb{R}_{\geq 0}$, which can be numerically evaluated, such that*

$$\frac{\mathbb{E}\left[\tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(u,v)) \cdot f^s_{\mathrm{C}}(u,v)\right]}{\tilde{t}_{(u,v)}(f_{\mathrm{C}}(u,v)) \cdot f_{\mathrm{C}}(u,v)} \leq U\Big(\tilde{t}_{(u,v)}(\cdot),\, f_{\mathrm{C}}(u,v)\Big).$$

Proof. Define the random variable

$$R := \frac{\tilde{t}_{(u,v)}(f_{\mathrm{C}}^{\mathrm{s}}(u,v)) \cdot f_{\mathrm{C}}^{\mathrm{s}}(u,v)}{\tilde{t}_{(u,v)}(f_{\mathrm{C}}(u,v)) \cdot f_{\mathrm{C}}(u,v)}.$$

The goal is to compute an upper bound on $\mathbb{E}[R]$.

The first step entails computing a Chernoff bound for the random variable $f_{\mathrm{C}}^{\mathrm{s}}(u,v)$. Since paths are sampled independently and flows belonging to the same customer request enjoy negative association, the following Chernoff bound holds [Dubhashi and Ranjan, 1996]:

$$\mathbb{P}\left(f_C^s(u,v) \leq (1+\delta) f_C(u,v)\right) \geq 1 - \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right)^{f_C(u,v)}$$

The bound can be rewritten in a more familiar form as a bound on the cumulative distribution function (CDF) of $f_C^s(u,v)$. Define the function $x \mapsto g(x)$:

$$g(x) = (x/f_{\mathrm{C}}(u,v) - 1),$$

Then we have that

$$\mathbb{P}\left(f_C^s(u,v) \leq x\right) \geq 1 - \left(\frac{e^{g(x)}}{(1+g(x))^{(1+g(x))}}\right)^{f_C(u,v)}$$

We then define the bijective function $y \mapsto q(y)$:

$$q(y) = \frac{\tilde{t}_{(u,v)}(y) \cdot y}{\tilde{t}_{(u,v)}(f_{\mathrm{C}}(u,v)) \cdot f_{\mathrm{C}}(u,v)},$$

whose inverse is denoted as $q^{-1}(\cdot)$. Note that $R = q(f_C^s(u,v))$. Since $q(\cdot)$ is bijective, the Chernoff bound for $f_{\mathrm{C}}^{\mathrm{s}}(u,v)$ can be transformed into a lower bound for the cumulative distribution function of R:

$$f_C^s(u,v) \leq x \quad \Leftrightarrow \quad R(f_C^s(u,v)) \leq R(x).$$

Then the following bound on the CDF of R holds:

$$\mathbb{P}\left(R(f_C^s(u,v)) \leq R(x)\right) \geq 1 - \left(\frac{e^{g(x)}}{(1+g(x))^{(1+g(x))}}\right)^{f_C(u,v)}. \tag{4.7}$$

The expected value of R can be expressed as a function of its cumulative distribution function CDF_R:

$$\mathbb{E}\left[R\right] = \int_{t=0}^{\infty} \left(1 - CDF_R(t)\right) dt \tag{4.8}$$

The CDF of R admits a lower bound:

$$\mathbb{P}\left(R \leq y\right) \geq 1 - \left(\frac{e^{g(q^{-1}(y))}}{(1+g(q^{-1}(y)))^{(1+g(q^{-1}(y)))}} \right)^{f_C(u,v)} \tag{4.9}$$

We now define the auxiliary random variable H with complementary cumulative distribution function:

$$\mathbb{P}\left(H \leq h\right) = 1 - \left(\frac{e^{g(q^{-1}(h))}}{(1+g(q^{-1}(h)))^{(1+g(q^{-1}(h)))}} \right)^{f_C(u,v)}.$$

Note that $\mathbb{E}[H]$ depends only on the analysis congestion model $\tilde{t}_{(u,v)}(\cdot)$ and the expected total flow.

The CDF of H is a lower bound on the CDF of R by Equation 4.9; therefore, $\mathbb{E}(H)$ is indeed an upper bound on $\mathbb{E}[R]$, according to Equation 4.8. Thus, $U\Big(\tilde{t}_{(u,v)}(\cdot),\, f_{\mathrm{C}}(u,v)\Big) = \mathbb{E}[H]$. This concludes the proof. □

A very similar procedure can be used to provide an upper bound on the travel time on a link.

Lemma 4.2.5 (Per-link travel time, no rebalancing)**.** *Consider a link $(u,v) \in \mathcal{E}$. Let $\tilde{t}_{(u,v)}(\cdot) \in \mathcal{T}$ be an analysis congestion model for link (u,v) and assume $\mathbb{E}\left[f^s_{\mathrm{C}}(u,v)\right] > 0$. Then, there exists a function $V : \mathcal{T} \times \mathbb{R} \mapsto \mathbb{R}_{\geq 0}$, which can be numerically evaluated, such that*

$$\frac{\mathbb{E}\left[\tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(u,v))\right]}{\tilde{t}_{(u,v)}(f_{\mathrm{C}}(u,v))} \leq V\Big(\tilde{t}_{(u,v)}(\cdot),\, f_{\mathrm{C}}(u,v)\Big).$$

Proof sketch The proof of Lemma 4.2.5 is identical to the proof of Lemma 4.2.4.

We are now in a position to characterize the effect of the rebalancing flow on the overall travel time of customer-carrying vehicles.

Lemma 4.2.6 (Per-link approximation factor)**.** *Consider a link $(u,v) \in \mathcal{E}$. Let $\tilde{t}_{(u,v)}(\cdot) \in \mathcal{T}$ be an analysis congestion model for link (u,v) and assume $\mathbb{E}\left[f^s_{\mathrm{C}}(u,v)\right] > 0$. Then, there exists a function $B : \mathcal{T} \times \mathbb{R} \mapsto \mathbb{R}_{\geq 0}$, which can be numerically evaluated, such that*

$$\frac{\mathbb{E}\left[\tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(u,v) + f^s_R(u,v)) \cdot f^s_{\mathrm{C}}(u,v)\right]}{\tilde{t}_{(u,v)}(f_{\mathrm{C}}(u,v)) \cdot f_{\mathrm{C}}(u,v)} \leq B\Big(\tilde{t}_{(u,v)}(\cdot),\, f_{\mathrm{C}}(u,v)\Big).$$

Proof. Three cases may arise.

- $f^s_{\mathrm{C}}(u,v) \leq c(u,v), f^s_{\mathrm{C}}(v,u) \leq c(v,u)$. The rebalancing flow on link (u,v) is $f^s_{\mathrm{R}}(u,v) \leq c(u,v) - f^s_{\mathrm{C}}(u,v)$. The resulting travel time is $\tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(u,v) + f^s_{\mathrm{R}}(u,v)) \leq \tilde{t}_{(u,v)}(c(u,v))$. Thus, for any realization of $f^s_{\mathrm{C}}(u,v)$, the increase in travel time admits an upper bound of $\Delta t_{\mathrm{R},(u,v)} \coloneqq \tilde{t}_{(u,v)}(c(u,v))/\tilde{t}_{(u,v)}(0)$ - as a result, in this case $B(\tilde{t}_{(u,v)}, f_{\mathrm{C}}(u,v)) \coloneqq \Delta t_{\mathrm{R},(u,v)} U(\tilde{t}_{(u,v)}, f_{\mathrm{C}}(u,v))$.

- $f^s_{\mathrm{C}}(u,v) \geq c(u,v), f^s_{\mathrm{C}}(u,c) \geq f^s_{\mathrm{C}}(v,u)$. The residual augmented capacity $\overline{c}_{\mathrm{R}}(u,v)$ is zero and the rebalancing flow is $f^s_{\mathrm{R}}(u,v) = 0$. The overall travel time of customer-carrying vehicles on link (u,v) is unchanged by the presence of the rebalancing flow: $B(\tilde{t}_{(u,v)}, f_{\mathrm{C}}(u,v)) \coloneqq U(\tilde{t}_{(u,v)}, f_{\mathrm{C}}(u,v))$.

- $f^s_{\mathrm{C}}(u,v) < f^s_{\mathrm{C}}(v,u), f^s_{\mathrm{C}}(v,u) \geq c(v,u)$. The residual capacity on link (u,v) is $\overline{c}_{\mathrm{R}}(u,v) = f^s_{\mathrm{C}}(v,u) - f^s_{\mathrm{C}}(u,v)$ and the overall flow on link (u,v) is $f^s_{\mathrm{C}}(u,v) + f^s_{\mathrm{R}}(u,v) \leq f^s_{\mathrm{C}}(v,u)$. The travel time is $\tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(u,v) + f^s_{\mathrm{R}}(u,v)) \leq \tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(v,u))$.

  The sampled customer flows $f^s_{\mathrm{C}}(u,v) = \sum_m f^s_m(u,v)$ and $f^s_{\mathrm{C}}(v,u) = \sum_m f^s_m(v,u)$ are independent. To see this, note that, for all customer flows $\in \mathcal{M}$ and for every edge $(u,v) \in \mathcal{E}$, either $f_m(u,v) = 0$ or $f_m(v,u) = 0$: if this was not the case, then subtracting $\min(f_m(u,v), f_m(v,u))$ from both $f_m(u,v)$ and $f_m(v,u)$ would result in a feasible solution of lower cost. Therefore, for every customer m, the path flow decomposition $\mathcal{F}_m$ contains path flows that may traverse either (u,v) or (v,u), but not both. Path flows belonging to different customers are sampled independently: as a result, $f^s_{\mathrm{C}}(u,v)$ and $f^s_{\mathrm{C}}(v,u)$ are independent.

  Then, for all realizations of $f^s_{\mathrm{C}}(u,v), f^s_{\mathrm{C}}(v,u)$ in this case,

$$\begin{aligned}\mathbb{E}\left[\tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(u,v) + f^s_{\mathrm{R}}(u,v)) \cdot f^s_{\mathrm{C}}(u,v)\right] &\leq \mathbb{E}\left[\tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(v,u)) \cdot f^s_{\mathrm{C}}(u,v)\right] \\ &= \mathbb{E}\left[\tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(v,u))\right] \mathbb{E}\left[f^s_{\mathrm{C}}(u,v)\right]\end{aligned}$$

  The upper bound in Lemma 4.2.5 can be used to bound $\mathbb{E}\left[\tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(v,u))\right]$:

$$\begin{aligned}\mathbb{E}\left[\tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(v,u))\right] \mathbb{E}\left[f^s_{\mathrm{C}}(u,v)\right] &\leq V(\tilde{t}_{(v,u)}(\cdot), f_{\mathrm{C}}(v,u))\tilde{t}_{(v,u)}(f_{\mathrm{C}}(v,u)) f_{\mathrm{C}}(u,v) \\ &\leq V(\tilde{t}_{(v,u)}(\cdot), f_{\mathrm{C}}(v,u))\tilde{t}_{(v,u)}(c(v,u)) f_{\mathrm{C}}(u,v) \\ &\leq V(\tilde{t}_{(v,u)}(\cdot), f_{\mathrm{C}}(v,u))\Delta t_{\mathrm{R},(u,v)}\tilde{t}_{(v,u)}(f_{\mathrm{C}}(u,v)) f_{\mathrm{C}}(u,v)\end{aligned}$$

  In this case,

$$B(\tilde{t}_{(u,v)}, f_{\mathrm{C}}(u,v)) \coloneqq V(\tilde{t}_{(v,u)}(\cdot), f_{\mathrm{C}}(v,u))\Delta t_{\mathrm{R},(u,v)}$$

In conclusion,

$$B(\tilde{t}_{(u,v)}, f_{\mathrm{C}}(u,v)) \coloneqq \max\left(\Delta t_{\mathrm{R},(u,v)} U(\tilde{t}_{(u,v)}, f_{\mathrm{C}}(u,v)), V(\tilde{t}_{(v,u)}(\cdot), f_{\mathrm{C}}(v,u))\Delta t_{\mathrm{R},(u,v)}\right)$$

□

Lemma 4.2.6 shows that the ratio between the expected overall travel time of customer-carrying vehicles on link (u,v) under Algorithm 2 (i.e., $\mathbb{E}\left[\tilde{t}_{(u,v)}(f^s_{\mathrm{C}}(u,v)) \cdot f^s_{\mathrm{C}}(u,v)\right]$) and the overall travel time on link (u,v) under the LP relaxation of the i-CRRP (i.e., $\tilde{t}_{(u,v)}(f_{\mathrm{C}}(u,v)) \cdot f_{\mathrm{C}}(u,v)$) is upper bounded by a function that depends only on the analysis congestion model $\tilde{t}_{(u,v)}(\cdot)$ and the expected

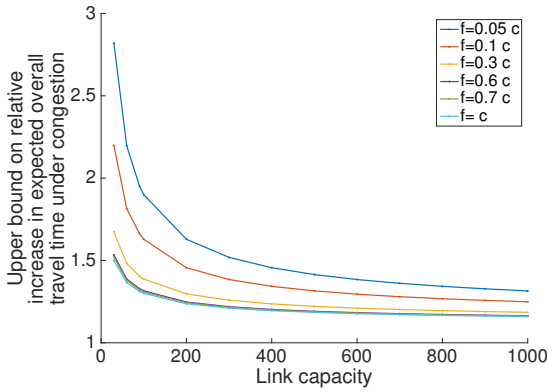

Figure 4.3: Upper bound B on the fractional increase in expected value of the overall travel time of customer-carrying vehicles on a link as a function of link flow and link capacity. The BPR link delay model is used.

total flow. Such a function can be numerically evaluated by computing $\mathbb{E}[H]$ (as defined in the proof sketch of Lemma 4.2.4), given $\tilde{t}_{(u,v)}(\cdot)$ and $f_C(u,v)$. As an example, consider the Bureau of Public Roads (BPR) delay model presented in Section 4.1.1. Figure 4.3 shows the bound B as a function of the capacity parameter (i.e., c in Equation (4.1)) and the total traffic flow. The bound is quite tight, especially when the total traffic flow is close to link capacity – thus the routes computed with Algorithm 2 appear to induce travel times on the road relatively close to the optimal congestion-free number.

The per-link bound in Lemma 4.2.6 can be easily extended to a system-wide bound as follows.

Theorem 4.2.7 (System-wide approximation factor)**.** *Let $\tilde{t}_{(u,v)}(\cdot) \in \mathcal{T}$ be an analysis congestion model for each link $(u,v) \in \mathcal{E}$. Then, the ratio between the expected number of vehicles on the road under Algorithm 2 and the number of vehicles on the road under the LP relaxation of the i-CRRP is upper bounded by:*

$$\frac{\sum_{(u,v)\in\mathcal{E}} B(\tilde{t}_{(u,v)}(\cdot),\, f_C(u,v))\, \tilde{t}_{(u,v)}(f_C(u,v)) \cdot f_C(u,v)}{\sum_{(u,v)\in\mathcal{E}} \tilde{t}_{(u,v)}(f_C(u,v)) \cdot f_C(u,v)}. \tag{4.10}$$

Proof. The proof of this statement follows trivially from the linearity of expectation and Lemma 4.2.6. □

4.3 Numerical Experiments

4.3.1 Performance of the randomized routing and rebalancing algorithm

We explore the performance of the randomized routing procedure described in Algorithm 2 on a real-world, asymmetric road network with real customer demands. We consider a road network of Manhattan with 3137 roads and 1351 intersections, derived from OpenStreetMap data and shown in Fig. 3.3. Customer requests are derived from 55412 actual taxi rides in Manhattan on March 1,

2012 from 6 to 8 p.m [1].

We adjust the capacities of the roads such that, on average, the flows induced by these trips are close to the onset of congestion. In order to guarantee feasibility of the LP relaxation of the i-CRRP, we relax the congestion constraints by introducing slack variables, each associated with a large cost. Since the road network is not symmetric (and, in particular, some roads may be one-way streets), it is not possible to compute augmented road capacities. To circumvent this, we associate slack variables to the congestion constraints in the LP relaxation of Problem 4.4: the cost associated with each slack variable is proportional to the effect that an increase in congestion would have on the overall customer travel time. Intuitively, the algorithm is allowed to select the *minimal* relaxation of the congestion constraints that guarantees feasibility of the rebalancing problem. We solve the i-CRRP with Algorithm 2 and compare the overall travel time of customer-carrying vehicles of the solution (computed with the BPR link delay function) with the *optimal* congestion-free overall travel time of customer-carrying vehicles (computed with the CPLEX MILP solver). In the problem instance considered in this experiment, the optimal travel time of customer-carrying vehicles coincide with the optimal travel time under the LP relaxation. However, we stress that the linear relaxation does not yield integral routes: thus, it is not suitable for real-time control of an AMoD system.

We consider 100 realizations of the randomized rounding algorithm. Table 4.1 summarizes our results. Figure 4.4 shows the distribution of the ratio between the required number of vehicles of the sampled solution and the required number of vehicles of the LP solution.

	LP	Rand. routing
Avg. cust. travel time [s]	91.552	91.716
Avg. num. congested edges	233	333.7
Avg. cong. (as a fraction of link capacity) on congested edges	36.5%	25.6%

Table 4.1: Randomized congestion-aware routing: results of the numerical simulations. The performance of the congestion-aware randomized routing algorithm is very close to the lower bound on performance provided by the solution to the LP relaxation.

Interestingly, for some simulations with low congestion (not shown for brevity), we observed that the overall customer travel time required by the sampled solution is sometimes smaller than the customer travel time in the LP relaxation. This counterintuitive result is due to the fact that the LP relaxation computes a congestion-free solution, even if this results in longer paths for the customers. The randomized routing algorithm, on the other hand, sometimes samples shorter paths that induce a small amount of congestion but, overall, result in smaller travel times.

The average execution time of Algorithm 2 on a commodity PC is 655 s; this time is dominated by the time required to solve Problem 4.2. In contrast, the execution time of the MILP solver is 950 s. Thus, the algorithm finds a high-quality solution 45% faster than than the exact solver.

[1] Courtesy of the New York Taxi and Limousine Commission.

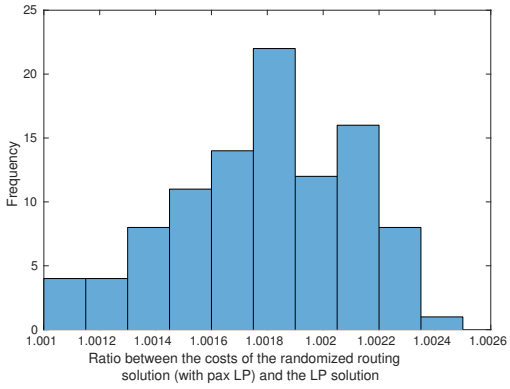

Figure 4.4: Distribution of the ratio between the overall customer travel time of the randomized routing solution and the overall travel time of the LP.

Algorithm 2 is not, in general, guaranteed to be faster than the MILP solver (indeed we observed that, for especially simple instances of the i-CRRP, performance of the MILP solver is comparable to our algorithm). Nevertheless, for large-scale problems, the randomized routing algorithm guarantees that the solution time will *always* be polynomial in the problem size (an especially relevant concern for real-time control) and the value of the solution will be close to the optimal one.

4.3.2 Performance of a receding-horizon implementation

Algorithm 2 holds promise as a real-time algorithm for the control of fleets of AMoD vehicles. While a receding-horizon implementation is beyond the scope of this work, we compare the performance of Algorithm 2 with the performance of the receding-horizon rebalancing algorithm proposed in Chapter 3 for a single problem instance. This instance can be seen as proxy for a single step of a receding-horizon algorithm. We consider the same New York City road network discussed in Section 4.3.1; we randomly sample 1000 customer arrivals (approximately corresponding to the number of customers arriving in two minutes in the original data set) and scale road capacity accordingly. The receding-horizon algorithm in Chapter 3 only computes rebalancing routes; customer-carrying vehicles are greedily routed along the fastest path.

We considered fifty realizations of the customer requests: on average, the overall travel time of customer-carrying vehicles was $5.36 \pm 0.62\%$ lower with Algorithm 2. The computation time was $28.6 \pm 5.7s$ on commodity hardware. Thus, the a receding-horizon implementation of the algorithm holds promise for real-time control of city-scale AMoD systems.

4.4 Conclusions and Future Work

In this chapter we presented a randomized algorithm for simultaneously computing customer routes and rebalancing routes on a capacitated road network in an autonomous mobility-on-demand system. Our goal is to find a set of routes that minimize the total travel time of customer-carrying vehicles:

since the problem is intractable for generic congestion models, we adopt a simple threshold congestion model that yields congestion-free routes. We formulate the routing and rebalancing problem as an integral Minimum-Cost Multi-Commodity Flow problem and, after showing that even this simple problem is NP-hard, we develop a sampling technique that extends known randomized routing algorithms to flows with multiple origins and destinations. The sampling technique may violate some of the congestion constraints: nevertheless, we prove that the expected overall travel time of customer-carrying vehicles produced by the randomized algorithm under a realistic congestion model remains very close to the optimal congestion-free travel time (a proxy for the optimal travel time). Numerical results on a realistic Manhattan road network show that the travel time of customers required by our algorithm is very close to the optimal travel time required by a congestion-free solution. Preliminary experiments show that a receding-horizon implementation of the algorithm yields promising performance when compared with state-of-the-art rebalancing algorithms.

This work paves the way for the development of large-scale congestion-aware routing and rebalancing algorithms for AMoD systems. Future work will explore a receding-horizon, closed-loop implementation of the algorithm and integration with state-of-the-art traffic simulators such as MATSIM and SUMO and characterize its performance in presence of stochastic fluctuations in the demand, travel times, and routing distribution. Additionally we would like to study other ways of reducing congestion such as staggering demands, ride-sharing, and integration with public transportation.

Chapter 5

On the interaction between Autonomous Mobility-on-Demand systems and the power network: models and coordination algorithms

In this chapter, we study the interaction between an AMoD fleet and the electric power network. We propose a *joint* model that captures the coupling between the two systems stemming from the vehicles' charging requirements. The model subsumes existing network flow models for AMoD systems and DC models for the power network, and it captures time-varying customer demand and power generation costs, road congestion, battery depreciation, and power transmission and distribution constraints. We then leverage the model to jointly optimize the operation of both systems. We devise an algorithmic procedure to losslessly reduce the problem size by bundling customer requests, allowing it to be efficiently solved by off-the-shelf linear programming solvers. Next, we show that the socially optimal solution to the coordination problem represents a general equilibrium, and we provide a dual decomposition algorithm that allows the transportation and power network operators to compute such equilibrium without sharing private information. We assess the performance of the model and algorithms by studying the implementation of a hypothetical electric-powered AMoD system in Dallas-Fort Worth, and its impact on the Texas power network. We show that coordination between the AMoD system and the power network can *reduce* the overall energy expenditure compared to the case where no cars are present (despite the increased demand

for electricity) and yield savings of $182M/year compared to an uncoordinated scenario. Finally, we provide a closed-loop receding-horizon implementation. Collectively, the results of this chapter provide a first-of-a-kind characterization of the interaction between electric-powered AMoD systems and the power network, and shed additional light on the economic and societal value of AMoD.

5.1 Introduction

Private vehicles are major contributors to urban pollution [OECD, 2014], which is estimated to cause over seven million premature deaths worldwide every year [World Health Organization, 2014]. Plug-in electric vehicles (EVs) hold promise to significantly reduce urban pollution, both by reducing carbon dioxide emissions from internal-combustion engine vehicles, and by enabling use of renewable and low-polluting power generators as a source of energy for transportation services. However, at present, adoption of EVs for private mobility has been significantly hampered by customers' concerns about limited range and availability of charging infrastructure [Evarts, 2013].

The emerging technology of self-driving vehicles might provide a solution to these challenges and thus might represent a key enabler for the widespread adoption of EVs. Specifically, fleets of self-driving vehicles providing on-demand transportation services (referred to as Autonomous Mobility-on-Demand, or AMoD, systems) hold promise to replace personal transportation in large cities by offering high quality of service at lower cost [Spieser et al., 2014] with positive effects on safety, parking infrastructure, and congestion. Crucially, EVs are especially well-suited to AMoD systems. On the one hand, short-range trips typical of urban mobility are well-suited to the current generation of range-limited EVs; on the other hand, intelligent fleet-wide policies for rebalancing and charging can ensure that vehicles with an adequate level of charge are available to customers, virtually eliminating "range anxiety," a major barrier to EV adoption [Evarts, 2013]. To fully realize this vision, however, one needs currently unavailable tools to manage the complex *couplings* between AMoD fleet management (e.g., for routing and charging the EVs) and the control of the power network. Specifically, one should consider

1. *Impact of transportation network on power network*: Concurrent charging of large numbers of EVs can have significant effects both on the stability of the power network and on the local price of electricity (including at the charging stations) [Sioshansi, 2012, Alizadeh et al., 2017, Hadley and Tsvetkova, 2009]. For example, [Hadley and Tsvetkova, 2009] shows that in California a 25% market penetration of (non-autonomous) EVs with fast chargers, in the absence of smart charging algorithms, would increase overall electricity demand in peak load by about 30%, and electricity prices by almost 200%.

2. *Impact of power network on transportation network*: Electricity prices can significantly affect travel patterns for EVs. In [Alizadeh et al., 2017], the authors show that changes in electricity

prices can radically alter the travel patterns and charging schedules of fleets of EVs in a simplified model of the San Francisco Bay Area. This, in turn, would affect electricity prices in a complex feedback loop.

The key idea behind this work is that, by intelligently routing fleets of autonomous EVs and, in particular, by harnessing the flexibility offered by the routes and schedules for the empty-traveling vehicles, one can *actively* control such complex couplings and guarantee high-performance for the overall system (e.g., high passenger throughput and lower electricity costs). Additionally, autonomous EVs provide a unique opportunity for joint traffic and energy production management, as they could act as mobile storage devices. That is, when not used for the fulfillment of trip requests, the vehicles could be routed to target charging stations in order to either absorb excess generated energy at time of low power demand (by charging) or inject power in the power network at times of high demand (by discharging).

Literature review: The integration of *non-autonomous* EVs within the power network has been addressed in three main lines of work. A first line of work addresses the problem of scheduling charging of EVs (i.e., optimizing the charging profile in *time*) under the assumption that the vehicles' charging schedule has no appreciable effect on the power network [Rotering and Ilic, 2011, Turitsyn et al., 2010, Tushar et al., 2012]. This assumption is also commonly made when selecting the locations of charging stations (i.e., optimizing the charging profile in *space*) [Goeke and Schneider, 2015, Pourazarm et al., 2016]. A high penetration of EVs would, however, significantly affect the power network. Thus, a second line of work investigates the effects of widespread adoption of EVs on key aspects such as wholesale prices and reserve margins, for example in macroeconomic [Hadley and Tsvetkova, 2009] and game-theoretical [Sioshansi, 2012], [Wang et al., 2010] settings. Accordingly, [Alizadeh et al., 2014, Alizadeh et al., 2017] investigate joint models for EV routing and power generation/distribution aimed at driving the system toward a socially-optimal solution. Finally, a third line of work investigates the potential of using EVs to regulate the power network and satisfy short-term spikes in power demand. The macroeconomic impact of such schemes (generally referred to as Vehicle-To-Grid, or V2G) has been studied in [Kempton and Tomić, 2005], where it is shown that widespread adoption of EVs and V2G technologies could foster significantly increased adoption of wind power. Going one step further, [Khodayar et al., 2013] proposes a unified model for EV fleets and the power network, and derives a joint dispatching and routing strategy that maximizes social welfare (i.e., it minimizes the *overall* cost borne by all participants, as opposed to maximizing individual payoffs). However, [Kempton and Tomić, 2005] does not capture the *spatial* component of the power and transportation networks, while [Khodayar et al., 2013] assumes that the vehicles' schedules are fixed.

The objective of this chapter is to investigate the interaction between AMoD and electric power systems (jointly referred to as Power-in-the-loop AMoD, or P-AMoD, systems) in terms of modeling and algorithmic tools to effectively manage their couplings (Figure 5.1). In this context, our work

improves upon the state of the art (in particular, [Alizadeh et al., 2014, Alizadeh et al., 2017]) along three main dimensions: (i) it considers a fleet of *shared* and *autonomous* EVs, which offer significant additional degrees of freedom for vehicle scheduling, routing, and charging; (ii) it provides efficient algorithms that can solve large-scale problems; and (iii) it characterizes the vehicles' ability to return power to the power network through V2G schemes, and its economic benefits.

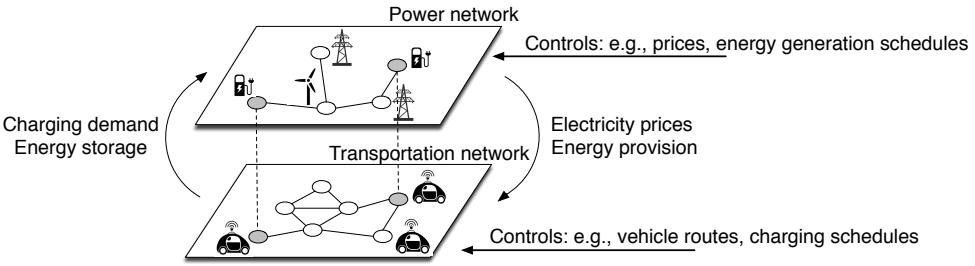

Figure 5.1: Couplings between AMoD and electric power systems. The system-level control of Power-in-the-loop AMoD systems entails the *coordinated* selection of routes for the autonomous vehicles, charging schedules, electricity prices, and energy generation schedules, among others.

Statement of contributions: Specifically, the contribution of this work is fivefold. First, we propose a joint *linear* model for P-AMoD systems. The model captures time-varying customer demand and generation prices, congestion in the road network, vehicle battery depreciation, power transmission constraints on the transmission lines, and transformer capacity constraints induced by the distribution network. Second, we leverage the model to design tools that optimize the operations of P-AMoD systems and, in particular, maximize social welfare. To this end, we propose an algorithmic procedure to losslessly reduce the dimensionality of the P-AMoD model. The procedure allows P-AMoD problems with hundreds of road links, time horizons of multiple hours, and any number of customers and vehicles to be optimized on commodity hardware. Third, we show that the socially-optimal solution to the P-AMoD problem is a general equilibrium, and we propose a distributed privacy-preserving algorithm that the transportation and power network operators can employ to compute the equilibrium without disclosing their private information.

Fourth, we apply the model and algorithms to a case study of a hypothetical deployment of an AMoD system in Dallas-Fort Worth, TX. We show that coordination between the AMoD system and the electric power network can have a significant positive impact on the price of electricity (remarkably, the overall electricity expenditure in presence of the AMoD system can be *lower* than in the case where no vehicles are present, despite the increased demand), while retaining *all* the convenience and sustainability benefits of AMoD. This suggests that the societal value of AMoD systems spans beyond mobility: properly coordinated, AMoD systems can deliver significant benefits to the wider community by helping increase the efficiency of the power network. Finally, we present a receding-horizon algorithm for P-AMoD that provides built-in closed-loop robustness and delivers computation times in the order of seconds on commodity hardware at the price of some suboptimality.

Organization: The remainder of this chapter is organized as follows. In the *Model Description and Problem Formulation* section we present a linear model that captures the interaction between an AMoD system and the power network. In the section on *Solution Algorithms*, we propose a procedure to losslessly reduce the size of the linear model by bundling customer requests. In the *Distributed solution to the P-AMoD problem* section, we show that the socially optimal solution to the P-AMoD problem is a general equilibrium and propose a privacy-preserving distributed optimization algorithm. In the *Numerical Experiments* section, we evaluate our model and algorithm on a case study of Dallas-Fort Worth. In the section on *A Receding-Horizon Algorithm for P-AMoD*, we propose a receding-horizon implementation. Finally, in the *Conclusions*, we draw conclusions and discuss directions for future work.

5.2 Model Description and Problem Formulation

We propose a linear, flow-based model that captures the interaction between an AMoD system and the power network. The model consists of two parts.

First, we extend the model in [Rossi et al., 2018] to a time-varying network flow model of an AMoD system with EVs. We assume that a Transportation Service Operator (TSO) manages the AMoD system in order to fulfill passenger trip requests within a given road network. The road links are subject to congestion, and the trip requests arrive according to an *exogenous* dynamical process. The TSO must not only compute the routes for the autonomous EVs (i.e. *vehicle routing*), but also issue tasks and routes for empty vehicles in order to, for example, realign the fleet with the asymmetric distribution of trip demand (i.e. *vehicle rebalancing*). Due to limited battery capacity, the vehicles need to periodically charge at charging stations. The price of electricity varies between charging stations – the charging schedule is determined by the TSO in order to minimize the fleet's operational cost.

The price of electricity itself is a result of the power system operation to balance supply and demand, and varies across the power grid. Thus, we next review the linear (DC) power flow model of the power network and the economic dispatch problem used to calculate market clearing prices for electricity. The power transmission network comprises energy providers that are connected to load buses through high-voltage transmission lines. Transmission capacities (dictated chiefly by thermal considerations) limit the amount of power that can be transferred on each transmission line. Load buses are connected to charging stations and other sources of power demand through the distribution systems: this system induces constraints on the amount of power that can be served to each load bus. Power demands other than those from charging stations are regarded as exogenous parameters in this work. The power network is controlled by an Independent System Operator (ISO). The ISO also determines prices at the load buses (and, consequently, at the charging stations) so as to guarantee grid reliability while minimizing the overall generation cost (a problem known as *economic*

dispatch).

The vehicles' charging introduces a critical coupling between the transportation and the power networks. The power demands due to charging influence the local price of electricity set by the ISO – the prices, in turn, affect the optimal charging schedule computed by the TSO. Accordingly, we conclude this section by describing the interaction between the two models, and we propose a joint model for Power-in-the-loop AMoD.

5.2.1 Network Flow Model of an AMoD system

We consider a time-varying, finite-horizon model. The time horizon of the problem is discretized in T time intervals; the battery charge level of the autonomous vehicles is similarly discretized in C charge levels, each corresponding to an amount of energy denoted as J_C.

Road network: The road network is modeled as a directed graph $R = (\mathcal{V}_R, \mathcal{E}_R)$, where $\mathcal{V}_R$ denotes the node set and $\mathcal{E}_R \subseteq \mathcal{V}_R \times \mathcal{V}_R$ denotes the edge set. Nodes $v \in \mathcal{V}_R$ denote either an intersection, a charging station, or a trip origin/destination. Edges $(v, w) \in \mathcal{E}_R$ denote the availability of a road link connecting nodes v and w. Each edge (v, w) has an associated length $d_{(v,w)} \in \mathbb{R}_{\geq 0}$, traversal time $t_{(v,w)} \in \{1, \ldots, T\}$, energy requirement $c_{(v,w)} \in \{-C, \ldots, C\}$, and traffic capacity $\overline{f}_{v,w} \in \mathbb{R}_{\geq 0}$. The length $d_{(v,w)}$ determines the mileage driven along the road link; the traversal time $t_{(v,w)}$ characterizes the travel time on the road link in absence of congestion; the energy requirement $c_{(v,w)}$ models the energy consumption (i.e., the number of charge levels) required to traverse the link in absence of congestion; the capacity $\overline{f}_{v,w}$ captures the maximum vehicle flow rate (i.e., the number of vehicles per unit of time) that the road link can accommodate without experiencing congestion.

Vehicles traversing the road network can recharge and discharge their batteries at charging stations, whose locations are modeled as a set of nodes $\mathcal{S} \subset \mathcal{V}_R$. Each charging station $s \in \mathcal{S}$ is characterized by a charging rate $\delta c_s^+ \in \{1, \ldots, C\}$, a discharging rate $\delta c_s^- \in \{-C, \ldots, -1\}$, a time-varying charging price $p_s^+(t) \in \mathbb{R}$, a time-varying discharging price $p_s^-(t) \in \mathbb{R}$, and a capacity $\overline{S}_s \in \mathbb{N}$. The charging and discharging rates δc_s^+ and δc_s^- correspond to the amount of energy (in unit charge levels) that the charger can provide to a vehicle (or, conversely, that a vehicle can return to the power grid) in one unit of time. For simplicity, we assume that the charging rates are fixed; however, the model can be extended to accommodate variable charging rates. The charging and discharging prices $p_s^+(t)$ and $p_s^-(t)$ capture the cost of one discrete energy level energy (or, conversely, the payment the vehicles receive for returning one unit of energy to the grid) at time t; in this work, $p_s^+(t) = p_s^-(t)$. The capacity $\overline{S}_s$ models the maximum number of vehicles that can simultaneously charge or discharge at station s. Charging and discharging (due both to driving activity and to vehicle-to-grid power injection) causes wear in the vehicles' batteries. The battery depreciation per unit charge or discharge is denoted as d_B. Battery depreciation captures the cost of replacing a battery at the end of its useful life; note, however, that the vehicle's battery capacity is assumed to remain constant during the model's finite horizon.

Expanded AMoD network: We are now in a position to rigorously define the network flow model for the AMoD system. We introduce an *expanded* AMoD network modeled as a directed graph $G = (\mathcal{V}, \mathcal{E})$. The graph G captures the time-varying nature of the problem and tracks the battery charge level of the autonomous vehicles. Specifically, nodes $\mathbf{v} \in \mathcal{V}$ model physical locations at a given time and charge level, while edges $e \in \mathcal{E}$ model road links and charging actions at a given time and charge level. Formally, a node $\mathbf{v} \in \mathcal{V}$ corresponds to a tuple $\mathbf{v} = (v_\mathbf{v}, t_\mathbf{v}, c_\mathbf{v})$, where $v_\mathbf{v} \in \mathcal{V}_R$ is a node in the road network graph R; $t_\mathbf{v} \in \{1, \dots, T\}$ is a discrete time; and $c_\mathbf{v} \in \{1, \dots, C\}$ is a discrete charge level. The edge set $\mathcal{E}$ is partitioned into two subsets, namely $\mathcal{E}_L$ and $\mathcal{E}_S$, such that $\mathcal{E}_L \cup \mathcal{E}_S = \mathcal{E}$ and $\mathcal{E}_L \cap \mathcal{E}_S = \emptyset$. Edges $e \in \mathcal{E}_L$ represent road links, whereas edges $e \in \mathcal{E}_S$ model the charging/discharging process at the stations. An edge $(\mathbf{v}, \mathbf{w})$ belongs to $\mathcal{E}_L$ when (i) an edge $(v_\mathbf{v}, v_\mathbf{w})$ exists in the road network graph edge set $\mathcal{E}_R$, (ii) the link $(v_\mathbf{v}, v_\mathbf{w}) \in \mathcal{E}_R$ can be traversed in time $t_\mathbf{w} - t_\mathbf{v} = t_{(v_\mathbf{v}, v_\mathbf{w})}$, and (iii) the battery charge required to traverse the link is $c_\mathbf{v} - c_\mathbf{w} = c_{(v_\mathbf{v}, v_\mathbf{w})}$. Conversely, an edge $(\mathbf{v}, \mathbf{w})$ represents a charging/discharging edge in $\mathcal{E}_S$ when (i) $v_\mathbf{v} = v_\mathbf{w}$ is the location of a charging station in $\mathcal{S}$ and (ii) the charging/discharging rate at the charging location $v_\mathbf{v}$ is $(c_\mathbf{w} - c_\mathbf{v})/(t_\mathbf{w} - t_\mathbf{v}) = \delta c^+_{v_\mathbf{v}}$ (charging) or $(c_\mathbf{w} - c_\mathbf{v})/(t_\mathbf{w} - t_\mathbf{v}) = \delta c^-_{v_\mathbf{v}}$ (discharging). Figure 5.2 (left) shows a graphical depiction of the graph G.

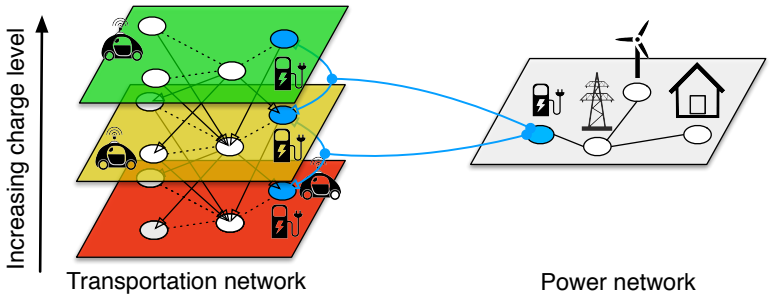

Figure 5.2: Augmented transportation and power networks. Nodes in the augmented transportation network (left) represent a location along with a given charge level (each layer of the augmented transportation network corresponds to a charge level). Dashed lines denote roads in the original transportation network and are not part of the augmented network. As vehicles travel on road links (modeled by black arrows in the augmented network), their charge level decreases. Blue nodes represent charging stations: the flows on charging and discharging edges affect the load at the corresponding nodes in the power network. For simplicity, only one time step is shown.

Customer and rebalancing routes: Transportation requests are represented by the set of tuples $\{(v_m, w_m, t_m, \lambda_m)\}_{m=1}^M$, where $v_m \in \mathcal{V}_R$ is the request's origin location, $w_m \in \mathcal{V}_R$ is the request's destination location, t_m is the requested pickup time, and λ_m is the average customer arrival rate (or simply customer rate) of request m at time interval t_m. Transportation requests are assumed to be known and deterministic.

The goal of the TSO is to compute a routing and recharging policy for the self-driving vehicles. To achieve this, we model vehicle routes as network flows [Ahuja et al., 1993]. Network flows are

an *equivalent representation* for routes. Indeed, any route can be represented as a network flow assuming value 1 on edges belonging to the route and 0 elsewhere; conversely, all network flows considered in this work can be represented as a collection of weighed routes [Ahuja et al., 1993, Ch. 3]. This representation allows us to leverage the rich theory of network flow: in particular, in the *Solution Algorithms* section we exploit this theory to *losslessly reduce* the dimensionality of the optimization problem.

We denote the *customer flow* as the rate of customer-carrying vehicles belonging to a specific transportation request $(v_m, w_m, t_m, \lambda_m)$ traversing an edge $e \in \mathcal{E}$. Formally, for request $m \in \{1, \dots, M\}$, the customer flow is a function $f_m(\mathbf{v}, \mathbf{w}) : \mathcal{E} \mapsto \mathbb{R}_{\geq 0}$, that represents the rate of customers belonging to request m traveling from location $v_{\mathbf{v}}$ to location $v_{\mathbf{w}}$ (or charging/discharging at location $v_{\mathbf{v}} = v_{\mathbf{w}}$) from time $t_{\mathbf{v}}$ to time $t_{\mathbf{w}}$, with an initial battery charge of $c_{\mathbf{v}}$ and a final battery charge of $c_{\mathbf{w}}$. Analogously, the rebalancing (or customer-empty) flow $f_0(\mathbf{v}, \mathbf{w}) : \mathcal{E} \mapsto \mathbb{R}_{\geq 0}$ represents the rate of empty vehicles traversing a road link or charging/discharging[1]. Customer flows must satisfy a *continuity* condition: customer-carrying vehicles entering a node at a given time and charge level must exit the same node at the same time and with the same charge level. Equation (5.1) enforces this condition:

$$\sum_{\mathbf{u}:(\mathbf{u},\mathbf{v})\in\mathcal{E}} f_m(\mathbf{u},\mathbf{v}) + 1_{v_{\mathbf{v}}=v_m} 1_{t_{\mathbf{v}}=t_m} \lambda_m^{c_{\mathbf{v}},\text{in}} = \sum_{\mathbf{w}:(\mathbf{v},\mathbf{w})\in\mathcal{E}} f_m(\mathbf{v},\mathbf{w}) + 1_{v_{\mathbf{v}}=w_m} \lambda_m^{t_{\mathbf{v}},c_{\mathbf{v}},\text{out}}, \forall \mathbf{v} \in \mathcal{V}, m \in \{1,\dots,M\}, \tag{5.1a}$$

$$\sum_{c=1}^{C} \lambda_m^{c,\text{in}} = \lambda_m, \qquad \sum_{t=1}^{T}\sum_{c=1}^{C} \lambda_m^{t,c,\text{out}} = \lambda_m, \qquad \forall m \in \{1,\dots,M\}. \tag{5.1b}$$

where the variable $\lambda_m^{c,\text{in}}$ denotes the customer rate departing with charge level c and the variable $\lambda_m^{t,c,\text{out}}$ denotes the customer rate reaching the destination at time t with charge level c; both are optimization variables. Function 1_x denotes the indicator function of the Boolean variable $x = \{\text{true, false}\}$, that is $1_x = 1$ if x is true, and $1_x = 0$ if x is false.

Rebalancing flows must satisfy a continuity condition analogous to the one for the customer flows. In addition, rebalancing flows must satisfy a *consistency* condition representing the fact that a customer may only depart the origin location if an empty vehicle is available. Finally, the initial position and charge level of the vehicles is fixed; the final position and charge level is an optimization variable (possibly subject to constraints, e.g., on the minimum final charge level). The constraints for the initial and final positions of the rebalancing vehicles at each node $\mathbf{v} \in \mathcal{V}$ are captured by a set of functions $N_I(\mathbf{v})$ and $N_F(\mathbf{v})$, respectively. Formally, $N_I(\mathbf{v})$, with $t_{\mathbf{v}} = 0$, denotes the number of rebalancing vehicles entering the AMoD system at location $v_{\mathbf{v}}$ at time $t_{\mathbf{v}}$ with charge level $c_{\mathbf{v}}$. Conversely, $N_F(\mathbf{v})$, with $t_{\mathbf{v}} = T$ denotes the number of rebalancing vehicles at location $v_{\mathbf{v}}$ at time $t_{\mathbf{v}}$

[1]Note that, in this chapter, we denote the rebalancing flow as f_0 for brevity; in previous chapters, the rebalancing flow is denoted as f_R.

with charge level $c_{\mathbf{v}}$. For $t_{\mathbf{v}} \neq 0$, $N_I(\mathbf{v}) = 0$; for $t_{\mathbf{v}} \neq T$, $N_F(\mathbf{v}) = 0$. The overall number of vehicles in the network is $\sum_{\mathbf{v}\in\mathcal{V}} N_I(\mathbf{v})$. Equation (5.2) simultaneously enforces the rebalancing vehicles' continuity condition, consistency condition, and the constraints on the initial and final locations:

$$\sum_{\mathbf{u}:(\mathbf{u},\mathbf{v})\in\mathcal{E}} f_0(\mathbf{u},\mathbf{v}) + \sum_{m=1}^{M} 1_{v_{\mathbf{v}}=w_m} \lambda_m^{t_{\mathbf{v}},c_{\mathbf{v}},\text{out}} + N_I(\mathbf{v}) =$$
$$\sum_{\mathbf{w}:(\mathbf{v},\mathbf{w})\in\mathcal{E}} f_0(\mathbf{v},\mathbf{w}) + \sum_{m=1}^{M} 1_{v_{\mathbf{v}}=v_m} 1_{t_{\mathbf{v}}=t_m} \lambda_m^{c_{\mathbf{v}},\text{in}} + N_F(\mathbf{v}), \forall \mathbf{v} \in \mathcal{V}. \tag{5.2}$$

Congestion: We adopt a simple *threshold* model for congestion: the vehicle flow on each road link is constrained to be smaller than the road link's capacity. The model is analogous to the one adopted in [Rossi et al., 2018] and is consistent with classical traffic flow theory [Wardrop, 1952]. This simplified congestion model is adequate for our goal of *controlling* the vehicles' routes and charging schedules, and ensures tractability of the resulting optimization problem; higher-fidelity models can be used for the *analysis* of the AMoD system's operations. Equation (5.3) enforces the road congestion constraint:

$$\sum_{c_{\mathbf{v}}=1}^{C} \sum_{m=0}^{M} f_m(\mathbf{v},\mathbf{w}) \leq \overline{f}_{(v_{\mathbf{v}},v_{\mathbf{w}})}, \forall (v_{\mathbf{v}}, v_{\mathbf{w}}) \in \mathcal{E}_R, t_{\mathbf{v}} \in \{1,\ldots,T\}. \tag{5.3}$$

Charging stations can simultaneously accommodate a limited number of vehicles. The station capacity constraint is enforced with Equation (5.4):

$$\sum_{\substack{(\mathbf{v},\mathbf{w})\in\mathcal{E}_S:\\ v_{\mathbf{v}}=v_{\mathbf{w}}=v}} \sum_{m=0}^{M} f_m(\mathbf{v},\mathbf{w}) \leq \overline{S}_{v_{\mathbf{v}}}, \forall v \in \mathcal{S}, t \in \{1,\ldots,T\}. \tag{5.4}$$

Network flow model of an AMoD system: The travel time T_M experienced by customers, a proxy for customer welfare, and the overall mileage D_V driven by (both customer-carrying and empty) vehicles, a proxy for vehicle wear, are given by

$$T_M = \sum_{(\mathbf{v},\mathbf{w})\in\mathcal{E}} t_{\mathbf{v},\mathbf{w}} \sum_{m=1}^{M} f_m(\mathbf{v},\mathbf{w}),$$
$$D_V = \sum_{(\mathbf{v},\mathbf{w})\in\mathcal{E}} d_{v_{\mathbf{v}},v_{\mathbf{w}}} \sum_{m=0}^{M} f_m(\mathbf{v},\mathbf{w}),$$

Note that T_M only includes the travel time of *customer-carrying* vehicles, whereas D_V includes the distance traveled by *all* vehicles. Also note that, for charging edges, $d_{v_{\mathbf{v}},v_{\mathbf{w}}} = 0$. The overall

cost of electricity incurred by the vehicles (including any credit from selling electricity to the power network) is

$$V_E = \sum_{(\mathbf{v},\mathbf{w})\in\mathcal{E}_S} \sum_{m=0}^{M} f_m(\mathbf{v},\mathbf{w})\delta c_{v_\mathbf{v}} p_{(\mathbf{v},\mathbf{w})},$$

where $\delta c_{v_\mathbf{v}} = \delta c^+_{v_\mathbf{v}}$ and $p_{(\mathbf{v},\mathbf{w})} = p^+_{v_\mathbf{v}}$ if $c_\mathbf{w} > c_\mathbf{v}$, $\delta c_{v_\mathbf{v}} = \delta c^-_{v_\mathbf{v}}$ and $p_{(\mathbf{v},\mathbf{w})} = p^-_{v_\mathbf{v}}$ otherwise.

The overall battery depreciation due to charging and discharging is

$$V_B = \sum_{(\mathbf{v},\mathbf{w})\in\mathcal{E}_S} \sum_{m=0}^{M} f_m(\mathbf{v},\mathbf{w})|\delta c_{v_\mathbf{v}}|d_B.$$

A more general question about depreciation: we do not currently include it in the sims. We can include it (a few days to redo the sims), keep the sims as is, or just do not talk about it and remove these references. Opinions?

The goal of the TSO is to solve the Vehicle Routing and Charging problem, that is, to minimize the aggregate societal cost borne by the AMoD users while satisfying all operational constraints. We define the customers' value of time (i.e., the monetary loss associated with traveling for one unit of time) as V_T and the operation cost per kilometer of the vehicles (including maintenance but excluding electricity costs) as V_D. The aggregate societal cost experienced by the AMoD users is then defined as

$$V_D D_V + V_E + V_B + V_T T_M. \tag{5.5}$$

We are now in a position to state the TSO's Vehicle Routing and Charging problem:

$$\underset{f_m,\lambda_m^{c,\text{in}},\lambda_m^{t,c,\text{out}},N_F}{\text{minimize}} \quad (5.5), \tag{5.6a}$$

$$\text{subject to} \quad (5.1), (5.2), (5.3), \text{ and } (5.4). \tag{5.6b}$$

5.2.2 Linear model of power network

In this work, the power network is modeled according to the well-known DC model [Kirschen and Strbac, 2004, Ch. 6], which, by assuming constant voltage magnitudes and determining the power flow on transmission lines solely based on voltage phase angles, represents an approximation to the higher-fidelity AC flow model [Glover et al., 2011]. In analogy with the treatment of the AMoD model, we discretize the time horizon of the problem in T time steps. The power grid is modeled as an undirected graph $P = (\mathcal{B}, \mathcal{E}_P)$, where $\mathcal{B}$ is the node set, commonly referred to as buses in the power engineering literature, and $\mathcal{E}_P \subseteq \mathcal{B} \times \mathcal{B}$ is the edge set, representing the

transmission lines. The subsets of buses representing generators and loads are defined as $\mathcal{G} \subset \mathcal{B}$ and $\mathcal{L} \subset \mathcal{B}$, respectively. Nodes that are neither loads nor generators are referred to as interconnects. Generators produce power and deliver it to the network, while loads absorb power from the network. Each generator $g \in \mathcal{G}$ is characterized by a maximum output power $\overline{p}_g(t)$, a minimum output power $\underline{p}_g(t)$, a unit generation cost $o_g(t)$, and maximum ramp-up and ramp-down rates $p_g^+(t)$ and $p_g^-(t)$, respectively. Transmission lines $e \in \mathcal{E}_P$ are characterized by a reactance x_e and a maximum allowable power flow $\overline{p}_e$ (due chiefly to thermal constraints). The reactance and the maximum allowable power flow do not vary with time. Each load node $l \in \mathcal{L}$ is characterized by a required power demand $d_l(t)$. The distribution network is not modeled explicitly; however, thermal constraints due to the distribution substation transformers are modeled by an upper bound $\overline{d}_l(t)$ on the power that can be delivered at each load node.

We define a generator power function $p : (\mathcal{G}, \{1, \ldots, T\}) \mapsto \mathbb{R}_{\geq 0}$, and a phase angle function $\theta : (\mathcal{B}, \{1, \ldots, T\}) \mapsto \mathbb{R}$. The generation cost is defined as

$$C_G = \sum_{t=1}^{T} \sum_{g \in \mathcal{G}} o_g(t) p(g, t)$$

The Economic Dispatch problem entails minimizing the generation cost subject to a set of feasibility constraints [Kirschen and Strbac, 2004], namely:

$$\begin{aligned}
\underset{p,\theta}{\text{minimize}} \quad & C_G && && \text{(5.7a)}\\
\text{subject to} \quad & \sum_{(u,v)\in\mathcal{E}_P} \frac{\theta(u,t)-\theta(v,t)}{x_{u,v}} + 1_{v\in\mathcal{G}} p(v,t) = 1_{v\in\mathcal{L}} d_v(t) + \sum_{(v,w)\in\mathcal{E}_P} \frac{\theta(v,t)-\theta(w,t)}{x_{v,w}} && \\
& && \forall v \in \mathcal{B}, t \in \{1,\ldots,T\}, && \text{(5.7b)}\\
& -\overline{p}_{b_1,b_2} \leq \frac{\theta(b_1,t)-\theta(b_2,t)}{x_{b_1,b_2}} \leq \overline{p}_{b_1,b_2} && \forall (b_1,b_2) \in \mathcal{E}_P, t \in \{1,\ldots,T\}, && \text{(5.7c)}\\
& \underline{p}_g(t) \leq p(g,t) \leq \overline{p}_g(t), && \forall g \in \mathcal{G}, t \in \{1,\ldots,T\}, && \text{(5.7d)}\\
& -p_g^-(t) \leq p(g,t+1) - p(g,t) \leq p_g^+(t) && \forall g \in \mathcal{G}, t \in \{1,\ldots,T-1\}, && \text{(5.7e)}\\
& d_l(t) \leq \overline{d}_l(t), && \forall l \in \mathcal{L}, t \in \{1,\ldots,T\}. && \text{(5.7f)}
\end{aligned}$$

Equation (5.7b) enforces power balance at each bus based on the so-called DC power flow equations; Equation (5.7c) encodes the transmission lines' thermal constraints; Equation (5.7d) encodes the generation capacity constraints; Equation (5.7e) encodes the ramp-up and ramp-down constraints; and Equation (5.7f) encodes the thermal constraints of substation transformers at load nodes.

The unit price of electricity at the load nodes is determined through a mechanism known as

Locational Marginal Pricing (LMP) [Kirschen and Strbac, 2004, Liu et al., 2009]. The LMP at a node is defined as the *marginal cost* of delivering one unit of power at the node while respecting all the system constraints. Accordingly, in this work, the LMP at each bus equals the sum of the dual variables (i.e., the shadow prices) corresponding to the power injection constraint (5.7b) and the substation transformer thermal constraint (5.7f) at the same bus in the Economic Dispatch problem (5.7).

5.2.3 Power-in-the-loop AMoD system

The vehicles' charging requirements introduce a *coupling* between the AMoD system and the power network, as shown in Figure 5.2. Specifically, the vehicles' charging schedule produces a load on the power network. Such a load on the power network affects the solution to the ISO's Economic Dispatch problem and, as a result, the LMPs. The change in LMPs, in turn, has an effect on the TSO's optimal charging schedule. In absence of coordination, this feedback loop can lead to system instability, as shown for the case of privately-owned, non-autonomous EVs in [Alizadeh et al., 2017].

In this section, we formulate a *joint model* for the TSO's Vehicle Routing and Charging problem and the ISO's Economic Dispatch problem. The goal of this model is to maximize the social welfare by minimizing the total cost of mobility (a profit-maximizing formulation would be similar) and the total cost of power generation and transmission. While the resulting solution is not directly actionable (since it requires the TSO and the ISO to coordinate and share their private information), pricing mechanisms can be designed to steer the system towards the optimum: we discuss such mechanisms in the section on *Distributed solution to the P-AMoD problem.*

The coupling between the AMoD model and the electric power model is mediated by the charging stations. A given charging station is represented both by a node $v \in \mathcal{V}_R$ in the road network and by a load node $l \in \mathcal{L}$ in the power network. To capture this correspondence, we define an auxiliary function $\mathcal{M}_{\mathrm{P,R}} : \mathcal{L} \mapsto \{\mathcal{V}_R \cup \emptyset\}$. Given a load node $b \in \mathcal{L}$, $\mathcal{M}_{P,R}(b)$ denotes the node in $\mathcal{V}_R$ (if any) that represents a charging station connected to b. We then define two additional functions, $\mathcal{M}^{+}_{\mathrm{P,G}} : (\mathcal{L}, \{1, \dots, T\}) \mapsto \{\mathcal{E}_S \cup \emptyset\}$ and $\mathcal{M}^{-}_{\mathrm{P,G}} : (\mathcal{L}, \{1, \dots, T\}) \mapsto \{\mathcal{E}_S \cup \emptyset\}$. The function $\mathcal{M}^{+}_{\mathrm{P,G}}$ (resp. $\mathcal{M}^{-}_{\mathrm{P,G}}$) maps a load node l and a time t to the set of charge (resp. discharge) edges in G corresponding to station $\mathcal{M}_{\mathrm{P,R}}(l)$ at time t. Formally,

$$\mathcal{M}^{+}_{\mathrm{P,G}}(l,t) : \{(\mathbf{v},\mathbf{w}) \in \mathcal{E}_S | v_{\mathbf{v}} = v_{\mathbf{w}}, v_{\mathbf{v}} \in \mathcal{M}_{\mathrm{P,R}}(l), c_{\mathbf{v}} < c_{\mathbf{w}}, t_{\mathbf{v}} \leq t < t_{\mathbf{w}}\},$$
$$\mathcal{M}^{-}_{\mathrm{P,G}}(l,t) : \{(\mathbf{v},\mathbf{w}) \in \mathcal{E}_S | v_{\mathbf{v}} = v_{\mathbf{w}}, v_{\mathbf{v}} \in \mathcal{M}_{\mathrm{P,R}}(l), c_{\mathbf{v}} > c_{\mathbf{w}}, t_{\mathbf{v}} \leq t < t_{\mathbf{w}}\}.$$

The load at a load bus l can be expressed as the sum of two components: an exogenous demand $d_{l,e}$ and the load due to the charger or chargers connected to that bus, quantitatively,

$$d_l(t) = d_{l,e}(t) + J_C \delta c^+_{\mathcal{M}_{\mathrm{P,R}}(l)} \sum_{(\mathbf{v},\mathbf{w}) \in M^+_{P,G}(l,t)} \sum_{m=0}^{M} f_m(\mathbf{v},\mathbf{w}) + J_C \delta c^-_{\mathcal{M}_{\mathrm{P,R}}(l)} \sum_{(\mathbf{v},\mathbf{w}) \in M^-_{P,G}(l,t)} \sum_{m=0}^{M} f_m(\mathbf{v},\mathbf{w}), \qquad \forall l \in \mathcal{L}, t \in \{1,\dots,T\}. \tag{5.8}$$

We are now in a position to state the Power-in-the-loop AMoD (P-AMoD) problem:

$$\underset{f_m, \lambda_m^{c,\mathrm{in}}, \lambda_m^{t,c,\mathrm{out}}, N_F, \theta, p}{\text{minimize}} \quad V_T T_M + V_D D_v + V_B + C_G, \tag{5.9a}$$

$$\text{subject to} \quad (5.1), (5.2), (5.3), (5.4), (5.7), \text{ and } (5.8). \tag{5.9b}$$

5.2.4 Discussion

Some comments are in order. The model assumes that the TSO and the ISO share the goal of maximizing social welfare and are willing to collaborate on a joint policy. This assumption is, in general, not realistic: not only do the TSO and ISO have different goals, but they are also generally reluctant to share the information required for successful coordination. However, once a socially optimal strategy is found, efficient coordination mechanisms can be designed that steer rational agents towards that strategy: in the section on *Distributed solution to the P-AMoD problem*, we show that the social optimum is a general equilibrium for a self-interested TSO, self-interested power generators, and a non-profit ISO acting as a market broker, and we propose a distributed *privacy-preserving* mechanism that an ISO and a TSO can adopt to compute such equilibrium.

The network flow model has some well-known limitations: chiefly, it does not capture the stochasticity of the customer arrival process, and it does not directly yield *integral* routes suitable for real-time control of vehicles. Furthermore, in this work, customer requests are assumed to be deterministic and known in advance, an assumption that is not consistent with the paradigm of on-demand mobility. To overcome these limitation, we propose a receding-horizon implementation of Problem (5.9) in the section on *A Receding-Horizon Algorithm for P-AMoD*. Moreover, future requests may be interpreted as the *expected* number of future transportation requests (which may be learned from historical data and demand models): accordingly, the model proposed in this section may be used for planning on timescales of days and hours, akin to the Day-Ahead-Market already in use in the electric power network [Kirschen and Strbac, 2004].

A spectrum of models to predict and control power flows are available, ranging from high-fidelity nonlinear AC models, which fully capture the physical behavior of the network, to linear DC models which assume constant voltage magnitudes, neglect reactive power and line resistances,

and assume that the phase angle differences between buses are small [Glover et al., 2011]. While the DC model has some shortcomings, chiefly the inability to handle voltage constraints [Hogan, 1996] and system-dependent accuracy [Stott et al., 2009], it offers several benefits that have resulted in its wide industry use. The DC model's linearity makes it amenable to large-scale optimization and easy to integrate within the economic theory upon which the transmission-oriented market design is based on [Stott et al., 2009]. Most importantly, the DC model is used by ISOs to calculate locational marginal prices for wholesale electricity transactions [Overbye et al., 2004]. Hence, the DC model is appropriate for high-level synthesis of joint control policies such as those considered in this work.

5.3 Solution Algorithms

The number of variables of the P-AMoD problem (5.9) is $(M+1)|\mathcal{E}|+MC(T+1)+|\mathcal{V}_R|C+T(|\mathcal{G}|+|\mathcal{B}|)$. The size of the edge set $\mathcal{E}$ is $|\mathcal{E}| = \Theta((|\mathcal{E}_R| + |\mathcal{S}|)CT)$, and the number of customer demands admits an upper bound $M = O(|\mathcal{V}_R|^2 T)$, since each customer demand is associated with an origin, a destination, and a departure time. The size of the problem is dominated by the customer flow variables in the road network – the number of such variables is $M|\mathcal{E}| = O((|\mathcal{V}_R|^2 T)(|\mathcal{E}_R| + |\mathcal{S}|)CT)$. Consider a typical problem with 25 road nodes, 200 road links, 30 charge levels, and a horizon of 20 time steps. Such problem results in a number of variables on the order of $2 \cdot 10^9$, which can not be solved even by state-of-the-art solvers on modern hardware [Mittelmann, 2016].

In this section, we propose a *bundling* procedure that collects multiple customer demands in a single customer flow without loss of information. The procedure allows one to reduce the number of network flows to $O(|\mathcal{V}_R|)$: as a result, the size of the prototypical problem above is reduced to $4 \cdot 10^6$ variables, well within reach of modern solvers. The procedure relies on the notion of *bundled customer flow*,

Definition 5.3.1 (Bundled customer flow)**.** *Consider the set of customer requests* $\{v_m, w_m, t_m, \lambda_m\}_{m=1}^M$. *Denote the set of customer destinations as* $\mathcal{D} := \{\cup_{m=1}^M w_m\}$. *For a given destination* $d_B \in \mathcal{D}$, *we define a* bundled customer flow *as a function* $f_{B,d_B}(\mathbf{u}, \mathbf{v}) : \mathcal{E} \mapsto \mathbb{R}_{\geq 0}$ *that satisfies*

$$\sum_{\mathbf{u}:(\mathbf{u},\mathbf{v})\in\mathcal{E}} f_{B,d_B}(\mathbf{u},\mathbf{v}) + \sum_{\substack{m\in\{1,\dots,M\}:\\ w_m=d_B}} 1_{v_\mathbf{v}=v_m} 1_{t_\mathbf{v}=t_m} \lambda_m^{c_\mathbf{v},in} = \sum_{\mathbf{w}:(\mathbf{v},\mathbf{w})\in\mathcal{E}} f_{B,d_B}(\mathbf{v},\mathbf{w}) + \sum_{\substack{m\in\{1,\dots,M\}:\\ w_m=d_B}} 1_{v_\mathbf{v}=w_m} \lambda_m^{t_\mathbf{v},c_\mathbf{v},out}, \forall \mathbf{v} \in \mathcal{V}, \quad (5.10a)$$

$$\sum_{c=1}^{C} \lambda_m^{c,in} = \lambda_m, \forall m \in \{1,\dots,M\} : w_m = d_B, \quad (5.10b)$$

$$\sum_{t=1}^{T}\sum_{c=1}^{C} \lambda_m^{t,c,out} = \lambda_m, \forall m \in \{1,\dots,M\} : w_m = d_B. \quad (5.10c)$$

Intuitively, the bundled customer flow for a given destination d_B can be thought of as the sum of customer flows (i.e., network flows satisfying Eq. (5.1)) for *all* customer requests whose destination is node d_B. A bundled customer flow is an *equivalent representation* for a set of customer flows belonging to customer requests sharing the same destination. The next lemma formalizes this intuition.

Lemma 5.3.2 (Equivalency between customer flows and bundled customer flows)**.** *Consider a network $G(\mathcal{V}, \mathcal{E})$ and a set of customer requests $\{v_m, w_m, t_m, \lambda_m\}_{m=1}^M$. Assume there exists a bundled customer flow $\{f_{B,d_B}(\mathbf{u}, \mathbf{v})\}_{(\mathbf{u},\mathbf{v})\in\mathcal{E}}$ that satisfies Equation (5.10) for a destination $d_B \in \mathcal{D}$. Then, for each customer request $\{v_m, d_B, t_m, \lambda_m\}$ with destination d_B, there exists a customer flow $f_m(\mathbf{u}, \mathbf{v})$ that satisfies Eq. (5.1). Furthermore, for each edge $(\mathbf{u}, \mathbf{v}) \in \mathcal{E}$, $f_{B,d_B}(\mathbf{u}, \mathbf{v}) = \sum_{m\in\{1,\dots,M\}:w_m=d_B} f_m(\mathbf{u}, \mathbf{v})$.*

Proof sketch: The proof is constructive. Define as path flow a network flow that has a fixed intensity on edges belonging to a path without cycles from the origin to the destination and zero otherwise. The flow decomposition algorithm [Ahuja et al., 1993, Ch. 3.5] is used to decompose the bundled customer flow into a collection of path flows, each with a single origin node $\mathbf{v} \in \mathcal{V}$ and destination node $\mathbf{w} \in \mathcal{V}$ with $v_{\mathbf{w}} = d_B$. The customer flow for customer request (v_m, d_B, t, λ) is then obtained as the sum of path flows leaving origin nodes $\{\mathbf{v} = (v_m, t_m, c)\}_{c=1}^C$ with total intensity λ_m.

We can leverage the result in Lemma 5.3.2 to solve the P-AMoD problem in terms of bundled customer flows, thus dramatically decreasing the problem size. The next theorem formalizes this intuition.

Theorem 5.3.3 (P-AMoD with bundled customer flows)**.** *Consider the following problem, referred to as the* bundled P-AMoD problem*:*

$$\begin{aligned} &\underset{\substack{f_0, f_{B,d_B}, \lambda_m^{c,in},\\ \lambda_m^{t,c,out}, N_F, \theta, p}}{\textit{minimize}} && V_T T_M + V_D D_v + V_B + C_G && (5.11)\\ &\textit{subject to} && (5.10) \quad \forall d_B \in \mathcal{D},\\ & && (5.2), (5.3), (5.4), (5.7), \textit{ and } (5.8), \end{aligned}$$

where each instance of $\sum_{m=1}^M f_m$ in the cost function and in Equations (5.2), (5.3), (5.4), (5.7), and (5.8) is replaced by $\sum_{d_B\in\mathcal{D}} f_{B,d_B}$. The bundled P-AMoD problem (5.11) admits a feasible solution if and only if the P-AMoD problem (5.9) admits a solution. Furthermore, the optimal values of Problem (5.9) and Problem (5.11) are equal.

Proof. (i) The bundled P-AMoD problem admits a solution if the P-AMoD problem admits a solution. Consider a solution to the P-AMoD problem. For each destination node, define the bundled

flow as the sum of the customer flows for customers directed to that destination:

$$f_{B,d_B} = \sum_{m:w_m=d_B} f_m \quad \forall d_B \in \mathcal{B}.$$

It is easy to verify that the resulting network flow satisfies Eq. (5.10). Also, the customer flows satisfy Equations (5.2), (5.3), (5.4), (5.7), and (5.8) and, for every edge, by construction $\sum_{d_B\in\mathcal{B}} f_{B,d_B} = \sum_{d_B\in\mathcal{B}}\sum_{m:w_m=d_B} f_m = \sum_m f_m$. Therefore the set of bundled customer flows $\{f_{B,d_B}\}_{d_B\in\mathcal{D}}$ satisfies Equations (5.2), (5.3), (5.4), (5.7), and (5.8) where each instance of $\sum_{m\in[1,M]} f_m$ is replaced by $\sum_{d_B\in\mathcal{D}} f_{B,d_B}$.

(ii) The P-AMoD problem admits a solution if the bundled P-AMoD problem admits a solution. Lemma 1 shows that, if there exists a set of bundled flows that satisfy Problem (5.11), then there exists a set of customer flows that satisfy Equation (5.1). Furthermore, for each edge, $\sum_m f_m = \sum_{d_B\in\mathcal{B}} f_{B,d_B}$. Since the bundled flows satisfies the modified version of Equations (5.2), (5.3), (5.4), (5.7), and (5.8), the customer flows also satisfy them.

(iii) The bundled P-AMoD problem and the P-AMoD problem have the same cost. Due to Lemma 1, $\sum_m f_m = \sum_{d_B\in\mathcal{B}} f_{B,d_B}$. The claim follows from the definition of the cost in Problem (5.11). □

The optimization problem in (5.11) can be solved with a number of variables that is $O((|\mathcal{V}_R|+1)|\mathcal{E}| + MC + |\mathcal{V}_R|C + T(|\mathcal{G}| + |\mathcal{E}_p| + |\mathcal{B}|))$. To see this, note that in Equation (5.10) the variables $\{\lambda_m^{t,c,\text{out}}\}_{\{m,t,c\}}$ only appear as part of the sum $\sum_{m\in\{1,\dots,M\}:w_m=d_B} \lambda_m^{t,c,\text{out}}$ and therefore may be replaced by the smaller set of variables $\{\lambda_{d_B}^{t,c,\text{out}}\}_{\{d_B,t,c\}}$, where $\lambda_{d_B}^{t,c,\text{out}} := \sum_{m\in\{1,\dots,M\}:w_m=d_B} \lambda_m^{t,c,\text{out}}$, without loss of generality. The number of variables grows quadratically with the number of nodes in the road network, and grows linearly with the number of edges in the road network, the time horizon, the number of charge levels, and the number of nodes, edges, and generators in the power network. Crucially, the size of Problem (5.11) *does not depend on the number of customer requests.*

5.4 Distributed solution to the P-AMoD problem

The model and solution algorithms presented in the previous sections assume that the TSO and the ISO both wish to maximize social welfare for given generation costs; also, in order to compute the socially optimal solution to the P-AMoD problem, the TSO and the ISO must be willing to share their private information (e.g. customer transportation requests and power generation costs). In this section, we provide theoretical results and algorithmic tools to overcome these unrealistic assumptions. In particular, we pose the P-AMoD problem as a perfectly competitive market where self-interested power generator operators sell power to the power network, a self-interested TSO buys and sells power from the power network, and a non-profit ISO acts as a market broker (similar to the model in [Wang et al., 2012]). In this framework, we show that the socially optimal solution to the P-AMoD problem is also a general equilibrium [Kirschen and Strbac, 2004] for the TSO and the

power generator operators if the ISO sets the price of electricity through Locational Marginal Prices. Next, we propose a distributed privacy-preserving algorithm that the TSO and the ISO can use to compute the equilibrium prices without sharing private information on transportation demand or generation costs.

5.4.1 A general equilibrium

Theorem 5.4.1 (The socially optimal solution of the P-AMoD problem is a general equilibrium)**.** *Consider an optimal solution* $\{f_m^\star, \lambda_m^{c,in\star}, \lambda_m^{t,c,out\star}, N_F^\star, \theta^\star, p^\star\}$ *to the P-AMoD Problem* (5.9)*. Also consider a perfectly competitive market where a self-interested TSO solves the Vehicle Routing and Charging problem* (5.6) *by selecting variables* $\{f_m, \lambda_m^{c,in}, \lambda_m^{t,c,out}, N_F\}$*, self-interested power generator operators sell power to the network by determining the revenue-maximizing power generation schedule* $\{p\}$*, and a non-profit ISO acts as a market broker by setting locational marginal prices and controlling phase angles* $\{\theta\}$*. Then* $(\{f_m^\star, \lambda_m^{c,in\star}, \lambda_m^{t,c,out\star}, N_F^\star\}, \{\theta^\star\}, \{p^\star\})$ *is a general equilibrium.*

Proof. It is easy to show that the optimal solution to the P-AMoD problem also maximizes the revenue of the power generator operators (see e.g. [Wang et al., 2012, Sec. 3]) if locational marginal pricing is used. Thus, we focus on showing that the optimal solution to the P-AMoD problem is also an optimal solution to the TSO's problem (5.6).

The key insight in the proof is that that the term V_E in the cost function of the Vehicle Routing and Charging problem (5.6) captures the marginal cost imposed by the TSO on the power network, aligning the TSO's incentives with the social optimum.

For ease of notation, we rewrite Equations (5.1)-(5.2) and (5.3)-(5.4) as respectively

$$\begin{aligned} &f_{\text{TSO}}^{\text{eq}}(f_m, \lambda_m^{\text{c,in}}, \lambda_m^{t,c,\text{out}}, N_F) = 0, \text{ (Eq. (5.1)-(5.2))}, && \text{with dual variables } \lambda_{\text{TSO}}^{\text{eq}}, \\ &f_{\text{TSO}}^{\text{ineq}}(f_m, \lambda_m^{\text{c,in}}, \lambda_m^{t,c,\text{out}}, N_F) \leq 0, \text{ (Eq. (5.3)-(5.4))}, && \text{with dual variables } \mu_{\text{TSO}}^{\text{ineq}}. \end{aligned}$$

Analogously, we rewrite Equation (5.7b) and Equations (5.7c)-(5.7f) as respectively

$$\begin{aligned} &f_{\text{ISO}}^{\text{eq}}(f_m, \theta, p) = 0, \text{ (Eq. (5.7b))}, && \text{with dual variables } \lambda_{\text{ISO}}^{\text{eq}}, \\ &f_{\text{ISO}}^{\text{ineq}}(f_m, \theta, p) \leq 0, \text{ (Eq. (5.7c)-(5.7f))}, && \text{with dual variables } \mu_{\text{ISO}}^{\text{ineq}}. \end{aligned}$$

Note that Equations (5.7b) and (5.7f) are the only constraints that depend both on variables controlled by the TSO (namely, $\{f_m\}$, and specifically the charging and discharging schedule of the electric vehicles) and on variables controlled by the ISO and the power generators (namely, $\{\theta\}$ and $\{p\}$).

The KKT stationarity conditions for the P-AMoD Problem (5.9) are

$$\frac{\partial(V_T T_M + V_D D_V + V_B)}{\partial f_m(\mathbf{v},\mathbf{w})} + \lambda^{\text{eq}}_{\text{TSO}} \cdot \frac{\partial f^{\text{eq}}_{\text{TSO}}}{\partial f_m(\mathbf{v},\mathbf{w})} + \mu^{\text{ineq}}_{\text{TSO}} \cdot \frac{\partial f^{\text{ineq}}_{\text{TSO}}}{\partial f_m(\mathbf{v},\mathbf{w})} + \lambda^{\text{eq}}_{\text{ISO}} \cdot \frac{\partial f^{\text{eq}}_{\text{ISO}}}{\partial f_m(\mathbf{v},\mathbf{w})} + \mu^{\text{ineq}}_{\text{ISO}} \cdot \frac{\partial f^{\text{ineq}}_{\text{ISO}}}{\partial f_m(\mathbf{v},\mathbf{w})} = 0, \quad \forall m \in \{0,\dots,M\}, (\mathbf{v},\mathbf{w}) \in \mathcal{E}, \tag{5.12a}$$

$$\lambda^{\text{eq}}_{\text{TSO}} \cdot \frac{\partial f^{\text{eq}}_{\text{TSO}}}{\partial \lambda^{c,\text{in}}_m} = 0, \qquad \forall c \in \{0,\dots,C\}, m \in \{0,\dots,M\}, \tag{5.12b}$$

$$\lambda^{\text{eq}}_{\text{TSO}} \cdot \frac{\partial f^{\text{eq}}_{\text{TSO}}}{\partial \lambda^{t,c,\text{out}}_m} = 0, \quad \forall c \in \{0,\dots,C\}, t \in \{1,\dots,T\}, m \in \{0,\dots,M\}, \tag{5.12c}$$

$$\lambda^{\text{eq}}_{\text{TSO}} \cdot \frac{\partial f^{\text{eq}}_{\text{TSO}}}{\partial N_F(\mathbf{v})} = 0, \qquad \forall \mathbf{v} \in \mathcal{V}, \tag{5.12d}$$

$$\lambda^{\text{eq}}_{\text{ISO}} \cdot \frac{\partial f^{\text{eq}}_{\text{ISO}}}{\partial \theta(v,t)} + \mu^{\text{ineq}}_{\text{ISO}} \cdot \frac{\partial f^{\text{ineq}}_{\text{ISO}}}{\partial \theta(v,t)} = 0, \qquad \forall v \in \mathcal{B}, t \in \{1,\dots,T\}, \tag{5.12e}$$

$$\frac{\partial C_G}{\partial p(g,t)} + \lambda^{\text{eq}}_{\text{ISO}} \cdot \frac{\partial f^{\text{eq}}_{\text{ISO}}}{\partial p(g,t)} + \mu^{\text{ineq}}_{\text{ISO}} \cdot \frac{\partial f^{\text{ineq}}_{\text{ISO}}}{\partial p(g,t)} = 0, \qquad \forall g \in \mathcal{G}, t \in \{1,\dots,T\}. \tag{5.12f}$$

We show that $\{f^\star_m, \lambda^{c,\text{in}\star}_m, \lambda^{t,c,\text{out}\star}_m, N^\star_F\}$ is an optimal solution to the TSO's Vehicle Routing and Charging Problem (5.6) for fixed $\{\theta^\star\}$ and $\{p^\star\}$.

For a given set of variables $\{\theta^\star\}$ and $\{p^\star\}$, the KKT conditions for Problem (5.6) are

$$\frac{\partial(V_T T_M + V_D D_V + V_B)}{\partial f_m(\mathbf{v},\mathbf{w})} + \frac{\partial(V_E)}{\partial f_m(\mathbf{v},\mathbf{w})} + \lambda^{\text{eq}}_{\text{TSO}} \cdot \frac{\partial f^{\text{eq}}_{\text{TSO}}}{\partial f_m(\mathbf{v},\mathbf{w})} + \mu^{\text{ineq}}_{\text{TSO}} \cdot \frac{\partial f^{\text{ineq}}_{\text{TSO}}}{\partial f_m(\mathbf{v},\mathbf{w})} = 0, \quad \forall m \in \{0,\dots,M\}, (\mathbf{v},\mathbf{w}) \in \mathcal{E}, \tag{5.13a}$$

$$\lambda^{\text{eq}}_{\text{TSO}} \cdot \frac{\partial f^{\text{eq}}_{\text{TSO}}}{\partial \lambda^{c,\text{in}}_m} = 0, \qquad \forall c \in \{0,\dots,C\}, m \in \{0,\dots,M\}, \tag{5.13b}$$

$$\lambda^{\text{eq}}_{\text{TSO}} \cdot \frac{\partial f^{\text{eq}}_{\text{TSO}}}{\partial \lambda^{t,c,\text{out}}_m} = 0, \quad \forall c \in \{0,\dots,C\}, t \in \{1,\dots,T\}, m \in \{0,\dots,M\}, \tag{5.13c}$$

$$\lambda^{\text{eq}}_{\text{TSO}} \cdot \frac{\partial f^{\text{eq}}_{\text{TSO}}}{\partial N_F(\mathbf{v})} = 0, \qquad \forall \mathbf{v} \in \mathcal{V}. \tag{5.13d}$$

The second term in Eq. (5.13a) is

$$\frac{\partial(V_E)}{\partial f_m(\mathbf{v},\mathbf{w})} = 1_{(\mathbf{v},\mathbf{w}) \in \mathcal{E}_S} p_{(\mathbf{v},\mathbf{w})} \delta c_{v_\mathbf{v}}$$

where $\delta c_{v_\mathbf{v}} = \delta c^+_{v_\mathbf{v}}$ if $c_\mathbf{w} > c_\mathbf{v}$ and $\delta c_{v_\mathbf{v}} = \delta c^-_{v_\mathbf{v}}$ otherwise.

Leveraging Eq. (5.8), the last two terms in Eq. (5.12a) can be rewritten as

$$\lambda^{\text{eq}}_{\text{ISO}} \cdot \frac{\partial f^{\text{eq}}_{\text{ISO}}}{\partial f_m(\mathbf{v},\mathbf{w})} + \mu^{\text{ineq}}_{\text{ISO}} \cdot \frac{\partial f^{\text{ineq}}_{\text{ISO}}}{\partial f_m(\mathbf{v},\mathbf{w})} =$$
$$\sum_{l\in\mathcal{B}}\sum_{t=1}^{T} \lambda^{\text{eq}}_{\text{ISO}}(l,t) J_C \left[\delta c^{+}_{\mathcal{M}_{\text{P,R}}(l)} 1_{(\mathbf{v},\mathbf{w})\in M^{+}_{P,G}(l,t)} + \delta c^{-}_{\mathcal{M}_{\text{P,R}}(l)} 1_{(\mathbf{v},\mathbf{w})\in M^{-}_{P,G}(l,t)}\right] +$$
$$\sum_{l\in\mathcal{B}}\sum_{t=1}^{T} \mu^{\text{ineq}}_{\text{ISO}}(l,t) J_C \left[\delta c^{+}_{\mathcal{M}_{\text{P,R}}(l)} 1_{(\mathbf{v},\mathbf{w})\in M^{+}_{P,G}(l,t)} + \delta c^{-}_{\mathcal{M}_{\text{P,R}}(l)} 1_{(\mathbf{v},\mathbf{w})\in M^{-}_{P,G}(l,t)}\right] =$$
$$J_C \delta c_{v_{\mathbf{v}}} \sum_{l\in\mathcal{B}}\sum_{t=1}^{T} \left[\left(1_{(\mathbf{v},\mathbf{w})\in M^{+}_{P,G}(l,t)} + 1_{(\mathbf{v},\mathbf{w})\in M^{-}_{P,G}(l,t)}\right)\left(\lambda^{\text{eq}}_{\text{ISO}}(l,t) + \mu^{\text{ineq}}_{\text{ISO}}(l,t)\right)\right].$$

Every edge $(\mathbf{v},\mathbf{w}) \in \mathcal{E}_S$ corresponds to a single load node $l \in \mathcal{B} : v_{\mathbf{v}} = \mathcal{M}_{\text{P,R}}(l)$ at a single time $t = t_{\mathbf{v}}$: thus, the expression above can be rewritten as

$$\lambda^{\text{eq}}_{\text{ISO}} \cdot \frac{\partial f^{\text{eq}}_{\text{ISO}}}{\partial f_m(\mathbf{v},\mathbf{w})} + \mu^{\text{ineq}}_{\text{ISO}} \cdot \frac{\partial f^{\text{ineq}}_{\text{ISO}}}{\partial f_m(\mathbf{v},\mathbf{w})} = J_C \delta c_{v_w} \left(\lambda^{\text{eq}}_{\text{ISO}}(l_{v_{\mathbf{v}}},t_{\mathbf{v}}) + \mu^{\text{ineq}}_{\text{ISO}}(l_{v_{\mathbf{v}}},t_{\mathbf{v}})\right),$$

where $l_{v_{\mathbf{v}}}$ is such that $v_{\mathbf{v}} = \mathcal{M}_{\text{P,R}}(l_{v_{\mathbf{v}}})$.

The vector $(\lambda^{\text{eq}}_{\text{ISO}} + \mu^{\text{ineq}}_{\text{ISO}})$ denotes the locational marginal price of energy at each bus in the power network, as discussed in the *Linear model of power network* section. That is,

$$p_{(\mathbf{v},\mathbf{w})} = J_C \left(\lambda^{\text{eq}}_{\text{ISO}}(l_{v_{\mathbf{v}}},t_{\mathbf{v}}) + \mu^{\text{ineq}}_{\text{ISO}}(l_{v_{\mathbf{v}}},t_{\mathbf{v}})\right), \text{ where } l_{v_{\mathbf{v}}} : v_{\mathbf{v}} = \mathcal{M}_{\text{P,R}}(l_{v_{\mathbf{v}}}).$$

(note that $p_{(\mathbf{v},\mathbf{w})}$ is the price per discrete energy level, whereas $(\lambda^{\text{eq}}_{\text{ISO}}(l,t) + \mu^{\text{ineq}}_{\text{ISO}}(l,t))$ is the price per unit of energy). Therefore, Equations (5.13a) and (5.12a) are identical. As a result, the KKT conditions for the TSO's problem (5.6) are verified whenever the KKT conditions for the P-AMoD problem (5.9) are verified, and $\{f^\star_m, \lambda^{c,\text{in}\star}_m, \lambda^{t,c,\text{out}\star}_m, N^\star_F\}$ is an optimal solution to Problem (5.6) for fixed $\{\theta^\star\}$ and $\{p^\star\}$.

In conclusion, $\{f^\star_m, \lambda^{c,\text{in}\star}_m, \lambda^{t,c,\text{out}\star}_m, N^\star_F\}$ is the solution to the TSO's VRCP (5.6) if the prices are set according to LMPs. In addition, the generation schedule $\{p^\star\}$ is the optimal (revenue-maximizing) schedule for self-interested power generator operators if the prices are set according to LMPs [Wang et al., 2012, Sec. 3].

That is, the set of variables $(\{f^\star_m, \lambda^{c,\text{in}\star}_m, \lambda^{t,c,\text{out}\star}_m, N^\star_F\}, \{\theta^\star\}, \{p^\star\})$ is a general equilibrium for a perfectly competitive market with a self-interested TSO, self-interested power generators, and a non-profit ISO. This concludes the proof. □

5.4.2 A distributed algorithm for the P-AMoD problem

Next, we show that the TSO and the ISO [Bertsekas, 1999, Ch. 6.4] can compute the socially optimal solution (and thus the general equilibrium) without disclosing their private information.

Our approach is similar to the one in [Alizadeh et al., 2017].

Consider a partial Lagrangian relaxation of Problem (5.9), i.e.

$$\underset{f_m,\lambda_m^{\mathrm{c,in}},\lambda_m^{t,c,\mathrm{out}},N_F,\theta,p}{\text{minimize}} \quad V_T T_M(f_m) + V_D D_v(f_m) + V_B(f_m) + C_G(p) + \lambda_{\mathrm{ISO}}^{\mathrm{eq}} f_{\mathrm{ISO}}^{\mathrm{eq}}(f_m,\theta,p) + \mu_{\mathrm{ISO}}^{\mathrm{ineq}} f_{\mathrm{ISO}}^{\mathrm{ineq}}(f_m,\theta,p) \tag{5.14a}$$

$$\text{subject to} \quad f_{\mathrm{TSO}}^{\mathrm{eq}}(f_m,\lambda_m^{\mathrm{c,in}},\lambda_m^{t,c,\mathrm{out}},N_F) = 0, \tag{5.14b}$$

$$f_{\mathrm{TSO}}^{\mathrm{ineq}}(f_m) \leq 0 \tag{5.14c}$$

In the dual decomposition algorithm, the TSO and the ISO iteratively optimize Problem (5.14) with respect to their own decision variables for a fixed value of the Lagrangian multipliers $\lambda_{\mathrm{ISO}}^{\mathrm{eq}}$ and $\mu_{\mathrm{ISO}}^{\mathrm{ineq}}$. Specifically, at step k of the iterative procedure, the TSO solves

$$\underset{f_m^k,\lambda_m^{\mathrm{c,in},k},\lambda_m^{t,c,\mathrm{out},k},N_F^k}{\text{minimize}} \quad V_T T_M(f_m^k) + V_D D_v(f_m^k) + V_B(f_m^k) + \lambda_{\mathrm{ISO}}^{\mathrm{eq},k-1} f_{\mathrm{ISO}}^{\mathrm{eq}}(f_m^k) + \mu_{\mathrm{ISO}}^{\mathrm{ineq},k-1} f_{\mathrm{ISO}}^{\mathrm{ineq}}(f_m^k), \tag{5.15a}$$

$$\text{subject to} \quad f_{\mathrm{TSO}}^{\mathrm{eq}}(f_m^k,\lambda_m^{\mathrm{c,in},k},\lambda_m^{t,c,\mathrm{out},k},N_F^k) = 0, \tag{5.15b}$$

$$f_{\mathrm{TSO}}^{\mathrm{ineq}}(f_m^k) \leq 0. \tag{5.15c}$$

The discussion in the Section on *A General Equilibrium* shows that minimizing the last two terms of Equation (5.15a) is equivalent to minimizing the cost of electricity V_E with prices $(\lambda_{\mathrm{ISO}}^{\mathrm{eq},k-1} + \mu_{\mathrm{ISO}}^{\mathrm{ineq},k-1})$. Thus, Problem (5.15) is equivalent to the Vehicle Routing and Charging Problem (5.6).

Analogously, at step k the ISO solves

$$\underset{\theta^k,p^k}{\text{minimize}} \quad C_G(p^k) + \lambda_{\mathrm{ISO}}^{\mathrm{eq},k-1} f_{\mathrm{ISO}}^{\mathrm{eq}}(\theta^k,p^k) + \mu_{\mathrm{ISO}}^{\mathrm{ineq},k-1} f_{\mathrm{ISO}}^{\mathrm{ineq}}(\theta^k,p^k) \tag{5.16}$$

The Lagrangian multipliers are then updated by the ISO as

$$\begin{aligned}\lambda_{\mathrm{ISO}}^{\mathrm{eq},k} &= \lambda_{\mathrm{ISO}}^{\mathrm{eq},k-1} + \alpha_k \left(f_{\mathrm{ISO}}^{\mathrm{eq}}(f_m^k,\theta^k,p^k) \right) \\ \mu_{\mathrm{ISO}}^{\mathrm{ineq},k} &= \max\left(0, \mu_{\mathrm{ISO}}^{\mathrm{ineq},k-1} + \alpha_k \left(f_{\mathrm{ISO}}^{\mathrm{ineq}}(f_m^k,\theta^k,p^k) \right)\right)\end{aligned}$$

for an appropriately chosen step size α_k (see Lemma 5.4.2 below), and the TSO is informed of the new proposed price of electricity (i.e., the new value of the sum of the Lagrange multipliers).

Note that the ISO only needs to know the TSO's proposed charging schedule to compute $f_{\mathrm{ISO}}^{\mathrm{eq}}(f_m^k,\theta^k,p^k)$ and $f_{\mathrm{ISO}}^{\mathrm{ineq}}(f_m^k,\theta^k,p^k)$; in particular, the TSO does not need to disclose the customers' demand or the planned vehicle routes. Conversely, the ISO only needs to inform the TSO of the proposed price of electricity: the generation costs and the power demands remain private.

The full dual decomposition algorithm is reported in Algorithm 3.

The next lemma proves that, if the step size α_k is "small enough", the algorithm converges.

Algorithm 3: Dual decomposition distributed algorithm for the P-AMoD problem

$k \leftarrow 1$
ISO sets $\lambda_{\text{ISO}}^{\text{eq},0}, \mu_{\text{ISO}}^{\text{ineq},0} \leftarrow$ dual solution to Problem (5.7) with $\{d_l\} = \{d_{l,e}\}$.
repeat
ISO informs TSO of $\lambda_{\text{ISO}}^{\text{eq},k-1} + \mu_{\text{ISO}}^{\text{ineq},k-1}$
TSO sets $\{f_m^k, \lambda_m^{c,\text{in},k}, \lambda_m^{t,c,\text{out},k}, N_F^k\}$ $\leftarrow$solution to Problem (5.6) with $p_{(\mathbf{v},\mathbf{w})} = J_C\left(\lambda_{\text{ISO}}^{\text{eq},k-1} + \mu_{\text{ISO}}^{\text{ineq},k-1}\right)$
ISO sets $\{\theta^k, p^k\} \leftarrow$ solution to Problem (5.16)
TSO informs ISO of proposed charging schedule f_m^k.
ISO updates $\lambda_{\text{ISO}}^{\text{eq},k} \leftarrow \lambda_{\text{ISO}}^{\text{eq},k-1} + \alpha_k f_{\text{ISO}}^{\text{eq}}(f_m^k, \theta^k, p^k)$
$\mu_{\text{ISO}}^{\text{ineq},k} \leftarrow \max(0, \mu_{\text{ISO}}^{\text{ineq},k-1} + \alpha_k f_{\text{ISO}}^{\text{ineq}}(f_m^k, \theta^k, p^k))$
$k \leftarrow k+1$
until $\|\lambda_{\text{ISO}}^{\text{eq},k} - \lambda_{\text{ISO}}^{\text{eq},k-1}\|^2 + \|\mu_{\text{ISO}}^{\text{ineq},k} - \mu_{\text{ISO}}^{\text{ineq},k-1}\|^2 \leq \varepsilon^2$

Lemma 5.4.2 (Convergence of Algorithm 3)**.** *If the step size α_k is chosen so that*

$$\begin{aligned} 0 < \alpha^k < & 2\,[-\left(V_T T_M(f_m^\star) + V_D D_v(f_m^\star) + V_B(f_m^\star) + C_G(p^\star)\right) \\ & + \left(V_T T_M(f_m^k) + V_D D_v(f_m^k) + V_B(f_m^k) + C_G(p^k)\right)] \\ & / \left(\|f_{\text{ISO}}^{\text{eq}}(f_m^k, \theta^k, p^k)\|^2 + \|f_{\text{ISO}}^{\text{ineq}}(f_m^k, \theta^k, p^k)\|^2\right), \end{aligned} \tag{5.17}$$

then Algorithm 3 converges to the optimal solution to Problem (5.9)*.*

Proof. The proof follows immediately from Proposition 6.3.1 in [Bertsekas, 1999]. □

Note that the optimal value of Problem (5.9), $(V_T T_M(f_m^\star) + V_D D_v(f_m^\star) + V_B(f_m^\star) + C_G(p^\star))$, is not known. The optimal value of the dual to (5.9) at step k provides a lower bound to the optimal value of the problem — however, neither the TSO nor the ISO can compute such value without access to the other's private information. In practical applications, a small, fixed α^k and an appropriate stopping criterion are used to ensure convergence.

5.5 Numerical Experiments

We study a hypothetical deployment of an AMoD system to satisfy medium-distance commuting needs in the Dallas-Fort Worth metroplex, with the primary objective of investigating the interaction between such system and the Texas power network. Specifically, we study a ten-hour interval corresponding to one commuting cycle, from 5 a.m. to 3 p.m., with 30-minute resolution. Data on commuting patterns is collected from the Census Transportation Planning Products (CTPP) 2006-2010 Census Tract Flows, based on the American Communities Survey (ACS) [Federal Highway Administration, 2014]. Departure times are gathered from ACS data

[United States Census Bureau, 2017]. Census tracts in the metroplex are aggregated in 25 clusters, as shown in Figure 5.3. We only consider trips starting and ending in different clusters: the total number of customer requests is 400,532. The commuters' value of time is set equal to \$24.40/hr, in accordance with DOT guidelines [U.S. Dept. of Transportation, 2015].

The road network, the road capacities, and the travel times are obtained from OpenStreetMap data [Haklay and Weber, 2008, Boeing, 2017] and simplified. The resulting road network, containing 25 nodes and 147 road links, is shown in Figure 5.3.

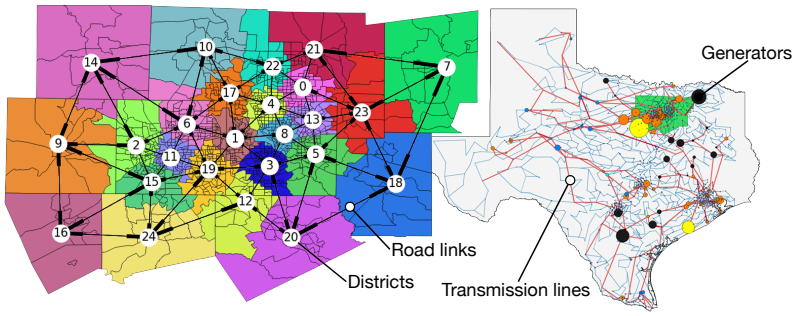

Figure 5.3: Left: Census tracts and simplified road network for Dallas-Fort Worth. Right: Texas power network model (from [Illinois Center for a Smarter Electric Grid (ICSEG), 2016]). The capacity of each edge equals the overall capacity of roads connecting the start and end clusters. The travel time between two nodes is the minimal travel time between the centroids of the corresponding clusters.

The battery capacity and power consumption of the EVs are modeled after the 2017 Chevrolet Bolt [Chevrolet, 2017]. The cost of operation of the vehicles, excluding electricity costs, is 48.6c/mile, in accordance with DOT guidelines [Bureau of Transportation Statistics, 2016]. The fleet consists of 150,000 vehicles, i.e. 1 AMoD vehicle for every 2.67 customers, similar to the 2.6 ratio in [Spieser et al., 2014]. To represent the possibility that vehicles might not begin the day fully charged, each EV starts the day with a 50% battery charge and is required to have the same level of charge at the end of the simulation.

We adopt a synthetic model of the Texas power network provided in [Illinois Center for a Smarter Electric Grid (ICSEG), 2016] and portrayed in Figure 5.3. The model provided does not contain power generation costs: we labeled each generator according to its source of power and assigned generation costs according to U.S. Energy Information Administration estimates [EIA, 2017]. Furthermore, the model is time-invariant; to model the time evolution of power loads and the availability of solar and wind power we used historical data from ERCOT, Texas's ISO [Electric Reliability Council of Texas (ERCOT), 2017], and we imposed ramp-up and ramp-down constraints of 10%/hr and 40%/hr on the generation capability of nuclear and coal power plants, respectively.

We compare the results of three simulation studies. In the baseline simulation study, no electric

vehicles are present: we consider the power network *in isolation* subject only to exogenous loads. In the P-AMoD simulation study, we solve Problem (5.11), which embodies the cooperation between the TSO and the ISO. Finally, in the uncoordinated simulation study, we first solve the TSO's Vehicle Routing and Charging problem with *fixed* electricity prices obtained from the baseline simulation study; we then compute the load on the power network resulting from the vehicles' charging and discharging, and solve the ISO's Economic Dispatch problem with the updated loads. The uncoordinated simulation study captures the scenario where the TSO attempts to minimize its passengers' cost while disregarding the coupling with the power network.

Table 5.1 and Figure 5.4 show the results.

	Baseline	P-AMoD	Uncoord.
Avg. cust. travel time [h]	-	1.2532	1.2532
Tot. energy demand [GWh]	517.498	520.541	520.979
Tot. electricity expenditure [k$]	39,604.71	39,264.84	39,629.50
w.r.t. baseline [k$]		-339.87	+24.79
Avg. price in DFW [$/MW]	78.68	75.79	77.47
TSO tot. elec. expenditure [k$]	-	227.98	296.82

Table 5.1: P-AMoD simulation results (one commuting cycle, 10 hours).

The quality of service experienced by TSO customers, measured by the average travel time, is virtually identical in the P-AMoD and in the uncoordinated case. The energy demand of the AMoD system is also very similar in both cases. On the other hand, the effect of coordination on the overall electricity expenditure is noticeable. Coordination between the TSO and the ISO causes a *reduction* in the total expenditure for electricity of $339,870 per commuting cycle compared to the baseline case, despite the increased demand! In other words, a P-AMoD system allows a TSO to deliver on-demand transportation without an increase in overall electricity expenditure – a remarkable, and perhaps surprising, finding. Instead, in the uncoordinated case, the total expenditure for electricity is *increased* by $24,790. This corresponds to a difference of $364,066 between the P-AMoD case and the uncoordinated case, which compounds to savings in electricity expenditure of $182M per year (assuming two commuting cycles per day and 250 work days per year).

Who benefits from the reduction in energy expenditure? From the last two rows in Table 5.1, one can see that the average price of electricity in the P-AMoD case is 2.16% lower than in the uncoordinated case in Dallas-Fort Worth (corresponding to savings of $122.3M/year). The energy expenditure of the TSO in the P-AMoD case is 23.5% lower than in the uncoordinated case (a saving of $69,740 per commuting cycle, corresponding to close to $35M/year). Finally, electricity customers outside of Dallas experience a small reduction of 0.75% in their energy expenditure. Thus, the majority of the benefits of coordination are reaped by customers of the power network in the region where the AMoD system is deployed; the TSO also benefits from a noticeable reduction in its electricity expenditure.

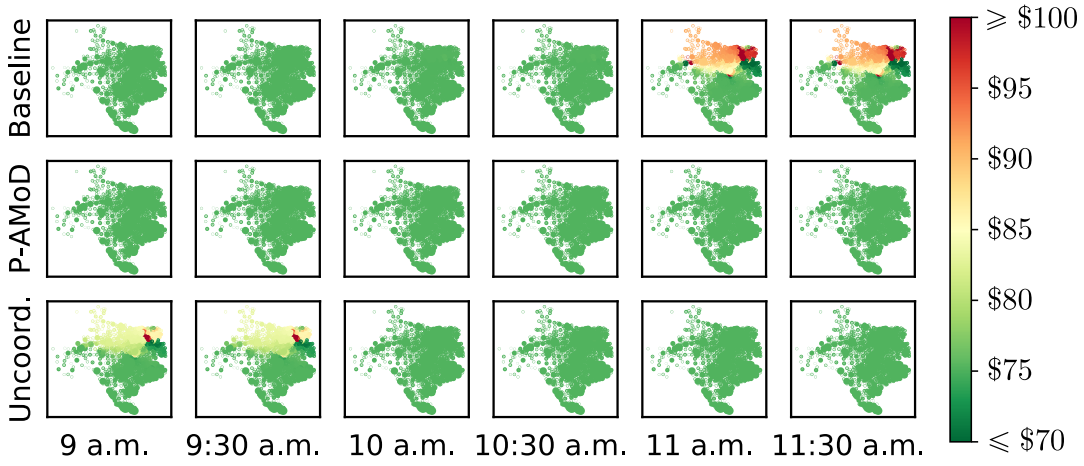

Figure 5.4: LMPs in Texas between 9 a.m. and 11:30 a.m. The presence of the AMoD fleet can reduce locational marginal prices; coordination between the TSO and the ISO can yield a further reduction.

Figure 5.4 shows this phenomenon in detail. The presence of the AMoD system results in a decrease in the LMPs with respect to the baseline case (11-11:30 a.m.). As electricity prices increase, empty vehicles travel to carefully chosen stations to sell their stored energy back to the network: this results in reduced congestion and lower prices in the power network, even in the absence of coordination. Crucially, coordination between the TSO and the ISO can result in further decreases in the price of electricity with respect to the uncoordinated case (9-9:30 a.m.), significantly curtailing the impact of the AMoD system on the power network. By leveraging their battery capacities and acting as mobile storage units, the EVs are able to reduce congestion in the power transmission network: this results in lower LMPs in the Dallas-Fort Worth region, and hence lower electricity expenditure. Simulations were carried out on commodity hardware (Intel Core i7-5960, 64 GB RAM) and used the MOSEK LP solver. The simulations required 3,923s for the P-AMoD scenario, 2,885s for the uncoordinated scenario, and 4.55s for the baseline scenario. While such computation times could be improved by using high-performance computational hardware, in the next section we present a receding-horizon algorithm for P-AMoD which, in addition to the intrinsic robustness benefits of closed-loop control, can be solved in seconds on commodity hardware.

5.6 A Receding-Horizon Algorithm for P-AMoD

Leveraging the structural insights from the network flow optimization problem of the previous sections, along with a few mild assumptions, we next devise a receding-horizon algorithm that is robust to the uncertainty in future demand. Additionally, this algorithm trades off some suboptimality, which we characterize with simulations, for very fast computation times.

To reduce the computational complexity of the optimization problem, we *decouple* the customer routing process from the P-AMoD optimization. The key assumption is that customer-carrying trips follow pre-computed routes and are never interrupted by a charging/discharging event. Formally, customer trips from node $i \in \mathcal{V}_R$ to node $j \in \mathcal{V}_R$ follow a fixed route $r_{i\to j}$ with an associated travel time of $t_{i\to j}$ and a required charge of $c_{i\to j}$. Thus, customer flows $\{f_{B,d_B}(\mathbf{u},\mathbf{v})\}_{(\mathbf{u},\mathbf{v}),d_B}$ are no longer part of the optimization variables and Equation (5.10a) is redundant. However, the initial and final charge of the customer-carrying vehicles $\{\lambda_m^{c,\text{in}}\}$ and $\{\lambda_m^{t,c,\text{out}}\}$ remain optimization variables. The following constraint ensures that charge is conserved along customer routes, that is, vehicles traveling from i to j and departing at time t at charge level c arrive at time $t + t_{i\to j}$ with charge $c - c_{i\to j}$:

$$\lambda_m^{t,c,\text{out}} = \begin{cases} \lambda_m^{c+c_{v_m\to w_m},\text{in}} & \text{if } t_m = t - t_{v_m\to w_m} \\ 0 & \text{otherwise} \end{cases} \tag{5.18}$$

for all $t \in \{1,\dots,T\}, c \in \{1,\dots,C\}, m \in \{1,\dots,M\}$.

The cost function is also modified to remove the customers' travel times, and road congestion constraints are adjusted to account for the traffic induced by customer-carrying vehicles. Specifically, the congestion induced by customer-carrying vehicles is *fixed* for given customer demand, since customers follow pre-defined routes. We denote the residual capacity of a road link $(v_\mathbf{v}, v_\mathbf{w}) \in \mathcal{E}_R$ at time t (i.e., the capacity of the road link one customer-carrying trips are accounted for) as $\overline{f}_{(v_\mathbf{v},v_\mathbf{w}),t}$. The congestion constraints on road links (5.3) becomes

$$\sum_{c_\mathbf{v}=1}^{C} f_0(\mathbf{v},\mathbf{w}) \le \overline{f}_{(v_\mathbf{v},v_\mathbf{w}),t_\mathbf{v}}, \qquad \forall (v_\mathbf{v}, v_\mathbf{w}) \in \mathcal{E}_R, t_\mathbf{v} \in \{1,\dots,T\}. \tag{5.19}$$

Only rebalancing vehicles traverse the charging and discharging links: thus, the capacity constraint of the charging stations (5.4) and the coupling equation (5.8) are rewritten as

$$\sum_{\substack{(\mathbf{v},\mathbf{w})\in\mathcal{E}_S:\\ v_\mathbf{v}=v_\mathbf{w}=v}} f_0(\mathbf{v},\mathbf{w}) \le \overline{S}_{v_\mathbf{v}}, \qquad \forall v \in \mathcal{S}, t \in \{1,\dots,T\}. \tag{5.20}$$

$$d_l(t) = d_{l,e}(t) + J_C \delta c^+_{\mathcal{M}_{\text{P,R}}(l)} \sum_{\substack{(\mathbf{v},\mathbf{w})\in\\ M^+_{P,G}(l,t)}} f_0(\mathbf{v},\mathbf{w}) + J_C \delta c^-_{\mathcal{M}_{\text{P,R}}(l)} \sum_{\substack{(\mathbf{v},\mathbf{w})\in\\ M^-_{P,G}(l,t)}} f_0(\mathbf{v},\mathbf{w}), \forall l \in \mathcal{L}, t \in \{1,\dots,T\}. \tag{5.21}$$

In order to adapt the problem for use in a receding-horizon implementation, several further modifications are required.

Outstanding customers

In Problem (5.9), future customer demand is assumed to be perfectly known – conversely, in a real implementation, unforeseen transportation requests may be submitted at any time. As a result, some customers may not be assigned to a vehicle when they arrive at their departure node.

We denote the set of such waiting customers as *outstanding requests*. Outstanding requests are assumed to wait at the departure station until a suitable vehicle is available. The departure time of the outstanding request is an optimization variable; the goal is to service such requests as quickly as possible.

Formally, outstanding requests are characterized as the set of M^o tuples $\{(v_{m^o}, w_{m^o}, \lambda_{m^o})\}_{m^o=1}^{M^o}$, where $v_{m^o} \in V_R$ is the outstanding request's origin location, $w_{m^o} \in \mathcal{V}_R$ is the outstanding request's destination location, and λ_{m^o} is the average arrival rate (i.e. the number of outstanding customers divided by the duration of one time step).

For each outstanding request $m^o \in [1, \dots, M^o]$, the set of variables $\{\lambda_{m^o}^{t,c,\text{in}}\}_{c,t}$ denotes the customer rate departing at time t at charge level c; in analogy with customer requests, the set of variables $\{\lambda_{m^o}^{t,c,\text{out}}\}_{t,c}$ denotes the customer rate reaching the destination at time t with charge level c. Both are optimization variables.

The following constraints ensure that outstanding requests are serviced within the optimization horizon, in analogy with Eq. (5.10b), (5.10c) and (5.18) for regular customers:

$$\sum_{t=1}^{T}\sum_{c=1}^{C} \lambda_{m^o}^{t,c,\text{in}} = \lambda_{m^o}, \qquad \forall m^o \in \{1, \dots, M^o\}, \tag{5.22a}$$

$$\sum_{t=1}^{T}\sum_{c=1}^{C} \lambda_{m^o}^{t,c,\text{out}} = \lambda_{m^o}, \qquad \forall m^o \in \{1, \dots, M^o\}, \tag{5.22b}$$

$$\lambda_{m^o}^{t,c,\text{out}} = \lambda_{m^o}^{t-t_{v_{m^o}\to w_{m^o}},\, c+c_{v_{m^o}\to w_{m^o}},\text{in}} \qquad \forall t \in \{1, \dots, T\}, c \in \{1, \dots, C\}, \; m^o \in \{1, \dots, M^o\}. \tag{5.22c}$$

The overall wait time of oustanding customers can then be computed as

$$T_M^o = \sum_{m^o=1}^{M^o} t \sum_{c=1}^{C} \lambda_{m^o}^{t,c,\text{in}}$$

Vehicle end charge

In order to achieve good closed-loop performance and trade off between servicing present demand and ensuring vehicles are available for future customers, the final charge level of rebalancing vehicles

is constrained to be higher than a given threshold $\underline{C}^T$:

$$f_0(\mathbf{v}, \mathbf{w}) = 0 \quad \forall (\mathbf{v}, \mathbf{w}) \in \mathcal{E} : c_{\mathbf{w}} \leq \underline{C}^T, t_{\mathbf{w}} = T \tag{5.23}$$

Feasibility

Problem (5.9) is not guaranteed to admit a solution for arbitrary transportation requests and arbitrary numbers of vehicles. To ensure persistent feasibility of the receding-horizon controller, slack variables (associated with a high cost) are introduced in Equations (5.10b), (5.10c), (5.22a), and (5.22b), allowing customer requests to be dropped to preserve feasibility. As a result, so long as the Economic Dispatch problem is feasible, the P-AMoD problem always admits a feasible solution where no customers are transported and no vehicle moves, charges, or discharges, ensuring persistent feasibility.

Fractional output

The output of Problem (5.9) is, in general, fractional: therefore it can not directly be used for control of a P-AMoD system. To overcome this, control actions are computed by sampling the first time step of the fractional optimal solution to the problem in a receding-horizon framework. For each customer request, we sample the charge levels of the vehicles assigned to the request; charging/discharging and rebalancing actions for customer-empty vehicles are also sampled. In detail,

- *Customer requests*: for each customer request m departing at time $t = 1$, the charge level of the customer-carrying vehicles that will be assigned to the request is selected by drawing $\lceil \lambda_m \rceil$ samples (that is, one sample per customer) according to the distribution of $\{\lambda_m^{c,\text{in}}\}_c$. Analogously, the departure time and charge level of vehicles assigned to outstanding customer request m^o are computed by drawing $\lceil \lambda_{m^o} \rceil$ samples according to the distribution of $\{\lambda_{m^o}^{t,c,\text{in}}\}_{t,c}$.

  Customers are then assigned to vehicles with a charge level corresponding to the sampled charge level. If no vehicles at the chosen charge level are available, a fall-back strategy is adopted where customers are assigned to the closest vehicle with charge sufficient to complete the trip.

- *Idle vehicles*: charging, discharging, and rebalancing actions are sampled from the distribution of the rebalancing flow $\{f_0(\mathbf{v}, \mathbf{w})\}_{(\mathbf{v},\mathbf{w})}$. Specifically, for each node $v \in \mathcal{V}_R$ and each charge level $c \in \{1, \ldots, C\}$, a number of edges $(\mathbf{v}, \mathbf{w})$ corresponding to the number of vehicles charging, discharging, or rebalancing at node v at charge c and time $t = 1$ are sampled according the distribution of $f_0(\mathbf{v}, \mathbf{w}) : v_{\mathbf{v}} = v, t_{\mathbf{v}} = 1, c_{\mathbf{v}} = c$. If the sampling procedure selects an edge corresponding to a charging link, a charging task is chosen; if an an edge corresponding to a discharging link is sampled, a discharging task is chosen; if an edge corresponding to a road

link is sampled, a rebalancing task to the destination of the road link is chosen. The chosen tasks are then assigned to idle vehicles.

The sampling procedure for customer requests and idle vehicles is detailed in the RHController procedure in Algorithm 4.

Algorithm 4: Real-time receding-horizon algorithm for the P-AMoD problem

procedure RHController(Customer requests, vehicle states)
  $\{f_0, \lambda_m^{c,\text{in}}, \lambda_m^{t,c,\text{out}}, \lambda_{m^o}^{t,c,\text{in}}, \lambda_{m^o}^{t,c,\text{out}}\} \leftarrow$ Solve Problem (5.24)
  for all customer request m with $t_m = 1$ **do**
    CustCharge$(v_m, w_m) \leftarrow$Sample $\lceil \lambda_m \rceil$ charge levels from $\{\lambda_m^{c,\text{in}}\}_c$
  for all outstanding customer request m^o **do**
    DepartureTimes$(m^o) \leftarrow$Sample $\lceil \lambda_m \rceil$ departure times from $\{\lambda_{m^o}^{t,c,\text{in}}\}_t$
    for all DepartureTime $\in$ DepartureTimes(m^o) **do**
      if DepartureTime $= 1$ **then**
        $c_o \leftarrow$ Sample charge level from $\{\lambda_{m_o}^{t,c,\text{in}}\}_c$
        CustCharge$(v_{m^o}, w_{m^o}) \leftarrow$ append c_o
  for all node $v \in \mathcal{V}_R$ **do**
    for all charge level $c \in [1, \dots, C]$ **do**
      $r \leftarrow \sum_{(\mathbf{v},\mathbf{w}):v_\mathbf{v}=v,t_\mathbf{v}=1,c_\mathbf{v}=c,} f_0(\mathbf{v}, \mathbf{w})$
      for $a = 1, \dots, r$ **do**
        $(\mathbf{v}, \mathbf{w}) \leftarrow$ Sample edge according to distribution of $f_0(\mathbf{v}, \mathbf{w})$ with $\mathbf{v}$ s.t. $v_\mathbf{v} = v, t_\mathbf{v} = 1, c_\mathbf{v} = c$
        if $(\mathbf{v}, \mathbf{w})$ is a charging link **then**
          Task $\leftarrow$ Charge at v
        else if $(\mathbf{v}, \mathbf{w})$ is a discharging link **then**
          Task $\leftarrow$ Discharge at v
        else if $(\mathbf{v}, \mathbf{w})$ is a road link **then**
          Task $\leftarrow$ Rebalance from $v_\mathbf{v}$ to $v_\mathbf{w}$
        IdleTasks$(v, c) \leftarrow$ Append Task
  return CustCharge, OutCustCharge, IdleTasks

5.6.1 A receding-horizon controller

We are now in a position to present the receding-horizon controller.

We denote the distance traveled by the rebalancing vehicles as

$$D_V^0 = \sum_{(\mathbf{v},\mathbf{w})\in\mathcal{E}} d_{v_\mathbf{v},v_\mathbf{w}} f_0(\mathbf{v}, \mathbf{w}),$$

and the depreciation of the rebalancing vehicles' batteries as

$$V_B^0 = \sum_{(\mathbf{v},\mathbf{w})\in\mathcal{E}_S} f_0(\mathbf{v}, \mathbf{w}) |\delta c_{v_\mathbf{v}}| d_B.$$

We define the *receding-horizon* suboptimal P-AMoD problem as

$$\begin{aligned} \underset{\substack{f_0,\lambda_m^{c,\text{in}},\lambda_m^{t,c,\text{out}},\\ \lambda_{m^o}^{t,c,\text{in}},\lambda_{m^o}^{t,c,\text{out}},N_F,\theta,p}}{\text{minimize}} \quad & T_M^o + V_D D_V^0 + V_B^0 + C_G \\ \text{subject to} \quad & (5.10b), (5.10c), (5.18), (5.2), (5.22), (5.19), (5.20), (5.23), (5.7), \text{ and } (5.21). \end{aligned} \tag{5.24}$$

Problem (5.24) has $O(|\mathcal{E}| + MC + |\mathcal{V}_R|C + T(|\mathcal{G}| + |\mathcal{E}_p| + |\mathcal{B}|))$ variables: compared to Problem (5.11), the problem size does not depend on the product of $|\mathcal{E}|$ and $|\mathcal{V}_R|$, resulting in an order-of-magnitude reduction in the overall number of required variables for prototypical problems.

We assess the performance of the receding-horizon P-AMoD controller with an agent-based simulation based on the same case study considered in the previous section. The receding-horizon problem is solved every two minutes with a 2-hour lookahead and a 30-minute time step. The performance of the algorithm is compared with an uncoordinated receding-horizon controller that optimizes the AMoD system's operations under the assumption that electricity prices stay constant. In order to characterize the randomized nature of the controller, ten trials are performed for each case. Table 5.2 shows the results.

	P-AMoD	Greedy
Avg. cust. travel time	36'30"	36'47"
Tot. electricity expenditure [k$]	40,434.17	40,583.98
Avg. price in DFW [$/MW]	81.93	82.61
TSO tot. elec. expenditure [k$]	229.05	260.89

Table 5.2: Real-time power-in-the-loop algorithm simulation results (one commuting cycle, 10 hours). Average over ten realizations.

Coordination results in savings of approximately $150,000 per commuting cycle (corresponding to almost $75M/year) with respect to the uncoordinated algorithm. In particular, cooperation between the TSO and the ISO results in 0.88% lower electricity prices in the Dallas Fort-Worth area. The TSO's expense in the coordinated case is comparable with the expense computed in the *Numerical Experiments*, which represents an *upper bound* on the performance of a receding-horizon P-AMoD controller. On the other hand, the average price of electricity in Texas increases by 1.7% compared to the baseline case studied in the *Numerical Experiments*. This is not unexpected, as we use a short 2-hour lookahead. An important direction of future research is to perform a detailed sensitivity analysis, and in particular to explore how the lookahead time affects the tradeoff between computational complexity, economic savings, and robustness of the algorithm to inaccuracies in demand forecasting.

The receding-horizon P-AMoD problem was solved in an average of 2.09s; thus, the algorithm is amenable to closed-loop control of large-scale systems.

5.7 Conclusions and Future Work

In this chapter we studied the interaction between an AMoD system and the electric power network. The model we proposed subsumes earlier models for AMoD systems and for the power network; critically, it captures the coupling between the two systems and allows for their *joint optimization*. We also proposed a numerical procedure to losslessly reduce the dimensionality of the P-AMoD optimization problem (Eq. (5.11)), making realistic problems amenable to efficient numerical optimization on commodity hardware. We showed that the jointly optimal solution to the P-AMoD problem is a general equilibrium, and we proposed a distributed privacy-preserving algorithm that allows agents to compute such equilibrium without sharing private information about customer requests, generation costs, or power demands. We applied our model and algorithms to a case study of an AMoD deployment in Dallas-Fort Worth, TX. The case study showed that coordination between the TSO and the ISO can result in a *reduction* in the overall electricity expenditure (despite the increase in demand), while having a negligible impact on the TSO's quality of service. Finally, we presented a receding-horizon algorithm for P-AMoD that delivers computation times in the order of seconds on commodity hardware and provides built-in robustness to uncertainty in future demand at the price of some suboptimality.

This work opens multiple avenues of research. First, we plan to capture the impact of cooperation between the TSO and the ISO on the power *distribution* network by incorporating convex optimal power flow models. Second, we will extend the AMoD model to capture other modes of provision of service, including heterogeneous fleets where vehicles may differ in size, seating capacity, and battery capacity, and ride-pooling mechanisms where multiple customers with similar origins and destinations can travel in the same vehicle. Third, the model of the power network considered in this work does not capture ancillary services such as regulation and spinning reserves. We will extend our model to capture those and evaluate the feasibility of using coordinated fleets of EVs to aid in short-term control of the power network. Fourth, we plan to expand the breadth of our case studies to include scenarios where the area of service of the TSO and ISO are similar. In particular, we will study the effect of deployment of AMoD systems in multiple cities in Texas on the state's power network, as well as smaller, isolated power network, e.g. Hawaii's. Finally, we wish to explore the effect of TSO-ISO coordination on penetration of renewable energy sources, and to determine whether large-scale deployment of AMoD systems can increase the fraction of renewable power sources in the generation power mix.

Chapter 6

Conclusions

6.1 Summary

AMoD systems can reshape the fabric of our cities, with positive effect on safety, access to mobility, and demand for parking infrastructure. However, the impact that these systems will have on traffic congestion and the electric power network remains an active area of research.

AMoD systems will increase vehicle-miles traveled compared to private cars due to empty "rebalancing" trips. If these trips take place on already-congested road, they could further increase traffic congestion. On the other hand, centralized control of AMoD fleets enables *coordinated, congestion-aware* routing strategies that can minimize the impact of empty trips on traffic and reduce the travel times of customer-carrying and empty vehicles alike. We propose a network flow model for AMoD systems in congested road network and rigorously prove that, under mild assumptions that are substantially satisfied in major U.S. cities, it is always possible to replace personal vehicles with an AMoD system without causing an increase in congestion for a given level of customer demand. This analytical insight inspires two algorithms for congestion-aware control of AMoD systems. First, we propose a receding-horizon routing algorithm for *rebalancing* vehicles; microscopic simulations based on real taxi data in Manhattan show that the algorithm provides superior performance (in terms of customer wait times and congestion) compared to state-of-the-art congestion-agnostic algorithms. Next, we turn our attention to customer-carrying vehicles: we further leverage the network flow model to design a randomized congestion-aware routing policy for customer-carrying as well as rebalancing vehicles, and show that the algorithm holds promise to further reduce wait times and road congestion.

AMoD systems are uniquely positioned to drive mass adoption of electric vehicles by mitigating the downsides of limited EV range through fleet-wide customer assignment and charging policies. However, large-scale adoption of EVs could put significant strain on the power network, causing large increases in power prices and network instability. To mitigate this, policies for control of AMoD

systems that account for the interaction with the power network (denoted as Power-in-the-loop AMoD) are needed. To this end, we design a network flow model for AMoD systems that captures the interaction with the electric power network and the time-varying nature of the joint system; we also develop algorithmic tools to losslessly reduce the size of the resulting model, making it amenable to efficient numerical optimization. Socially-optimal policies for joint control of AMoD systems and the power network require cooperation between AMoD operators and power network operators: we show that such cooperation can be enforced as a general economic equilibrium with appropriate pricing and that the equilibrium prices can be computed with a distributed privacy-preserving mechanism. Simulation results on a case study in Dallas-Fort Worth show that coordination can result in savings of hundreds of millions of dollars per year for power network customers and tens of millions per year for the AMoD operator; a receding-horizon algorithm is able to realize many of these gains in a real-time setting, trading off optimality for fast computation times.

All in all, these results show that, far from being a threat to our cities, AMoD systems can not only deliver massive benefits in terms of access to mobility and better land usage, but also deliver (i) virtually no negative impact to urban congestion and (ii) very positive impacts on the electric power network.

6.2 Contribution

Specifically, this thesis makes the following contributions:

Models We devise novel network flow models that capture the interaction between AMoD systems, urban congestion, and the electric power network. In particular, we develop a time-invariant network flow model that captures the impact of AMoD on urban congestion (Section 3.2). We then extend the model to also capture time-varying customer demand, the state of charge of autonomous electric vehicles, and, crucially, their interaction with the power network (Section 5.2). These models are amenable to efficient optimization and incorporate a number of operational constraints relevant to AMoD systems; as discussed in the next section, they can be readily extended to accommodate alternative cost functions and additional operational constraints. We leverage the structure of the AMoD models to reduce the number of variables in the problem with no loss in accuracy, thus enabling the efficient optimization of large-scale problems (Section 5.3).

Structural properties of AMoD We explore the structural properties of the network flow models to gain insight into the fundamental limitations of the congestion-aware AMoD problem. Doing so, we prove a (perhaps surprising) existential result with significant policy implications: under mild assumptions, for a given level of transportation demand, AMoD systems can replace private vehicles with *no increase in congestion* (despite the increase in vehicle-miles traveled) if empty vehicles are properly routed. We verify that the assumptions underlying this result are substantially verified

for major U.S. cities, and leverage the existential result to design efficient congestion-aware control algorithms.

Efficient control algorithms We leverage the network flow models to design efficient control algorithms for operational control of AMoD systems. In particular, we design two model-predictive control algorithms and show that they are amenable to a real-time implementation:

- In Section 3.4, we propose a receding-horizon, congestion-aware routing algorithm for rebalancing vehicles. The algorithm exploits the totally unimodular structure of the rebalancing problem to compute congestion-free routes for empty vehicles. Its execution time on a prototypical case study in Manhattan is under 0.5s.

- In Section 5.6, we design a receding-horizon control algorithm for Power-in-the-loop AMoD systems. The algorithm computes routes and charging schedules for rebalancing vehicles in an AMoD system and accounts for their interaction with the electric power network with the goal of reducing generation costs (and, indirectly, electricity prices). By leveraging the models presented in Section 5.2 and simplifications that trade a small amount of optimality for performance, it achieves computation times of approximately 2s on a prototypical case study in Dallas-Fort Worth, TX.

We also leverage randomized routing schemes to design congestion-aware routing algorithms for customer-carrying as well as rebalancing vehicles, and rigorously prove that the resulting solution has a very small degree of suboptimality (Section 4.2). A receding-horizon implementation is beyond the scope of this work: however, preliminary results on a case study in Manhattan suggest that such an implementation could reduce customer service times by an additional 5%.

Distributed pricing mechanisms for P-AMoD We rigorously show that the socially optimal solution to the P-AMoD problem is also a general equilibrium for self-interested AMoD operators and power generator operators if the price of electricity is set by the ISO according to locational marginal pricing. Computing locational marginal prices requires the ISO to have access to private information from both power generators and the AMoD system: while generator operators routinely share such data with the ISO, it is unlikely that an AMoD operator would be willing to share information on customer demand with a third party. To overcome this difficulty, in Section 5.4 we propose a dual decomposition algorithm that the AMoD operator and the ISO can use to compute market-clearing prices in a distributed fashion and without disclosing private information, extending previous results in [Alizadeh et al., 2017].

Case studies We assess the impact of congestion-aware AMoD and power-in-the-loop AMoD through two high-fidelity case studies based on real-world data. In Section 3.5.3, we test the

congestion-aware routing algorithm through a 24-hour microscopic simulation based on 500 000 real taxi trips in Manhattan, and show that congestion-aware control of AMoD systems can result in a 22% decrease in service times for AMoD customers and lower congestion throughout the transportation network. In Section 5.5, we study the impact of P-AMoD on a hypothetical deployment of an AMoD system in Dallas-Fort Worth. Our results show that coordination between the transportation and the power network can *reduce* electricity prices compared to the case where no vehicles are present, despite the increased demand; this results in savings of hundreds of millions of dollars per year shared between power network customers and the AMoD operator, with no adverse impact on service times for AMoD customers.

6.3 Future Directions

The potential impact of AMoD systems on the built environment is deep and multi-faceted. While this work focuses on exploring the interaction between AMoD systems, urban congestion, and the power network, the results in this thesis suggest a number of promising directions for future research.

Impact of AMoD systems on pollution The models developed in Chapter 5 focus on reducing the cost of electricity generation (and, indirectly, electricity prices) by harnessing the interaction between electric self-driving cars and the power network. However, a number of alternative optimization objectives can be readily incorporated in the model. The use of AMoD systems to reduce pollution is especially interesting. The interaction between AMoD systems and the power network can be harnessed to reduce the adverse effects of pollution by (i) shifting the geographic impact of pollution from urban roads to power plants (which are often located away from city centers) and (ii) enabling wider adoption of clean energy sources such as wind and solar using the vehicles' batteries as a buffer to counter intermittent availability of renewables. In order to capture the impact of pollution on human health, power plant emission models and microscopic air quality models (e.g. [Byun and Schere, 2006, US EPA Office of Research and Development, 2018a]) can be leveraged to assess the spatial impact of individual power plants on pollution. Population data and available epidemiological models (e.g. [US EPA Office of Research and Development, 2018b]) can then be used to model the impact of pollution on human health, capturing the effect of particulate, sulphur, and NOx emissions from individual plants on air quality in cities and the resulting impact on public health (see e.g. [Driscoll et al., 2015]). The public health effect of each individual power plant can be included in the P-AMoD model as a cost, with the goal of minimizing the overall impact of electric power generation and AMoD on pollution and the resulting health outcomes.

Interaction between AMoD systems and public transit Deployments of AMoD systems in our cities will coexist with public transportation, in particular subway and light rail. Will self-driving cars act as feeders for public transit (thus solving the last-mile problem and increasing ridership of

existing transit systems), or will they compete with it? The tools developed in this thesis can be readily extended to model multi-modal transportation by representing different transportation modes (e.g. AMoD, subways, buses, and walking) as different subgraphs; "interchange edges" connecting the subgraphs can model the customers' ability to switch modes of transportation. These models can be used to compute socially-optimal *multi-modal* routes that optimize goals including system throughput, customer travel times, and pollutant emissions.

Socially-optimal solutions are useful from a policy perspective but cannot be implemented directly: AMoD systems and public transit are generally managed by different entities, and customers select their routes according to their own private value of time and their price sensitivity. However, a socially-optimal solution can be enforced as a competitive equilibrium through appropriate pricing (similarly to the results in Section 5.4); the simple, quasi-analytical structure of network flow models can be leveraged to design road tolls and pricing schemes for AMoD systems and public transit which ensure that the decisions of selfish agents are aligned with the social optimum.

Population shifts, induced demand, and land value Increased access to mobility due to AMoD systems will induce major shifts in the accessibility and desirability of housing and workplace locations. This has the potential to drastically reshape our cities (akin to how private car ownership enabled urban sprawl in the U.S.). From a societal perspective, AMoD will cause large population shifts, enabling people to live and work in neighborhoods previously underserved by public transit; from an economic perspective, the shift in residential property values could top $1 trillion in the United States (and close to $280B in Los Angeles alone!) [Levin, 2018].

Network flow models do not directly capture the effect of AMoD on population shifts and induced transportation demand. However, they can readily be used to provide inputs to economic models for location choice and for land value (e.g. [Alonso, 1964, Capozza and Helsley, 1989]). These models, in turn, can provide estimates of the shifts in demand for transportation induced by AMoD, which can be fed back as inputs to the models developed in this thesis.

This multidisciplinary, closed-loop approach will enable policy-makers to assess the societal and economic impact of the *feedback loop* between AMoD, housing, and labor. Critically, it will allow to readily assess the impact of transportation, labor, and welfare policies, and enable the identification of policies which ensure that the benefits of AMoD systems are shared across society.

High fidelity simulations A number of open-source microscopic agent-based transportation simulators are available (e.g. MATSim [Horni et al., 2016] and SUMO [Krajzewicz et al., 2012]). These simulators faithfully capture multi-modal transportation, customer choice models, and urban congestion. However, existing simulators do not capture the operations of AMoD systems and, in particular, the vehicle rebalancing process; in addition, no existing simulator captures the feedback interaction between fleets of electric vehicles and the electric power network.

In Chapter 3 we briefly introduce an extension of the MATSim simulator that models the operations of AMoD systems. However, further development is needed to capture the interaction between transportation and the built environment, and in particular the power transmission and distribution network, in order to develop tools that allow the evaluation of the impact of AMoD on society with microscopic simulations at the regional scale.

Real-world implementation Limited deployments of AMoD systems with human supervision are already underway in several U.S. cities (e.g. Waymo in Phoenix, AZ [Waymo, 2018] and GM Cruise in San Francisco, CA [Davies, 2017]); more systems are expected to come online and become available to the general public in the next few years (e.g. GM Cruise [Etherington, 2017] and nuTonomy [Reuters, 2017]). We plan to infuse our algorithms in such systems through collaborations with industry partners. In addition, we are exploring the possibility of testing congestion-aware and charging-aware rebalancing policies on centrally-routed, *human-operated* car-sharing systems.

Real-world testing of the performance of our algorithms offers a number of benefits that cannot be realized through simulations alone:

- **Validation** Implementation on a real-world system will enable further validation of the assumptions made in the models proposed in this paper and an assessment of the robustness of our models and algorithms to violations of these assumptions.
- **Data collection** Real-world deployments will produce vast amounts of data on transportation demand, customer preferences, and exogenous traffic congestion. This information can be infused in the models proposed in this work, increasing their accuracy and predictive power.
- **Real-world benefits** The goal of this thesis is to build tools to manage the interaction between AMoD systems and the built environment: implementing the results of this work on real-world AMoD systems will allow us to achieve the ultimate goal of fostering the positive externalities of AMoD and avoiding the negative ones, thus fully realizing the benefits of AMoD systems in our cities.

Bibliography

[Ahuja et al., 1993] Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993). *Network Flows: Theory, Algorithms and Applications*. Prentice Hall.

[Alizadeh et al., 2017] Alizadeh, M., Wai, H.-T., Chowdhury, M., Goldsmith, A., Scaglione, A., and Javidi, T. (2017). Optimal pricing to manage electric vehicles in coupled power and transportation networks. *IEEE Transactions on Control of Network Systems*, 4(4):863–875.

[Alizadeh et al., 2014] Alizadeh, M., Wai, H.-T., Scaglione, A., Goldsmith, A., Fan, Y. Y., and Javidi, T. (2014). Optimized path planning for electric vehicle routing and charging. In *Allerton Conf. on Communications, Control and Computing*.

[Alonso, 1964] Alonso, W. (1964). *Location and Land Use*. Harvard Univ. Press.

[Balmer et al., 2009] Balmer, M., Rieser, M., Meister, K., Charypar, D., Lefebvre, N., and Nagel, K. (2009). MATSim-t: Architecture and simulation times. In *Multi-Agent Systems for Traffic and Transportation Engineering*, chapter 3.

[Barnard, 2016] Barnard, M. (2016). Autonomous cars likely to increase congestion. Available at `http://cleantechnica.com/2016/01/17/autonomous-cars-likely-increase-congestion`.

[Barth and Todd, 1999] Barth, M. and Todd, M. (1999). Simulation model performance analysis of a multiple station shared vehicle system. *Transportation Research Part C: Emerging Technologies*, 7(4):237–259.

[Berbeglia et al., 2010] Berbeglia, G., Cordeau, J.-F., and Laporte, G. (2010). Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202(1):8–15.

[Bertsekas, 1999] Bertsekas, D. (1999). *Nonlinear programming*. Athena Scientific, 2 edition.

[Boeing, 2017] Boeing, G. (2017). OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems*, 65:126–139.

[Borrelli et al., 2017] Borrelli, F., Bemporad, A., and Morari, M. (2017). *Predictive Control for Linear and Hybrid Systems*. Cambridge Univ. Press.

[Bureau of Public Roads, 1964] Bureau of Public Roads (1964). Traffic assignment manual. Technical report, U.S. Dept. of Commerce, Urban Planning Division.

[Bureau of Transportation Statistics, 2016] Bureau of Transportation Statistics (2016). National transportation statistics. Technical report, U.S. Dept. of Transportation.

[Byun and Schere, 2006] Byun, D. and Schere, K. (2006). Review of the governing equations, computational algorithms, and other components of the Model-3 Community Multiscale Air Quality (CMAQ) modeling system. *Applied Mechanics Reviews*, 59(2):51–77.

[Capozza and Helsley, 1989] Capozza, R. D. and Helsley, R. W. (1989). The fundamentals of land prices and urban growth. *Journal of Urban Economics*, 26(3):295–306.

[Chevrolet, 2017] Chevrolet (2017). Bolt EV. Available at `http://www.chevrolet.com/bolt-ev-electric-vehicle`. Retrieved on June 5, 2017.

[Cornuejols et al., 1990] Cornuejols, G., Nemhauser, G. L., and Wolsey, L. A. (1990). The uncapacitated facility location problem. In Mirchandani, P. B. and Francis, R. L., editors, *Discrete Location Theory*, pages 119–171. John Wiley & Sons.

[Daganzo, 1994] Daganzo, C. F. (1994). The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4):269–287.

[Davies, 2017] Davies, A. (2017). GM's robocar service drives employees around SF for free. Available at `https://www.wired.com/story/gm-cruise-anywhere-self-driving-san-francisco/`. Retrieved on March 2, 2018.

[Driscoll et al., 2015] Driscoll, C. T., Buonocore, J. J., Levy, J. I., Lambert, K. F., Burtraw, D., Reid, S. B., Fakhraei, H., and Schwartz, J. (2015). Us power plant carbon standards and clean air and health co-benefits. *Nature Climate Change*, 5:535—540.

[Dubhashi and Ranjan, 1996] Dubhashi, D. P. and Ranjan, D. (1996). Balls and bins: A study in negative dependence. *BRICS Report Series*, 3(25):1–27.

[EIA, 2017] EIA (2017). Levelized cost and levelized avoided cost of new generation resources in the annual energy outlook 2017. Technical report, U.S. Energy Information Administration.

[Electric Reliability Council of Texas (ERCOT), 2017] Electric Reliability Council of Texas (ERCOT) (2017). Grid information. Available at `http://www.ercot.com/gridinfo/`.

[Etherington, 2017] Etherington, D. (2017). GM and Cruise on track to field a self-driving ride hailing service by 2019. Available at `https://techcrunch.com/2017/11/30/`

gm-and-cruise-on-track-to-field-a-self-driving-ride-hailing-service-by-2019/. Retrieved on March 2, 2018.

[Evarts, 2013] Evarts, E. (2013). Many americans are just a plug away from owning an electric car. https://www.yahoo.com/news/many-americans-just-plug-away-owning-electric-car-160000286.html. Retrieved on March 13, 2018.

[Even et al., 1976] Even, S., Itai, A., and Shamir, A. (1976). On the complexity of timetable and multicommodity flow problems. *SIAM Journal on Computing*, 5(4):691–703.

[Fagnant and Kockelman, 2014] Fagnant, D. J. and Kockelman, K. M. (2014). The travel and environmental implications of shared autonomous vehicles, using agent-based model scenarios. *Transportation Research Part C: Emerging Technologies*, 40:1–13.

[Federal Highway Administration, 2014] Federal Highway Administration (2014). Census Transportation Planning Products (CTTP) 2006-2010 Census Tract Flows. Technical report, U.S. Dept. of Transportation.

[Ford and Fulkerson, 1962] Ford, L. R. and Fulkerson, D. R. (1962). *Flows in Networks*. Princeton Univ. Press.

[Glover et al., 2011] Glover, J., Sarma, M., and Overbye, T. (2011). *Power System Analysis and Design*. Cengage Learning, fifth edition.

[Goeke and Schneider, 2015] Goeke, D. and Schneider, M. (2015). Routing a mixed fleet of electric and conventional vehicles. *European Journal of Operational Research*, 245(1):81–99.

[Goldberg et al., 1998] Goldberg, A., Oldham, J., Plotkin, S., and Stein, C. (1998). An implementation of a combinatorial approximation algorithm for minimum-cost multicommodity flow. In *Int. Conf. on Integer Programming and Combinatorial Optimization*.

[Goldberg et al., 1990] Goldberg, A. V., Tardos, E., and Tarjan, R. E. (1990). Network flow algorithms. In *Algorithms and Combinatorics. Volume 9: Paths, Flows, and VLSI-Layout*. Springer-Verlag.

[Hadley and Tsvetkova, 2009] Hadley, S. W. and Tsvetkova, A. A. (2009). Potential impacts of plug-in hybrid electric vehicles on regional power generation. *The Electricity Journal*, 22(10):56–68.

[Haklay and Weber, 2008] Haklay, M. and Weber, P. (2008). OpenStreetMap: User-generated street maps. *IEEE Pervasive Computing*, 7(4):12–18.

[Hart et al., 1968] Hart, P. E., Nilsson, N. J., and Raphael, B. (1968). A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems, Science, & Cybernetics*, 4(2):100–107.

[Hogan, 1996] Hogan, W. W. (1996). Markets in real electric networks require reactive prices. In *Electricity Transmission Pricing and Technology*, chapter 7. Springer Netherlands, Dordrecht.

[Horni et al., 2016] Horni, A., Nagel, K., and Axhausen, K. W., editors (2016). *The Multi-Agent Transport Simulation MATSim*. Ubiquity Press.

[Iglesias et al., 2016] Iglesias, R., Rossi, F., Zhang, R., and Pavone, M. (2016). A BCMP network approach to modeling and controlling Autonomous Mobility-on-Demand systems. In *Workshop on Algorithmic Foundations of Robotics*.

[Iglesias et al., 2019] Iglesias, R., Rossi, F., Zhang, R., and Pavone, M. (2019). A BCMP network approach to modeling and controlling autonomous mobility-on-demand systems. *Int. Journal of Robotics Research*, 38(2–3):357–374.

[Illinois Center for a Smarter Electric Grid (ICSEG), 2016] Illinois Center for a Smarter Electric Grid (ICSEG) (2016). Texas 2000-June 2016 synthetic power case.

[Janson, 1991] Janson, B. N. (1991). Dynamic traffic assignment for urban road networks. *Transportation Research Part B: Methodological*, 25(2–3):143–161.

[Karmarkar, 1984] Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395.

[Karp, 1975] Karp, R. M. (1975). On the computational complexity of combinatorial problems. *Networks*, 5(1):45–68.

[Kempton and Tomić, 2005] Kempton, W. and Tomić, J. (2005). Vehicle-to-grid power fundamentals: Calculating capacity and net revenue. *Journal of Power Sources*, 144(1):268–279.

[Kerner, 2009] Kerner, B. S. (2009). *Introduction to modern traffic flow theory and control: the long road to three-phase traffic theory*. Springer Berlin Heidelberg, first edition.

[Khodayar et al., 2013] Khodayar, M. E., Wu, L., and Li, Z. (2013). Electric vehicle mobility in transmission-constrained hourly power generation scheduling. *IEEE Transactions on Smart Grid*, 4(2):779–788.

[Kirschen and Strbac, 2004] Kirschen, D. S. and Strbac, G. (2004). *Fundamentals of Power System Economics*. John Wiley & Sons, first edition.

[Krajzewicz et al., 2012] Krajzewicz, D., Erdmann, J., Behrisch, M., and Bieker, L. (2012). Recent development and applications of SUMO - Simulation of Urban MObility. *Int. Journal On Advances in Systems and Measurements*, 5(3&4):128–138.

[Le et al., 2015] Le, T., Kovács, P., Walton, N., Vu, H. L., Andrew, L. L. H., and Hoogendoorn, S. S. P. (2015). Decentralized signal control for urban road networks. *Transportation Research Part C: Emerging Technologies*, 58:431–450.

[Leighton et al., 1995] Leighton, T., Makedon, F., Plotkin, S., Stein, C., Tardos, É., and Tragoudas, S. (1995). Fast approximation algorithms for multicommodity flow problems. *Journal of Computer and System Sciences*, 50(2):228–243.

[Levin et al., 2017] Levin, M. W., Kockelman, K. M., Boyles, S. D., and Li, T. (2017). A general framework for modeling shared autonomous vehicles with dynamic network-loading and dynamic ride-sharing application. *Computers, Environment and Urban Systems*, 64:373 – 383.

[Levin et al., 2016] Levin, M. W., Li, T., Boyles, S. D., and Kockelman, K. M. (2016). A general framework for modeling shared autonomous vehicles. In *Annual Meeting of the Transportation Research Board*.

[Levin, 2018] Levin, P. (2018). $\tilde{\$}1$ trillion of real estate is on the move...here's why. Available at `https://medium.com/99-mph/1-trillion-of-real-estate-is-on-the-move-heres-why-94ee9233e5eb`. Retrieved on March 3, 2018.

[Lighthill and Whitham, 1955] Lighthill, M. J. and Whitham, G. B. (1955). On kinematic waves. II. a theory of traffic flow on long crowded roads. *Proc. of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 229(1178):317–345.

[Liu et al., 2009] Liu, H., Tesfatsion, L., and A., C. A. (2009). Derivation of locational marginal prices for restructured wholesale power markets. *Journal of Energy Markets*, 2(1):3–27.

[Maciejewski and Bischoff, 2017] Maciejewski, M. and Bischoff, J. (2017). Congestion effects of autonomous taxi fleets. *Transport*.

[Maciejewski et al., 2017] Maciejewski, M., Bischoff, J., Hörl, S., and Nagel, K. (2017). Towards a testbed for dynamic vehicle routing algorithms. In *Int. Conf. on Practical Applications of Agents and Multi-Agent Systems - Workshop on the application of agents to passenger transport (PAAMS-TAAPS)*.

[Mitchell et al., 2010] Mitchell, W. J., Borroni-Bird, C. E., and Burns, L. D. (2010). *Reinventing the automobile: Personal urban mobility for the 21st century*. MIT Press.

[Mittelmann, 2016] Mittelmann, H. D. (2016). Decision tree for optimization software. http://plato.asu.edu/guide.

[Mitzenmacher and Upfal, 2005] Mitzenmacher, M. and Upfal, E. (2005). *Probability and computing: Randomized algorithms and probabilistic analysis*. Cambridge Univ. Press.

[Neuburger, 1971] Neuburger, H. (1971). The economics of heavily congested roads. *Transportation Research*, 5(4):283–293.

[OECD, 2014] OECD (2014). The cost of air pollution - health impacts of road transport. Technical report, Organisation for Economic Co-operation and Development (OECD).

[Orlin, 1997] Orlin, J. B. (1997). A polynomial time primal network simplex algorithm for minimum cost flows. *Mathematical Programming*, 78(2):109–129.

[Osorio and Bierlaire, 2009] Osorio, C. and Bierlaire, M. (2009). An analytic finite capacity queueing network model capturing the propagation of congestion and blocking. *European Journal of Operational Research*, 196(3):996–1007.

[Overbye et al., 2004] Overbye, T. J., Cheng, X., and Sun, Y. (2004). A comparison of the AC and DC power flow models for LMP calculations. In *Hawaii Int. Conf. on System Sciences*.

[Papageorgiou et al., 1991] Papageorgiou, M., Hadj-Salem, H., and Blosseville, J.-M. (1991). ALINEA: A local feedback control law for on-ramp metering. *Transportation Research Record: Journal of the Transportation Research Board*, (1320):58–64.

[Pavone, 2015] Pavone, M. (2015). Autonomous Mobility-on-Demand systems for future urban mobility. In *Autonomes Fahren*. Springer.

[Pavone et al., 2011] Pavone, M., Smith, S. L., Frazzoli, E., and Rus, D. (2011). Load balancing for Mobility-on-Demand systems. In *Robotics: Science and Systems*.

[Pavone et al., 2012] Pavone, M., Smith, S. L., Frazzoli, E., and Rus, D. (2012). Robotic load balancing for Mobility-on-Demand systems. *Int. Journal of Robotics Research*, 31(7):839–854.

[Peeta and Mahmassani, 1995] Peeta, S. and Mahmassani, H. S. (1995). System optimal and user equilibrium time-dependent traffic assignment in congested networks. *Annals of Operations Research*, 60(1):81–113.

[Pérez et al., 2010] Pérez, J., Seco, F., Milanés, V., Jiménez, A., Díaz, J. C., and De Pedro, T. (2010). An RFID-based intelligent vehicle speed controller using active traffic signals. *Sensors*, 10(6):5872–5887.

[Pourazarm et al., 2016] Pourazarm, S., Cassandras, C. G., and Wang, T. (2016). Optimal routing and charging of energy-limited vehicles in traffic networks. *Int. Journal of Robust and Nonlinear Control*, 26(6):1325–1350.

[Raghavan and Tompson, 1987] Raghavan, P. and Tompson, C. D. (1987). Randomized rounding: A technique for provably good algorithms and algorithmic proofs. *Combinatorica*, 7(4):365–374.

[Reuters, 2017] Reuters (2017). NuTonomy hopes for second-quarter 2018 launch of paid Singapore self-driving car rides. Available at `https://www.reuters.com/article/idUSKCN1AY2IC`. Retrieved on March 2, 2018.

[Rossi et al., 2018] Rossi, F., Zhang, R., Hindy, Y., and Pavone, M. (2018). Routing autonomous vehicles in congested transportation networks: Structural properties and coordination algorithms. *Autonomous Robots*, 42(7):1427–1442.

[Rotering and Ilic, 2011] Rotering, N. and Ilic, M. (2011). Optimal charge control of plug-in hybrid electric vehicles in deregulated electricity markets. *IEEE Transactions on Power Systems*, 26(3):1021–1029.

[Seow et al., 2010] Seow, K. T., Dang, N. H., and Lee, D. H. (2010). A collaborative multiagent taxi-dispatch system. *IEEE Transactions on Automation Sciences and Engineering*, 7(3):607–616.

[Sioshansi, 2012] Sioshansi, R. (2012). OR Forum—modeling the impacts of electricity tariffs on plug-in hybrid electric vehicle charging, costs, and emissions. *Operations Research*, 60(3):506–516.

[Smith et al., 2013] Smith, S. L., Pavone, M., Schwager, M., Frazzoli, E., and Rus, D. (2013). Rebalancing the rebalancers: Optimally routing vehicles and drivers in Mobility-on-Demand systems. In *American Control Conference*.

[Spieser et al., 2014] Spieser, K., Treleaven, K., Zhang, R., Frazzoli, E., Morton, D., and Pavone, M. (2014). Toward a systematic approach to the design and evaluation of Autonomous Mobility-on-Demand systems: A case study in Singapore. In *Road Vehicle Automation*. Springer.

[Srinivasan, 1999] Srinivasan, A. (1999). A survey of the role of multicommodity flow and randomization in network design and routing. In *Randomization Methods in Algorithm Design*.

[Stott et al., 2009] Stott, B., Jardim, J., and Alsaç, O. (2009). DC power flow revisited. *IEEE Transactions on Power Systems*, 24(3):1290–1300.

[Tardos, 1985] Tardos, É. (1985). A strongly polynomial minimum cost circulation algorithm. *Combinatorica*, 5(3):247–255.

[Tarjan, 1997] Tarjan, R. E. (1997). Dynamic trees as search trees via euler tours, applied to the network simplex algorithm. *Mathematical Programming*, 78(2):169–177.

[Templeton, 2010] Templeton, B. (2010). Traffic congestion & capacity. Available at `http://www.templetons.com/brad/robocars/congestion.html`. Retrieved on March 2, 2018.

[Treiber et al., 2000] Treiber, M., Hennecke, A., and Helbing, D. (2000). Microscopic simulation of congested traffic. In *Traffic and Granular Flow '99*. Springer Berlin Heidelberg.

[Treleaven et al., 2011] Treleaven, K., Pavone, M., and Frazzoli, E. (2011). An asymptotically optimal algorithm for pickup and delivery problems. In *Proc. IEEE Conf. on Decision and Control*.

[Treleaven et al., 2012] Treleaven, K., Pavone, M., and Frazzoli, E. (2012). Models and efficient algorithms for pickup and delivery problems on roadmaps. In *Proc. IEEE Conf. on Decision and Control*.

[Treleaven et al., 2013] Treleaven, K., Pavone, M., and Frazzoli, E. (2013). 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.

[Turitsyn et al., 2010] Turitsyn, K., Sinitsyn, N., Backhaus, S., and Chertkov, M. (2010). Robust broadcast-communication control of electric vehicle charging. In *IEEE Int. Conf. on Smart Grid Communications (SmartGridComm)*.

[Tushar et al., 2012] Tushar, W., Saad, W., Poor, H. V., and Smith, D. B. (2012). Economics of electric vehicle charging: A game theoretic approach. *IEEE Transactions on Power Systems*, 3(4):1767–1778.

[United States Census Bureau, 2017] United States Census Bureau (2017). American Community Survey. Commuting in the United States: 2009. Supplemental Table B: Time of Departure. Available at `https://www.census.gov/hhes/commuting/data/commuting.html`.

[Urmson, 2014] Urmson, C. (2014). Just press go: Designing a self-driving vehicle. Available at `http://googleblog.blogspot.com/2014/05/just-press-go-designing-self-driving.html`. Retrieved on March 2, 2018.

[U.S. Dept. of Transportation, 2015] U.S. Dept. of Transportation (2015). Revised departmental guidance on valuation of travel time in economic analysis. Technical report.

[US EPA Office of Research and Development, 2018a] US EPA Office of Research and Development (2018a). Community Multiscale Air Quality (CMAQ). Available at `https://github.com/USEPA/CMAQ`.

[US EPA Office of Research and Development, 2018b] US EPA Office of Research and Development (2018b). Environmental Benefits Mapping and Analysis Program - Community Edition (BenMAP-CE)). Available at `https://www.epa.gov/benmap`.

[Wang et al., 2012] Wang, G., Negrete-Pincetic, M., Kowli, A., Shafieepoorfard, E., Meyn, S., and Shanbhag, U. V. (2012). Dynamic competitive equilibria in electricity markets. In *Control and optimization methods for electric smart grids*. Springer.

[Wang et al., 2010] Wang, L., Lin, A., and Chen, Y. (2010). Potential impact of recharging plug-in hybrid electric vehicles on locational marginal prices. *Naval Research Logistics*, 57(8):686–700.

[Wardrop, 1952] Wardrop, J. G. (1952). Some theoretical aspects of road traffic research. *Proc. of the Institution of Civil Engineers*, 1(3):325–362.

[Waymo, 2018] Waymo (2018). Early ride program. Available at `https://waymo.com/apply/`. Retrieved on March 2, 2018.

[Wilkie et al., 2014] Wilkie, D., Baykal, C., and Lin, M. C. (2014). Participatory route planning. In *ACM SIGSPATIAL*.

[Wilkie et al., 2011] Wilkie, D., van den Berg, J. P., Lin, M. C., and Manocha, D. (2011). Self-aware traffic route planning. In *Proc. AAAI Conf. on Artificial Intelligence*.

[World Health Organization, 2014] World Health Organization (2014). 7 million premature deaths annually linked to air pollution. `http://www.who.int/mediacentre/news/releases/2014/air-pollution/en/`. Retrieved on March 2, 2018.

[Xiao et al., 2015] Xiao, N., Frazzoli, E., Luo, Y., Li, Y., Wang, Y., and Wang, D. (2015). Throughput optimality of extended back-pressure traffic signal control algorithm. In *Mediterranean Conf. on Control and Automation*.

[Yang and Koutsopoulos, 1996] Yang, Q. and Koutsopoulos, H. N. (1996). A microscopic traffic simulator for evaluation of dynamic traffic management systems. *Transportation Research Part C: Emerging Technologies*, 4(3):113–129.

[Zhang and Pavone, 2015] Zhang, R. and Pavone, M. (2015). A queueing network approach to the analysis and control of Mobility-on-Demand systems. In *American Control Conference*.

[Zhang and Pavone, 2016] Zhang, R. and Pavone, M. (2016). Control of robotic Mobility-on-Demand systems: A queueing-theoretical perspective. *Int. Journal of Robotics Research*, 35(1–3):186–203.

[Zhang et al., 2016] Zhang, R., Rossi, F., and Pavone, M. (2016). Model predictive control of Autonomous Mobility-on-Demand systems. In *Proc. IEEE Conf. on Robotics and Automation*.