Ride Sharing

Siddhartha Banerjee and Ramesh Johari

Abstract Ridesharing platforms such as Didi, Lyft, Ola and Uber are increasingly important components of the transportation infrastructure. However, our understanding of their design and operations, and their effect on society at large, is not yet well understood. From an academic perspective, these platforms present challenges in large-scale learning, real-time stochastic control, and market design. Their popularity has led to a growing body of academic work across several disciplines, with researchers addressing similar questions with vastly different tools and models. Our aim in this chapter is to outline the main challenges in ridesharing, and to present an approach to modeling, optimizing, and reasoning about such platforms. We describe how rigorous analysis has been used with great success in designing efficient algorithms for real-time decision making, in informing the market design aspects of these platforms, and in understanding the impact of these platforms in their larger societal context.

1 Introduction

Since their founding over the last decade, ridesharing platforms have experienced extraordinary growth. At their core, these platforms reduce the friction in matching and dispatch for transportation. They do so based on a pair of matched driver and passenger mobile apps; a typical transaction starting with a potential passenger opening her app and requesting a ride, following which a centralized dispatcher matches her to a nearby driver if one is available. However, underlying this simple model are three features which fuel much of the success of these firms:

Siddhartha Banerjee
Cornell University, Ithaca NY, USA e-mail: sbanerjee@cornell.edu

Ramesh Johari
Stanford University, Palo Alto CA, USA e-mail: ramesh.johari@stanford.edu

1. *Data Collection and Analytics:* The driver and passenger apps enable extremely high temporal and spatial resolution for data collection. Ridesharing platforms track the position of all drivers and passengers in the system. Moreover, modern graph analytics and predictive models allow the platform to leverage this data to obtain very good estimates of travel-times, instantaneous demand, and long-term driver and passenger engagement metrics.
2. *Real-time Operations and Control:* An important reason behind the success of ridesharing platforms is their reliability and lack of friction in requesting a ride. Critical to this is the ability of the platform to rebalance demand and supply over time and space. A key tool for this purpose is *dynamic pricing*: ridesharing platforms adjust prices in real-time; in addition, however, the platforms have several other real-time dispatch and rebalancing tools, as well as different means for regulating and/or pooling instantaneous demand.
3. *Market design:* Ridesharing platforms typically do not employ drivers, but rather, create a marketplace between passengers and freelance drivers. Drivers can choose when and where to work (or not), and earn a share of the earnings per ride. To deal with this uncertainty in supply, ridesharing platforms need to understand the longer term equilibrium impacts of their real-time control on driver and passenger decisions, as well as on the platform's overall performance.

The challenge in studying ridesharing platforms is that the above features interact with each other in fundamental ways. To analyze any particular aspect of the platform, one has to account for its effect on the others – for example, to test a new pricing strategy, the platform must control for short timescale spatio-temporal variations in demand and supply, as well as long-term effect on driver and passenger entry decisions. On the other hand, incorporating all aspects simultaneously may lead to intractable models.

Our main aim in this chapter is to outline a stochastic-network based micro-foundation for ridesharing platforms. The framework we present is adapted primarily from our prior work on these questions [9, 8], which in turn were based on our experience in working on the design of pricing and matching algorithms at Lyft [1]. The popularity of ridesharing platforms has in recent years led to a growing body of work by researchers across several disciplines, including applied probability, optimization and network algorithms, economics and market design, transportation and urban planning, and even statistical physics. These works address similar problems, but using vastly differing models and tools, and this makes it difficult to translate the findings across different fields. While we do not in any way claim that the framework we present herein is the only way to model such platforms, we do believe that it captures the salient features of ridesharing, while being amenable to analysis and simulation. Our hope is that having such a common modeling framework will help unify the insights of researchers working in this exciting area.

The rest of the chapter is organized as follows. First, in Section 2, we provide a high-level description of the essential features of a ridesharing platform, and outline the different operational and market design challenges facing the platform. Next, in

[1] `www.lyft.com`

Section 3, building on our prior work in [9, 8], we describe how queueing-network models can be adapted to study ridesharing platforms. In particular, we focus on how they capture the critical features of such platforms: the high-resolution state description and two-sided nature, the real-time pricing and control tools, and the longer-term strategic interactions of drivers, passengers and the platform. In Section 4, we briefly summarize the operational and market design insights that can be derived from this modeling framework; in particular, we focus on how the model has been used to develop efficient control algorithms (based on results from [8]) and market mechanisms (following ideas outlined in [9]). Finally, in Section 5, we survey some of the related literature in ridesharing, and more generally, on control of stochastic networks and two-sided marketplaces.

2 Anatomy of a Modern Ridesharing Platform

In this section we describe the basic anatomy of a ridesharing platform. We divide our presentation in three parts: first, we discuss a fundamental separation of *timescales* that should guide any modeling of a ridesharing platform. Next, we discuss the *strategic choices* that guide the behavior of drivers and passengers on the platform. Finally, we discuss *operation and design* of the platform itself, taking both the timescale separation and incentives into account.

2.1 Timescales

There is an intrinsic timescale separation in the strategic interaction of drivers and passengers, as well as operation of the platform. In particular, ridesharing platforms have distinct behaviors on the following two timescales:

(i) A fast timescale (roughly, intra-minute), which captures the instantaneous dynamics of cars and passengers in the network; and
(ii) A slow timescale (roughly, intra-week), over which drivers make decisions as to how much and when to be on the platform.

The fast timescale provides the backdrop for the short-term operational and market design choices of the platform (especially pricing and matching), while the strategic consequences of these choices unfold over the slower timescale. While passengers primarily make entry decisions on the fast timescale (”Do I want to take this ride, given the current price and availability?”), drivers make such decisions on a longer timescale. This separation of the agent dynamics thus provides a convenient separation between how the platform’s policies affect drivers and passengers: control policies influence instantaneous passenger-vehicle dynamics, while the aggregate effect of these policies affect the longer-term entry decisions of drivers. Both timescales are discussed in more detail in the next subsection. We note that this viewpoint ignores other dynamics: for example, intra-hour changes in demand rates, or intra-year interactions between ridesharing firms and public transit providers. However, we argue that the two timescales we consider are crucial for understanding the first-order behavior of ridesharing platforms.

2.2 Strategic choices

What are the main strategic choices made by participants in ridesharing platforms? We already alluded to the strategic modeling of one side of the platform: for the most part, passengers can be modeled by assuming they make an instantaneous decision of whether to participate based on price (and possibility also availability information) on the platform. Platforms refer to ”app-opens” as opportunities to potentially engage a passenger; a subsequent ”ride request” refers to the passenger actually choosing to request a ride. In what follows, we will typically assume that passengers choose to request a ride as long as the price of the ride is below a private reservation value.

Drivers exhibit far more complex strategic behavior. While there is some evidence that drivers will locally optimize on short timescales (e.g., perhaps moving to a nearby block if there is evidence that prices are higher there), for the most part it is reasonble to assume that drivers are relatively *inelastic* on short timescales, as noted above.

Instead, the key choice made by drivers on a longer timescale is *entry* – both where they choose to drive, as well as what days and times during the week they choose to do so. Drivers make these decisions in response to what they observe on shorter timescales, forming expectations based on their experiences while driving. These entry decisions can be quite sophisticated, reflecting spatio-temporal differences in the driver’s experience within the platform.

The incentive structure of platforms can be quite complex, in ways that we do not necessarily capture in the models discussed in this chapter. For example, both sides *rate* each other after a ride is complete; these rating systems play a key role in determining, for drivers in particular, whether they are allowed to stay in the platform. As another example, experienced drivers are relatively sophisticated about time-of-day effects (i.e., when demand is expected to be higher or lower), and they will choose to keep their app online or offline accordingly. Platforms can also provide longer-term incentives to drivers, particularly through the *fee* structure (i.e., what percentage is given to drivers as their pay); we do not study the optimal design of fees. Finally, drivers are also constantly making choices about how and whether to *multihome* – i.e., participate in multiple platforms at once. Multihoming has important consequences for the availability of drivers in each platform, and warrants further attention from academic researchers studying ridesharing.

2.3 *Operation and market design*

A platform's operation and market design can be roughly summarized by three pieces: *information revelation*; *pricing*; and *dispatch*.

First, platforms reveal information to both passengers and drivers about the state of the system. For passengers, this is in the form of ETA (estimated time of arrival), which captures the distribution of drivers locally near a passenger. For drivers, the platform provides information on the distribution of demand. There is also extensive information collected and visible on the ratings provided to each side of the platform.

Second, a crucial aspect of these platforms is that they constantly make choices about the fare that will be charged to passengers. The typical model is that platforms publish a *base fare* schedule, that determines a baseline rate for any trip. Next, they will modify this base fare by a *multiplier* that adjusts the base fare to account for local demand-supply imbalances: when supply is scarce, the multiplier increases. In the past, platforms would not publish a fare estimate, because passengers were not asked to enter a destination at the time of the ride request; in these settings, the platform simply displayed the price multiplier to the passenger at time of ride request. Now, platforms publish the expected fare for a ride to the passenger at the time of ride request, in response to solicitation of the destination. These fares incorporate the price multiplier.

Once ride requests are made, platforms must actually match drivers to passengers; this is *dispatch*. Platforms typically match passengers to their *closest* driver. A more recent development in ridesharing is the introduction of *carpooling* (UberPool, Lyft Line); these products match *multiple passengers* with a single driver. In addition, platforms are becoming more sophisticated in how they manage the dispatch problem; for example, while in the past occupied drivers were not considered in the dispatch problem, now platforms will anticipate the fact that a driver will free up before making the next match. Such policies reduce driver idle times.

We have only provided a brief overview of the operational aspects of these platforms, focusing on the elements that are most important for our models below. Of course, in reality there is a great deal more complexity. For example, platforms must work to develop a product interface that allows passengers and drivers to make good choices; they must develop marketing mechanisms and onboarding mechanisms to attract driver supply; and they are working more and more to provide sophisticated long-term incentives to drivers, as noted above. These topics are important dimensions of the platforms, and may provide fruitful avenues for future study.

3 A Modeling Framework for Ridesharing Platforms

In this section, we outline a formal stochastic model of a ridesharing platform, and formulate the various associated control and market design problems. The basic framework we introduce below is adapted from the models proposed in [9] and [8].

It provides a rich modeling framework for ridesharing platforms, allowing us to study many different features, controls and metrics. Moreover, despite its complexity, the model turns out to be surprisingly amenable to analysis. In subsequent sections, we describe how the framework can be used to study control policies for the system dynamics, market-design questions for the driver-passenger marketplace and inter-platform interactions. Moreover, we also describe how the framework can be extended to study other design aspects of such platforms.

As we discuss before, a key to modeling ridesharing platforms is identifying the appropriate timescales for different agent interactions. To this end, our framework combines a Markov chain model with time-invariant parameters for capturing the instantaneous dynamics of vehicles and passengers (the *fast-timescale*), with an equilibrium analysis that captures entry decisions of drivers and passengers as well as the objectives of the platform, based on the average system performance (the *slow-timescale*. Although the model below allows for modeling fairly complex agent behavior, we focus on a particular behavioral model, wherein we assume that passengers primarily react under the fast-timescale (i.e., to instantaneous vehicle availability and prices), while drivers react under the slow-timescale (i.e., based on long-term average earnings). This is a choice we make based on a combination of our experience on working on these platforms [2], as well as for pedagogical reasons: as described in Section 2, these interactions capture the first-order behavior of these platforms, while enabling a tractable analysis of the stochastic platform dynamics and long-term strategic interactions.

That said, we note that our modeling choices ignore four important dynamics: (i) short-term fluctuations in system parameters (e.g., changing demand or bursty arrivals), (ii) short-term strategic behavior of drivers (e.g., strategic repositioning and ride cancellations), (iii) long-term effects on passenger behavior (e.g., demand screening due to persistent low availability or high prices), and (iv) competitive interactions between platforms (e.g., price cuts, driver retention incentives). Understanding the impact these interactions is important, but are to some extent secondary to the questions we consider. Thus, though some of our results, as well as a growing body of work by others, apply to these questions, we choose not to dwell on them in this chapter.

3.1 Modeling Stochastic Dynamics of the Platform

We now define a stochastic model for the fast-timescale dynamics of a ridesharing system. The main elements of the model are summarized in Figure 1.

State space and Markovian dynamics: We model the fast-timescale dynamics of a ridesharing platform using a *stochastic processing network* framework [33]: We consider a partition of a city into a set of *n stations* (corresponding to locations or neighborhoods in a city), and use a continuous-time Markov chain to track the

[2] SB worked there in 2014-15, and was involved in designing their early pricing algorithms.

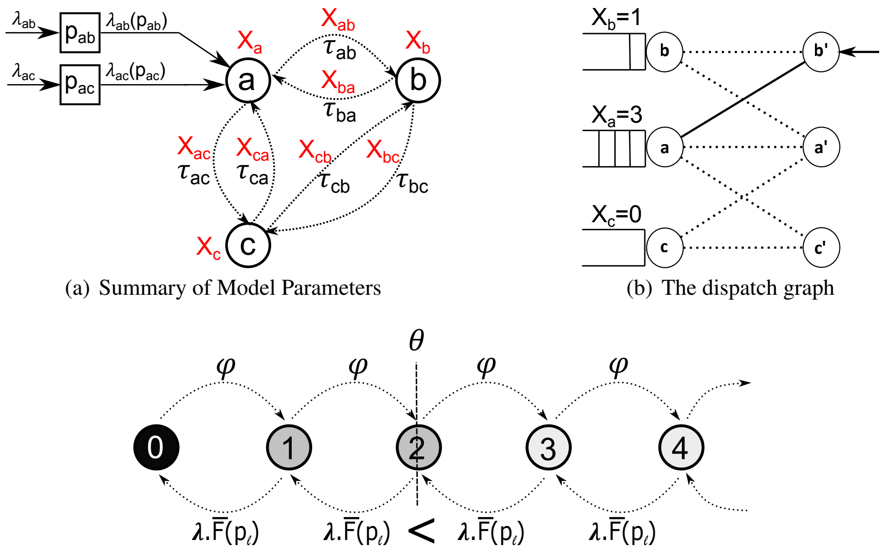
positions of k *units* (i.e., vehicles) which are either idle at these stations or transiting (i.e., in a ride) between them. Each ride involves a driver picking up a passenger in one region, and dropping her off in another. These rides can be modulated via a set of controls – primarily pricing, but also dispatch and empty-unit rebalancing. To study the efficacy of different controls, we analyze the long-term average performance of the system; in other words, we study various metrics of interest in *steady state*. We henceforth use $V = \{1, 2, \ldots, n\}$ to denote the set of stations, and $E = \{(i, j) \in V \times V\}$ to be the set of source-destination pairs

Formally, at any time $t \geq 0$, the state of the ridesharing system is denoted as $\mathbf{X}(t) = \{\{X_i(t)\}_{i \in V}, \{X_{ij}(t)\}_{ij \in E}\}$, where $X_v(t)$ denotes the number of units which are idle at station i, and $X_{ij}(t)$ denotes the number of units in transit (either in a ride, or rebalancing) between stations i and j. As per our assumption, the sum over all states must at all times add up to k. Denoting $N = n + \binom{n}{2}$ to be the dimension of $\mathbf{X}$, we have that the state space of the Markov chain is given by $\mathscr{S}_{N,k} = \left\{\{x_i, x_{ij}\} \in \mathbb{N}_0^N \,|\, \sum_{i \in V} x_i + \sum_{ij \in E} x_{ij} = k\right\}$. Note that the state-space is finite. Since our focus is on the long-run average performance, i.e., under the steady state of the Markov chain, for ease of notation, we henceforth suppress the dependence on time t.

Passenger arrivals and ride requests: Potential passengers who desire to travel between stations i and j (henceforth, type-ij passengers) arrive at station i following a stochastic process with average rate λ_{ij} (alternately, average inter-arrival time $1/\lambda_{ij}$). It is reasonable to assume the inter-arrival times to be independent, and hence, invoking the Palm-Khintchine theorem (cf. Chapter 14 in [32]), we assume that type-ij passengers arrive according to a Poisson process of rate λ_{ij}.

To model the 'willingness-to-pay' of the passengers, we assume that each type-ij passenger has a ride value drawn independently from a distribution $F_{ij}(\cdot)$. Upon arrival at i, a customer is quoted a *point-to-point* price p_{ij} (which may potentially depend on the current state $\mathbf{X}(t)$); she then requests a ride if her value exceeds this price, i.e. with probability $1 - F_{ij}(p_{ij})$. At this point, if at least one unit is available at station i (or more generally, at any sufficiently 'nearby' station), then she is matched to it. If on the other hand she is unwilling to pay the price, or is not matched to a vehicle, then she leaves the system *immediately*. We assume that F_{ij} has a density and that all values are positive with some probability, i.e. $F_{ij}(0) < 1$.

The passenger dynamics outlined above is referred to in the stochastic modeling community as a *loss-system* model. The possibility of immediate departure without a ride request captures the possibility that a passenger typically has outside options (walking/public transit/other ridesharing firms) which she turns to if the platform proves unattractive at the moment. In practice, with dynamic pricing, some passengers may tend to wait for a while to see if prices change (although ridesharing platforms may also freeze their prices for a given passenger, while still adjusting them for others), or cars become available. Such heterogeneity in passenger impatience can be accommodated in our model to some extent (for example, as a *negative* queue, akin to lost-sales models in inventory systems). However, the effects of such behavior is not well understood in practice, and currently, most ridesharing firms do not specifically account for passenger impatience in their policies.



(a) Summary of Model Parameters

(b) The dispatch graph

(c) Idle driver queue under state-dependent pricing

Fig. 1 *Illustrating the stochastic dynamics of a ridesharing platform*:
Figure 1(a) summarizes the primary components of our model for a ridesharing platform (in this case, with state-independent pricing). The platform depicted has 4 stations $V = \{a,b,c,d\}$, and m vehicles. Random variables are depicted in red; here, the random process $\{X_v\}$ tracks the number of idle units at stations, while $\{X_{ij}\}$ tracks the units in transit between stations i and j (with mean travel-time τ_{ij}). Zooming into station a, we see that passengers with destination $\{b,c\}$ arrive at a according to Poisson processes with rate $\{\lambda_{ab},\lambda_{ac}\}$; these arrivals are then 'thinned' to $\lambda_{av}(p_{av})$ by setting (state-independent, but destination dependent) prices $\{p_{ab},p_{ac}\}$. (Adapted from [8])
In Figure 1(b), we depict the bipartite graph for the dispatch problem, for the network in Figure 1(a) under the assumption that station pairs (a,b) and (a,c) are close enough to use each other's supply. An arriving passenger at station b can thus be matched to a vehicle at either station a or b – in the figure, we choose to match the arrival to a vehicle at station a. (Adapted from [10])
Figure 1(c) shows the birth-death chain for the number of idle drivers in a single station, under *local* state-dependent pricing policies. The arrival rate ϕ of vehicles to the station is determined by the overall network. The rate of departures (i.e., matched rides), however, is modulated by the pricing policy. Here, we have depicted a base arrival rate of passengers λ, and a simple *single-threshold* local pricing policy, where the platform uses a 'base' fare p_ℓ when the number of drivers is greater than a threshold θ, else charges a 'primetime' price $p_h > p_\ell$ (hence the queue drains slower when there are $\leq \theta$ drivers). (Adapted from [9])

Vehicle travel times: Once a unit is dispatched to serve a passenger, it then needs to go pick up and drive the passenger to her destination station. We use the state variable $X_{ij}(t)$ to track the number of units in transition between stations i and j. When a customer engages a unit to travel from i to j, the state changes to $\mathbf{X} - e_i + e_{ij}$ (i.e., $X_i \to X_i - 1$ and $X_{ij} \to X_{ij} + 1$). The unit remains in transit for a random time, drawn independently from some general distribution $G_{ij}(\cdot)$ with mean τ_{ij}. Upon reaching its destination, the unit drops off the passenger, and the system state changes to $\mathbf{X} - e_{ij} + e_j$.

For convenience, we will assume henceforth that transit times are exponentially distributed, with average transit-times τ_{ij}. This is primarily for ease of exposition, as it allows us to keep track of only the number of vehicles in transit between any two stations (as opposed to their exact time of arrival; this follows from the memoryless property of the exponential distribution). We note though that the predictions of the model remain essentially unchanged for any general (independent) travel time with mean τ_{ij}. We also assume that the demand characteristics and ride rewards are independent of the actual transit times (dependence on average transit times τ_{ij} can be embedded in the model parameters). Finally, note that we do not model stochastic correlation in travel times (e.g., that might arise because trips share a common road network) — a potentially interesting direction for future work.

We conclude with two additional observations about transit-times. First, we note that though the above discussion is primarily for vehicles dropping-off passengers, the transit times also apply to settings where empty vehicles move between stations to improve the demand-supply balance. Second, in many cases, the model and results greatly simplify if we assume that transit times are identically zero: in particular, note that in such a setting, we only need to keep track of idle vehicles at the stations. Introducing transit times tends to complicate analysis as it may lead to situations where availability is low as almost all vehicles are in transit (this corresponds to the so-called *heavy-traffic* regime in queueing models). Understanding the significance of transit times are in designing ridesharing policies is an under-explored question, and one which we will not deal with in this chapter in any significant detail.

3.2 Platform Controls

We now consider three primary ways in which the platform can intervene to affect the fast-timescale dynamics: (i) demand modulation via *pricing*, (ii) demand redirection via *dispatch*, and (iii) supply redirection via *empty-vehicle rebalancing*. We describe these in details below; note however that all these different controls are essentially linear transformations of the demand and supply flows, and moreover, can be combined together (and often are in practice).

Demand modulation (pricing): By adjusting the price p_{ij} for a ride from i to j, the platform can modulate the rate at which such rides are requested. To understand the effect of such a price, it is useful to define the *inverse demand* (or *quantile*) function $q_{ij} = 1 - F_{ij}(p_{ij})$ [3]. Now, for a fixed pricing policy $\mathbf{p}$ with corresponding quantiles $\mathbf{q}$, the *effective demand rate* from i to j (i.e. type-ij passengers with value exceeding p_{ij}) follows a Poisson process with rate $\lambda_{ij}q_{ij}$ – this follows from the probabilistic thinning property of a Poisson process.

[3] It is convenient to assume that the density of F_{ij} is positive everywhere in its domain, implying that there is a 1-1 mapping between prices and quantiles; this allows us to write $p_{ij} = F_{ij}^{-1}(1-q_{ij})$. We note however that this is not necessary for the results we present.

The most general model for pricing in the above model is that of *global state-dependent* prices, where the platform selects $p_{ij}(t)$ at time t as a *function of the overall state* $\mathbf{X}(t)$ – this induces a state-dependent Poisson process of type-ij passengers with rate $\lambda_{ij}q_{ij}(\mathbf{X})$. A natural relaxation of this is that of *local state-dependent* prices, where p_{ij} is a function of the *local state* $X_i(t)$ at the source. Finally, in *state-independent* pricing, p_{ij} is set to be independent of the instantaneous system state. The three pricing schemes decrease in complexity, and moreover, require decreasing levels of system engineering to enable – understanding their comparative behavior is thus of great importance. [8] study the relation between global state-dependent and state-independent prices, while [9] focus on local state-independent prices to understand the value of dynamic pricing in ridesharing platforms.

Demand redirection (dispatch): Though it is typically infeasible to redirect passengers to nearby stations (although this has been experimented with by Lyft and Uber in some markets), what is often possible is to match an incoming ride request station i to units which are idle at 'nearby' stations. This is based on the underlying assumption that passengers are insensitive to small delays in pickup as compared to pickup time of the nearest unit. This is not strictly true in practice, as passengers are known to be sensitive to the pickup time ('ETA'); however, it is a convenient abstraction for our model, and moreover, can be refined by incorporating probabilistic ride cancellations due to longer pickup times. Moreover, longer dispatches may affect drivers, and this can be modeled by a cost for each possible dispatch decision.

To formally define a dispatch policy, we define a *compatibility graph* $G = (V,E)$ on the set of stations, with edges between pairs of stations that are near enough such that a passenger arriving at one can be served using a unit from the other (see Figure 1(b) for an example). As with pricing, we can define a state-dependent dispatch policy $\mu(\mathbf{X})$ which, for each ride requested at station i, decides from which station in $\{i\} \cup \{j : (i,j) \in E\}$, the customer is served. Such a dispatch policy now induces a rate $f_{ij}(\mu)$ of customers arriving at i that travel to j using a unit from k, and a rate $z_{ik}(\mu)$ of customers arriving at i who are matched to a unit at k. Note that if μ is chosen in a state-independent manner (wherein a request is randomly routed to a neighboring station), then it may lead to a failed dispatch despite there being idle units; such policies however are more tractable to analyze, and hence have been considered in [40] and [8]. More recently, state-dependent dispatch policies were analyzed by [10] (albeit without costs for dispatch from neighboring stations).

Supply redirection (rebalancing): This is a catch-all for any control policy which allows the platform to affect the position of a unit at the end of a ride, i.e., whether the unit remains at the destination station or moves to another station without a passenger. Such a control is sometimes referred to as *empty-car rebalancing*, and such rebalancing typically is modeled as incurring a cost (for vehicle miles traveled/idle time of drivers). In practice, rebalancing is less common in current ridesharing systems (in comparison to bikesharing/carsharing systems), as drivers themselves choose where to go when idle – however, platforms do try to influence these decisions via information displays, incentive schemes, etc. With the potential introduction of autonomous vehicles, such control may become more prevalent.

We can model a rebalancing policy as a state-dependent control $\mathbf{r}(\mathbf{X})$ which, for each trip ending at a station i, redirects the unit to some station j (which could be i). This results in an increase in state X_{ij}, and has associated cost c_{ij}. Since redirection is costly for drivers, it is natural to assume that redirected units arriving at a station are not redirected again. Details of how to incorporate this in the above model are provided in [14] and [8].

3.3 Platform Objectives

Given the above system dynamics (with *fixed* parameters $k, \lambda_{ij}, F_{ij}, \tau_{ij}$), our aim is to study the long-term average performance of various platform metrics. More precisely, we want to design control policies to maximize relevant performance metrics under the *stationary distribution* $\pi(\mathbf{x})$ of the Markov chain induced by our controls. Note that for given n, k and under any policy, the resulting Markov chain is finite-state (since the number of stations and units is fixed); furthermore, it is irreducible under weak assumptions on the prices and the demand (see [8] for details). Now, using basic Markov chain theory, we have that our system has a unique steady-state distribution $\pi(\cdot)$ with $\pi(\mathbf{x}) \geq 0 \forall \mathbf{x} \in \mathscr{S}_{N,k}$ and $\sum_{\mathbf{x} \in \mathscr{S}_{N,k}} \pi(\mathbf{x}) = 1$.

Following [8], we consider objective functions that decompose into per-ride reward functions $I_{ij}(p)$, which correspond to the reward obtained from a passenger riding between stations i and j at price p. In particular, such a structure admits three canonical objectives:

- *Volume of Trade* or Throughput: the total rate of rides in the system (setting $I_{ij}(p) = 1$).
- *Social welfare*: the contribution to social welfare from each $i \to j$ ride is given by $I_{ij}(p) = \mathbb{E}_{V \sim F_{ij}}[V | V \geq p]$.
- *Revenue*: to find the platform's revenue rate (assuming it keeps a fraction $1-\gamma$ of the earnings), we set $I_{ij}(p) = (1-\gamma) \cdot p$.

To formally define the objective, we focus on the case of pricing. Now, for a given objective $I_{ij}(\cdot)$, the aim of the platform is to select prices $\mathbf{p}(\mathbf{X})$ (with corresponding quantiles $\mathbf{q}(\mathbf{X})$) that maximizes the rate of reward accumulation under the stationary distribution. This can be written as:

$$\mathrm{OBJ}_m(\mathbf{p}) = \sum_{\mathbf{x} \in \mathscr{S}_{N,k}} \pi(\mathbf{x}) \cdot \left(\sum_{i,j} \lambda_{ij} q_{ij}(\mathbf{x}) I_{ij}\big(p_{ij}(\mathbf{x})\big) \right), \tag{1}$$

where $\pi(\mathbf{x})$ is the stationary distribution of the Markov chain under pricing policy $\mathbf{p}$. Equation (1) can be understood as follows: at any station i, customers destined for j arrive via a Poisson process with rate λ_{ij}, and find the system in state $\mathbf{x} \in \mathscr{S}_{N,k}$ with probability $\pi(\mathbf{x})$ (this follows from the 'Poisson Averages See Time Averages' or PASTA property [33]). They are then quoted a price $p_{ij}(\mathbf{x})$, and engage a ride with probability $q_{ij}(\mathbf{x}) = 1 - F_{ij}(p_{ij}(\mathbf{x}))$. The resulting ride contributes $I_{ij}(q_{ij}(\mathbf{x}))$ to the expected objective. Recall that unavailability of units is captured by our assumption that $q_{ij}(\mathbf{x}) = 0$ whenever $x_i = 0$.

Though the above equation is most naturally written in terms of prices, it turns out to be non-concave even for a single station. However, a standard price-theoretic trick (for example, see [29]) in such cases is to instead write the objective in terms of quantiles, whereupon it turns out to be concave for most cases of interest. In particular, abusing notation to define $I_{ij}(q) := I_{ij}(F_{ij}^{-1}(1-q))$, and defining *reward curves* $R_{ij}(q) := q \cdot I_{ij}(q)$, it can be shown that $R_{ij}(q)$ are concave in q for throughput and welfare under any distribution, and for revenue under *regular* distributions (a wide class of distributions which includes all increasing hazard-rate distributions; see [8, 29] for details).

However, the convexity of $R_{ij}(q)$ does not imply that the optimization problem in Equation 1 admits a tractable solution, as we still have to determine the average under the stationary distribution. This involves solving a fixed-point constraint, which in general can be non-tractable. In fact, [8] provide an example which shows that the problem is non-concave even for a setting with 3 stations and a single unit!

3.4 Local Controls and Closed Queueing Models

Although the stochastic dynamics described above is complex, it is still amenable to study via simulation. Moreover, in some special cases, its analysis can be greatly simplified using classical results from queueing theory [47, 34]. In particular, a critical tool used in [9, 8] is the fact that *under state independent control policies (pricing, dispatch, rebalancing), as well as under local state-dependent pricing, the stationary distribution of the resulting Markov chains is known in closed form*. This now allows us to study the design of control policies in an analytic way. We now briefly provide some background behind this methodology.

The general Markov chain described in the previous sections (involving a fixed number k of units, located in one of N queues) is a special example of a *closed queueing network* [34, 47]. Closed here refers to the fact that the number of units remains constant; in contrast, in open networks, units may arrive and depart from the system. These networks are well-studied in applied probability, and in general may have complex stationary distributions. However, a critical property uniting the settings mentioned above (state-independent controls, local state-dependent pricing) is that the resulting Markov chain in each case is *quasireversible* [34]. This is a particular structural property of Markov chains which generalizes the notion of reversibility. The exact definition is somewhat technical, but for our purposes, the crucial fact is that *quasireversibility is sufficient to ensure the stationary distribution is product form*, i.e.

$$\pi(\mathbf{x}) = \frac{1}{Z} \prod_i f_i(x_i) \prod_{ij} f_{ij}(x_{ij}),$$

where Z is the appropriate normalizing constant. The exact form of the local potentials f_i, f_{ij} depend on the precise nature of the system and controls; see [9, 8] for details. For illustration purposes, we develop this in more detail below for the special case of state-independent pricing and instantaneous transfers.

An important property of state-independent control prices is that the rate of units departing from any station i at any time t when $X_i(t) > 0$ is a constant, independent of the state of the network. The resulting model is a special case of a closed queueing model proposed by Gordon and Newell [23].

Definition 1 *A* Gordon-Newell network *is a continuous-time Markov chain on states* $\mathbf{x} \in \mathscr{S}_{N,k}$*, in which for any state* $\mathbf{x}$ *and any* $i, j \in [n]$*, the chain transitions from* $\mathbf{x}$ *to* $\mathbf{x} - e_i + e_j$ *at a rate* $\mu_i r_{ij} \mathbb{1}_{\{x_i(t)>0\}}$*, where* $\mu_i > 0$ *is referred to as the* service rate *at station i, and* $r_{ij} \geq 0$ *are the* routing probabilities *that satisfy* $\sum_j r_{ij} = 1$.

In other words, if units are present at a station i in state $\mathbf{x}$, then departures from that station occur according to a Poisson distribution with rate $\mu_i > 0$; conditioning on a departure, the destination j is chosen according to state-independent routing probabilities r_{ij}. For this network, the resulting steady-state distribution $\{\pi_{\mathbf{p},m}(\mathbf{x})\}_{\mathbf{x}\in\mathscr{S}_{N,k}}$ was established to be product form via the celebrated Gordon-Newell theorem.

Theorem 2 (Gordon-Newell Theorem [23]). *Consider a k-unit n-station Gordon-Newell network with transition rates* μ_i *and routing probabilities* r_{ij}*. Let* $\{w_i\}_{i\in[n]}$ *denote the invariant distribution associated with the routing probability matrix* $\{r_{ij}\}_{i,j\in[n]}$*, and define the* traffic intensity *at station i as* $\rho_i = w_i / \sum_j r_{ij}$*. Then the stationary distribution is given by:*

$$\pi(\mathbf{x}) = \frac{1}{G_m} \prod_{j=1}^{n} (\rho_j)^{x_j}, \tag{2}$$

with normalization constant $G_m = \sum_{\mathbf{x}\in\mathscr{S}_{k,m}} \prod_{j=1}^{n} (\rho_j)^{x_j}$.

To see that the Markovian dynamics resulting from state-independent pricing policies fulfill the conditions of Gordon-Newell networks, observe that fixing a price p_{ij} (with corresponding q_{ij}) results in a Poisson process with rate $\lambda_{ij} q_{ij}$ of arriving customers *willing to pay price* p_{ij}. These customers engage a unit only if one is available, otherwise they leave the system. Thus, given quantiles $\mathbf{q}$, the time to a departure from station i is distributed exponentially with rate $\mu_i = \sum_j \lambda_{ij} q_{ij}$ when $X_i > 0$ and with rate 0 otherwise. Further, conditioned on an arriving customer having value at least equal to the quoted price, the probability that the customer's destination is j, is $r_{ij} = \lambda_{ij} q_{ij} / \sum_k \lambda_{ik} q_{ik}$, independent of system state. Now we can use the Gordon-Newell theorem to simplify the objective function in Equation (1) to get an explicit function of the quantiles $\mathbf{q}$. The functions obtained are somewhat involved, and hence we omit them here; interested readers should refer [8] for details.

3.5 Modeling Endogenous Entry of Drivers

Finally, we turn our attention to agent behavior in the slow-timescale – in particular, we discuss how the above model can be used to model the endogenous entry decisions made by drivers.

At a high level, the slow timescale allows us to capture the marketplace aspect of ridesharing platforms, by allowing us to specify the strategic aspects of agent-platform interactions. In particular, as we mention before, our primary use of the slow timescale is to model the endogenous entry decision of drivers, which thereby determines the number of vehicles k in equilibrium. We note though that a similar idea of determining the parameters of the fast-timescale model based on strategic considerations at a slower timescale can be used to model other agent interactions – in particular, an interesting open question is to model the effect of high ETAs or prices on passenger rates.

Our treatment here follows the model in [9]. The main assumption for the slow-timescale driver decisions is that each potential driver in the pool has a *reservation earning-rate* (or earning-rate target), and makes an *endogenous entry decision* (i.e., determines whether or not to work on the platform) by comparing their expected earning-rate on the platform to this target. We assume that the earning-rate target for each driver is drawn i.i.d. from some distribution G_d. On the other hand, the earning rate of a driver on the platform depends on the specific wage structure implemented by the platform – this is an aspect which different platforms have experimented with, and one whose effects are not yet well understood. For this chapter, as in [9] (and as above), we assume that the driver gets a fraction γ of the price of each ride that he is matched to. Note that this is the policy currently followed by most ridesharing firms.

For convenience, we henceforth analyze the behavior of a single station under local state-dependent pricing; this can however be extended to the entire network using standard queueing theoretic tools. The setting is depicted in Figure 1(c). Let X represent the instantaneous number of idle units at the station, and suppose the potential pool of drivers is of size $\bar{k}$. Given pricing policy $p(X)$, the passenger arrival rate is $\lambda\bar{F}(p(X_i))$ (where $\bar{F}(\cdot) = 1 - F(\cdot)$); the equilibrium number of drivers k is now determined by the system performance in the fast timescale (which in turn depends on k). Formally, the equilibrium rates of passenger requests and number of drivers must satisfy:

$$\lambda(X) = \lambda\bar{F}(p(X)) \qquad , \qquad k = \bar{k}G_d\left(\frac{\eta}{\iota+\tau}\right), \tag{3}$$

where η denotes the expected per-ride earnings, ι the expected waiting time for a driver between rides, and τ the expected ride time. Exact expressions for these can be computed using the product form characterization of the stationary distribution discussed in Section 3.4. Note though that η and ι depend on λ and μ, as well as the pricing policy $p(X)$. Details of these computations, and of the existence/uniqueness of the equilibrium, are given in [9].

4 Analyzing the Model: Key Findings

We now briefly describe some of the insights that can we obtain by analyzing the queueing-theoretic model described laid out in the preceding sections. First, we summarize the results from [8], where for any given number of units K, the authors show how we can design control policies for the fast-timescale dynamics, with strong performance guarantees. On the positive side, these policies surprisingly turn out to be state-independent. On the negative side, however, the results do not give insight into the number of units K that emerge in equilibrium. Moreover, the results critically depend on having full knowledge of the Markov chain parameters (in particular, passenger arrival rates λ_{ij} and willingess-to-pay distributions F_{ij}).

To characterize the equilibrium behavior of the system, as well as understand how to achieve good performance without perfect knowledge of system parameters, we turn to the use of state-dependent prices (in particular, local state-dependent prices). In Section 4.2, we summarize the results from [9]. Here, the authors show that while on the one hand state independent and dependent prices are asymptotically the same, the latter is much more robust to mis-specifications in system parameters. In more detail, they show that on the one hand, as the number of drivers and rate of passenger arrivals jointly scale to infinity, state-dependent pricing becomes asymptotically equal to state-independent pricing; on the other hand, they show that state-dependent prices are much less sensitive compared to state-independent prices under small perturbations in the system parameters.

4.1 Fast-Timescale Control of Platform Dynamics

We first turn to the question of choosing control policies for a given k that maximize the objective considered in Equation 1. Note thought that these controls are extremely high-dimensional as they can in general depend on the instantaneous state of the system; moreover, as we discuss in Section 3.3, the problem is non-convex even in simple settings.

In [8], the authors circumvent these problems via a novel technique for deriving control policies based on a convex relaxation which they term the *elevated flow relaxation*. The main idea behind this technique is to construct a concave pointwise upper bound for the objective, which is convex and hence admits a tractable optimization. This can be done essentially by assuming an infinite supply at each node, while simultaneously introducing additional flow-conservation constraints to capture the balance of units arriving to and exiting from any node in the stationary distribution. The authors prove that their new elevated objective is bounded below by the original objective, and thus optimal solutions in the elevated optimization problem are bounded below in value by optimal solutions in the original optimization problem. More importantly, they also prove lower bounds on the performance of natural state-independent policies provided by the relaxation, thereby establishing their approximation guarantees. They do so via the following three-step program:

1. First, they derive efficiently-computable upper bounds for the performance of any control policy, which encode essential ‘conservation laws’ of the system (in particular, flow balance at nodes and capacity constraints on number of vehicles on the road), while being amenable to optimization.
2. Next, they show that under an infinite-supply limit, where $k \nearrow \infty$ while all other parameters stay fixed, the achievable objective values under *state-independent control policies* exactly match the set of achievable upper bounds defined by the elevated flow relaxation.
3. Finally, using the product-form characterization of the stationary distribution under state-independent policies, they show that the performance of any policy in a setting with k units is within a factor of $1+(n-1)/k$ (assuming instantaneous transfers) of its performance in the infinite-supply setting [4].

The versatility of the above framework allows it to be extended to more complex *multi-objective settings*, where the goal is to optimize some objective subject to a lower bound on another. A canonical example of this is the so-called *Ramsey pricing problem* [41], where the platform aims to design a pricing policy to maximize system revenue subject to a lower bound on the system welfare; this is very relevant for most nascent ridesharing platforms, who are aiming to build a customer base. The authors also extend their analysis to include travel times, and show that increasing travel times may lead the approximation factor to degrade from $1+O(1/k)$ under instantaneous transfers to $1+O(1/\sqrt{k})$ in the worst case. One way to understand this phenomena is to note that assuming all other parameters stay fixed, increasing travel times leads to an increase in the amount of ‘work’ each incoming passenger needs from the k units; in the extreme case, most units are in transit at any time, and get engaged immediately as soon as they become idle.

The results in [9] also recover and unify several other existing results in this area, and provide a general framework for deriving approximation algorithms for many other settings. In particular, the techniques provide an elementary proof of the so-called large-market (or ‘fluid limit’) optimality of the proposed state-independent policy; this form of limiting result was also obtained around the same time by [14] (for rebalancing) and [40] (for dispatch), via more technical limit interchange arguments.

Finally, the authors of [8] also show that the bounds obtained via the above technique are *tight* compared to the optimal policy. This follows from an example earlier proposed in [52], comprising of a ring of n stations, with equal request rate between every pair of adjacent stations except for one *bottleneck* link. The same example can also be modified to show similar bounds on the performance of *any* state-independent policy. Going beyond such policies is difficult; however, some recent work [10] has proposed state-dependent algorithms which for large k, and under some additional conditions, can be shown to have $1+e^{-O(k)}$ competitive ratio.

[4] Here we define the approximation factor as the ratio of the objective of the optimal policy to that of the proposed policy; this convention, which ensures the approximation ratio is always greater than 1, is common in the approximation algorithms literature.

4.2 The Slow Timescale: Pricing and Driver Entry

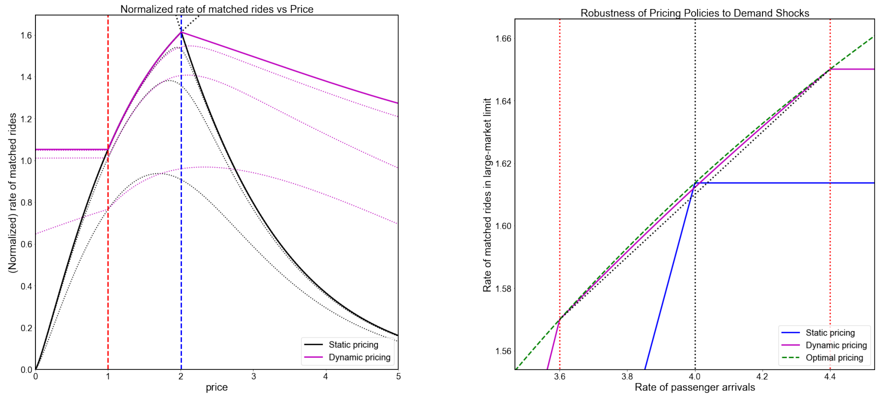
The previous section provides a solution for the chief fast-timescale operational design questions in a ridesharing platform – given a supply of units, how best to use pricing, dispatch and rebalancing to balance the demand and supply. We next turn to understanding the slow-timescale response of drivers to such policies – in particular, we want to understand the impact of pricing policies on the overall marketplace equilibrium. To this end, we summarize the findings of [9], who use the micro-foundations in Section 3 to compare the impact of two pricing schemes: (i) state-independent (i.e., *static*) pricing, where the price is fixed as a function of the fast-timescale system parameters, but not the instantaneous state [5]), and (ii) *dynamic* pricing policies, where the prices react to the system state. Following the discussion in Section 3.4, they focus on local state-dependent pricing policies, as these admit closed-form stationary distributions; moreover, they focus on a simple class of *threshold* policies, wherein the platform raises the price whenever the number of available drivers in a region falls below a threshold.

The results in [8] which we summarized in Section 4.1 indicate that when the supply is fixed, then state-independent policies, if correctly chosen, are very competitive compared to the optimal policy. The main contribution of [9] is in understanding the effects of the pricing scheme on the equilibrium supply of drivers, and also, on characterizing the sensitivity of the two policies to parameter uncertainty. The major technical hurdle in doing so is that unlike the fast-timescale stationary distribution, the equilibrium of the system (as defined in Equation 3) does not admit a closed-form expression for the driver/passenger arrival rates.

The Large-Market Scaling: To circumvent this, the authors of [9] study the system under a *large-market scaling*, wherein they consider a sequence of systems parametrized by ν, wherein $\bar{K}(\nu) = \bar{K}_0\nu$ and $\lambda(\nu) = \lambda\nu$, and all other system parameters, as well as the pricing policy $\mathbf{p}$, are held fixed. They then let ν approach ∞, and study the *normalized* equilibrium supply state, i.e. $\lim_{\nu\to\infty} K(\nu)/\nu$, of the limiting system. For dynamic pricing policies, in addition to scaling $\bar{K}_0$ and μ_0, they keep the set of prices fixed, but allow the threshold $\theta(\nu)$ to scale with ν.

Under the large-market scaling, the authors of [9] characterize the equilibrium rates for the limiting system in closed form, for both static and dynamic pricing. An example of this convergence and limiting characterization can be seen graphically in Figure 2(a), which is similar to the plots given in [9]; here we have plotted the normalized equilibrium throughput (i.e., rate of matched rides) vs. static price p (the green curves), and also, for a class of dynamic pricing policies (the maroon curves) where we keep one price fixed at the red vertical dotted line. The dotted curves are numerically computed for $n \in \{1, 10, 100, 1000\}$, and can be seen to be monotonically converging up to the solid curves, which plot the theoretical large-market limits characterized in [9].

[5] More specifically, these correspond to quasi-static policies, where the price remains fixed for blocks of time on the order of hours, but can be changed over slower timescales to reflect change in average demand/supply. Such policies are commonly used by traditional taxi firms.



(a) Pricing policies in the large-market limit

(b) Robustness of dynamic pricing

Fig. 2 *Impact of pricing policies in ridesharing platforms: Figure 2(a) depicts the normalized equilibrium throughput in a ridesharing platform under static pricing (in black), and under dynamic pricing (in maroon) with one price fixed at the red vertical line. The dotted lines show the throughput curve for different values of the scaling parameter ν, with higher curves corresponding to higher values of ν. The solid curves plot the theoretical large-market limits. Note that in the large-market limit, the optimal throughput under both policies is the same (indicated by the black vertical dotted line).*
Figure 2(b) demonstrates the sensitivity of pricing policies to demand uncertainty: For a fixed $\bar{K}_0$, we consider baseline passenger arrival rate $\lambda \in 4 \pm 10\%$, and compare the normalized throughput under (i) the optimal static policy with $\lambda = 4$ (indicated by the black vertical dotted line), and (ii) the dynamic-pricing policy which sets p_ℓ based on $\lambda = 3.6$, and p_h based on $\lambda = 4.4$ (indicated by the red vertical dotted lines). The dashed green curve shows the performance of the optimal static-pricing corresponding to the actual λ.
We generated the plots for a single-node network, following the model described in [9]; in particular, we use demand value-distribution $F \sim Exponential(0.5)$, and driver reservation-value distribution $G_d \sim Exponentia(0.8)$.

Optimal Performance of Pricing Policies: One surprising aspect of Figure 2(a) is that the optimal throughput in the large-market limit over the dynamic pricing policies we consider appears to coincide with that obtained under static pricing. The authors of [9] show, however, that this fact is true for *all* threshold dynamic pricing policies under fairly weak conditions: in particular, they prove that *as long as all passenger value distributions F_{ij} have an increasing hazard rate, then the optimal normalized throughput in the large-market limit under dynamic (i.e., local state-dependent) pricing collapses to that obtained under the optimal static (i.e., state-independent) pricing policy.* This combined with the results in [8] shows that *the platform cannot improve performance much by employing state-dependent pricing.* Similar results are shown in [9] for revenue and welfare, and also for multi-threshold pricing policies.

We note here that the result given above is asymptotic; the plots in Figure 2(a) clearly show that dynamic pricing does have gains over static pricing for smaller values of the scaling parameter ν. The non-trivial aspect is that the difference in performance vanishes in the limit. Note also that the performance of a dynamic pricing policy with prices (p_ℓ, p_h) is not identical to the performance with static price p_ℓ or p_h, and in fact, it can be shown that passengers experience both prices in the large-market limit.

Robustness of Pricing Policies: More importantly, we note that the result that optimal static pricing and optimal dynamic pricing are asymptotically equivalent requires a key assumption: that the platform has knowledge of system parameters (e.g., the exogenous arrival rates of drivers and passengers, and distributions of reservation values). What should the platform do when these parameters are not well-known, and may even be highly variable?

To address this issue, the second main result in [9] establishes a significant benefit that dynamic pricing holds over static pricing: *robustness*. Specifically, the authors of [9] show that if the system operator chooses the optimal threshold dynamic (resp., static) pricing policy assuming some predicted system parameters $\bar{K}_0, \lambda$, but the true parameters deviate from the predictions, then *dynamic pricing maintains a much higher share of the optimal throughput relative to the optimal static pricing*. This property is graphically depicted in Figure 2(b); for a more formal characterization of this property, we refer the reader to [9].

5 Related Literature

In this section, we summarize the intellectual foundations our work builds on, as well as provide a brief survey of the growing literature on ridesharing across many fields. One of the great attractions of ridesharing is that the underlying questions lie at the intersection of several disciplines - economics, stochastic modeling and control, operations management, and network algorithms. The models and algorithms we have covered in this chapter borrows ideas from all these disciplines, and in a sense, we believe such a merging of ideas is critical to understanding these platforms. On the other hand, the diversity of disciplines makes it difficult to properly reference and comment on all the work related to this topic, and thus we acknowledge at the outset that our discussion below should be viewed as a survey of the main issues, rather than a comprehensive index of all relevant research.

Queueing networks and stochastic control. Our model for the fast-timescale dynamics builds on the rich toolbox of *queueing network models*, starting from the seminal work of Jackson [31] on open networks (i.e., where the number of units can change), and Gordon and Newell [23] and Baskett et al. [11] on closed networks (where the number of units is fixed, as in our setting). An excellent survey of these models is provided in the books by Kelly [34], Kelly and Yudovina [33] and Serfozo [47]. More recently, these models have found extensive use in several applied disciplines, including in the design of communications networks [49] and computer systems [28]. A more recent line of work develops a similar theory for *matching queues*, obtaining surprising product form characterizations [1, 51, 37].

Optimal control of open queueing networks also has a long history, going back to the work of Whittle [54], and more recently, these ideas have been extended to open matching networks [15, 39]. However, there is much less work for closed networks. This in part due to the lack of a closed-form expression for the normalization constant (though in our setting, it is computable in $O(nm)$ time via iterative techniques [16, 42]). Consequently, most previous work on closed queueing networks used heuristics, with limited or no guarantees. In particular, heuristics based on ensuring 'fairness' properties (which are similar to the circulation constraints we discuss in Section 4.1) have been used in transportation setting to optimize weighted throughput [21] and minimize rebalancing costs [55].

The first formal approximation guarantees for control of closed networks was given by Waserhole and Jost [52], who derived a pricing policy for maximizing throughput; this result is a special case of the results of [8] which we present in Section 4.1. Other recent works [14, 40] have formally characterized the large-market limits for closed queueing networks, and proposed asymptotically optimal rebalancing and dispatch policies (these results can however be derived directly using the techniques in [8]).

Parallel to the work on control, there has also been a long line of research on strategic behavior in queueing systems; see [38] for early work in this area and [30] for an overview of these models. Typically, these works consider systems with a fixed number of servers, who serve arriving customers who are sensitive to price and delay. In contrast, our model considers strategic behavior on the part of the servers (i.e., the drivers). In this respect our treatment is closer to the recent work on queues with strategic servers [24, 22].

Economics of two-sided platforms. From an economics standpoint, the strategic aspects of our micro-foundations are in the spirit of the literature on the price theory of two-sided platforms [43, 17, 44, 4, 5]; refer to [53] for an excellent summary and unification of this literature. This line of work typically studies the design two-sided markets under exogenously specified utility functions for agents. One critical difference in our approach is in building up the market model from the underlying stochastic dynamics, rather than specifying it exogenously. This is critical as having a dynamic model allows us to study dynamic pricing, which is one of the hallmarks of ridesharing platforms.

Stylized market models like the ones referenced above have been started being used for studying ridesharing as well. In particular, several recent works study the impact of spatial and temporal variations in prices on driver decisions in the fast-timescale [18, 12], a topic which we have not covered in this chapter. On the other hand, there is less work on using these to study inter-firm competition; one partial move in this direction is the work of [46], which studies the additional societal costs of multi-firm ridesharing ecosystems under exogenous demand fragmentation.

An important difference in our treatment of the ridesharing marketplace is that we incorporate both stochastic dynamics and strategic interactions. Doing so is challenging as one needs to reason about market equilibria under a combination of dynamics and strategic interactions. One important approach that helps circumvent this is that of using *large-market limits*; this is the approach we adopt in Section 4.2, following the treatment in [9]. Large-market limits have grown in importance in recent years; see [35, 7] for examples of this in the matching market literature, and [6] for an application of this approach for dynamic matching markets.

Pricing and operations management. A unique feature of ridesharing firms is that it is one of very few marketplaces where the platform can set the prices. Consequently, the study of such platforms shares commonalities to that of monopolist pricing. In particular, the comparison of static and dynamic pricing policies is a core topic of the literature on revenue management; refer [50, 13] for an overview of pricing approaches, and [20] for an analysis of dynamic pricing based on current inventory levels. More recent work applies techniques from approximate dynamic programming to tackle problems in logistics with dynamic arrivals and pricing [2, 36, 27]. Though similar to the fast-timescale control problem, these approaches typically can deal only with small systems, as their dimensionality scales rapidly with the number of stations; moreover, many of the techniques have no provable guarantees.

We note that though the results in the revenue management literature are similar in spirit to ours (in particular, the optimality of static pricing in a large-system limit), there is a very significant difference in the underlying settings. In particular, while the primary concern of monopolist pricing in a one-sided platform is to regulate demand in the face of changing inventory levels, the role of prices in ridesharing is to simultaneously regulate both instantaneous demand-supply mismatches and the entry decisions of drivers.

Data-driven simulations and empirical studies. Finally, in addition to the theoretical studies we mention above, there is also a parallel line of work which studies the same questions from a numerical perspective, either via data-driven simulation models, or experimental studies. Though such studies are of great importance, a big roadblock in this space is the lack of good data-sets and academic access to ridesharing platforms. An indicator of this state of affairs is the fact that several of the studies below were possible due to the New York City taxi dataset (`http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml`), which has played an important role as a public repository in this space. While we recognize the difficulties in data sharing, we emphasize that it is critical that the academic community works to cultivate publicly available data-sets that can be used as reference points for research in this area, across a range of platforms, geographies, and timescales.

In terms of data-driven simulation, a notable work is that of Resti et al. [45], which used the NYC taxi dataset to study the benefits of *pooling* – combining multiple rides into one. The work introduced the idea of compatible rides based on a diversion threshold constraint, where two ride-requests are considered to be amenable to pooling if the total origin-to-destination time (with diversions) for each ride is within an additive constant of the trip time without pooling. This idea proved influential in the subsequent design of Lyft Line and Uber Pool; moreover, it also spurred further research into the algorithmic challenges of determining such pooled rides in real-time and at scale [48, 3].

Empirical work on the internal dynamics of ridesharing platforms is challenging, as it depends on access to their proprietary data. As a consequence, most recent work has involved collaborations between external researchers and data scientists. Several of these study the behavior of dynamic pricing. For example, [25] presents a natural experiment that occurred when Uber's surge pricing algorithm failed over the New Year's Eve celebration in 2014-2015. In [19], the authors claim that dynamic pricing incentivizes more drivers to participate in the platform, and to meet supply shortages (analogous to the entry decisions described above in our analysis). The paper [26] highlights the difference in timescales in market equilibration: in the short run driver supply is relatively inelastic, but in the long run driver supply enters in response to persistently higher prices.

6 Conclusion

In this chapter, we have outlined a stochastic modeling framework for ridesharing platforms, which is based on the ideas presented in our prior work on this topic [9, 8], as well as work by several others on similar problems. In particular, in Section 3, we have presented this model in great detail, discussing all the assumptions that underlie this model, and arguing as to why these capture first-order phenomena in such platforms. We have also pointed out which aspects of these platforms lie outside our model.

One reason for our championing of this framework is its success in providing both theoretical insights and practical guidance into the design of pricing and dispatch policies on these platforms; we have summarized some of these results in Section 4. Our hope, however, is that this framework will go beyond the results presented to inspire and unify future studies into ridesharing platforms. To this end, we have outlined many open questions – in particular, there is a need for research into understanding the effect of driver strategic behavior on the fast timescale; the robustness of control policies to medium timescale changes in system parameters; the structure of efficient policies for ride pooling; the role of autonomous vehicles in the ridesharing landscape; the design of longer-term contracts for drivers and passengers; the interaction between multiple ridesharing platforms and between ridesharing and public transit; and the overall impact of ridesharing on society as a whole. These are difficult problems, and may not have any clear-cut answer – however, we hope the framework we have presented here will provide a benchmark for studying all these questions. We eagerly look forward to future research on these topics.

Acknowledgements The authors would like to thank the data science team at Lyft, particularly Chris Sholley; part of this work was carried out when SB was a technical consultant at Lyft. We gratefully acknowledge support from the National Science Foundation via grants CMMI-1234955 and CNS-1343253, the DARPA GRAPHS program, and Army Research Office grant W911NF-17-1-0094.

References

1. Adan, I., Weiss, G.: Exact fcfs matching rates for two infinite muti-type sequences. Operations Research **60**, 475–489 (2012)
2. Adelman, D.: Price-directed control of a closed logistics queueing network. Operations Research **55**(6), 1022–1038 (2007)
3. Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., Rus, D.: On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. Proceedings of the National Academy of Sciences **114**(3), 462–467 (2017)
4. Armstrong, M.: Competition in two-sided markets. The RAND Journal of Economics **37**(3), 668–691 (2006)
5. Armstrong, M., Wright, J.: Two-sided markets, competitive bottlenecks and exclusive contracts. Economic Theory **32**(2), 353–380 (2007)
6. Arnosti, N., Johari, R., Kanoria, Y.: Managing congestion in decentralized matching markets. Available at SSRN 2427960 (2014)
7. Azevedo, E.M., Budish, E.: Strategy-proofness in the large as a desideratum for market design. In: Proceedings of the 13th ACM Conference on Electronic Commerce, pp. 55–55. ACM (2012)
8. Banerjee, S., Freund, D., Lykouris, T.: Pricing and optimization in shared vehicle systems: An approximation framework. In: Proceedings of the 2017 ACM Conference on Economics and Computation, pp. 517–517. ACM (2017)
9. Banerjee, S., Johari, R., Riquelme, C.: Pricing in ride-sharing platforms: A queueing-theoretic approach. In: Proceedings of the 2015 ACM Conference on Economics and Computation, pp. 517–517. ACM (2015)
10. Banerjee, S., Kanoria, Y., Qian, P.: The value of state dependent control in ride-sharing systems. arXiv preprint arXiv:1803.04959 (2018)
11. Baskett, F., Chandy, K.M., Muntz, R.R., Palacios, F.G.: Open, closed, and mixed networks of queues with different classes of customers. Journal of the ACM (JACM) **22**(2), 248–260 (1975)
12. Bimpikis, K., Candogan, O., Daniela, S.: Spatial pricing in ride-sharing networks. Available at SSRN 2868080 (2016)
13. Bitran, G., Caldentey, R.: An overview of pricing models for revenue management. Manufacturing & Service Operations Management **5**(3), 203–229 (2003)
14. Braverman, A., Dai, J., Liu, X., Ying, L.: Empty-car routing in ridesharing systems. arXiv preprint arXiv:1609.07219 (2016)
15. Bušic, A., Meyn, S.: Optimization of dynamic matching models. arXiv preprint (arXiv:1411.1044) (2014)
16. Buzen, J.P.: Computational algorithms for closed queueing networks with exponential servers. Communications of the ACM **16**(9), 527–531 (1973)
17. Caillaud, B., Jullien, B.: Chicken & egg: Competition among intermediation service providers. RAND journal of Economics pp. 309–328 (2003)
18. Castillo, J.C., Knoepfle, D., Weyl, G.: Surge pricing solves the wild goose chase. In: Proceedings of the 2017 ACM Conference on Economics and Computation, pp. 241–242. ACM (2017)
19. Chen, M.K., Sheldon, M.: Dynamic pricing in a labor market: Surge pricing and flexible work on the uber platform. In: Proceedings of the 2016 ACM Conference on Economics and Computation, p. 455. ACM (2016)
20. Gallego, G., Van Ryzin, G.: Optimal dynamic pricing of inventories with stochastic demand over finite horizons. Management science **40**(8), 999–1020 (1994)
21. George, D.K.: Stochastic modeling and decentralized control policies for large-scale vehicle sharing systems via closed queueing networks. Ph.D. thesis, The Ohio State University (2012)
22. Gopalakrishnan, R., Doroudi, S., Ward, A.R., Wierman, A.: Routing and staffing when servers are strategic. Operations Research **64**(4), 1033–1050 (2016)

23. Gordon, W.J., Newell, G.F.: Closed queuing systems with exponential servers. Operations research **15**(2), 254–265 (1967)
24. Gurvich, I., Lariviere, M., Moreno, A.: Staffing service systems when capacity has a mind of its own. Available at SSRN 2336514 (2014)
25. Hall, J., Kendrick, C., Nosko, C.: The effects of ubers surge pricing: A case study. The University of Chicago Booth School of Business (2015)
26. Hall, J.V., Horton, J.J., Knoepfle, D.T.: Labor market equilibration: Evidence from uber. Tech. rep., Working Paper, 1–42 (2017)
27. Hampshire, R.C., Massey, W.A., Wang, Q.: Dynamic pricing to control loss systems with quality of service targets. Probability in the Engineering and Informational Sciences **23**(02), 357–383 (2009)
28. Harchol-Balter, M.: Performance modeling and design of computer systems: queueing theory in action. Cambridge University Press (2013)
29. Hartline, J.D.: Mechanism design and approximation. Book draft. October **122** (2013)
30. Hassin, R., Haviv, M.: To queue or not to queue: Equilibrium behavior in queueing systems, vol. 59. Springer Science & Business Media (2003)
31. Jackson, J.R.: Jobshop-like queueing systems. Management science **10**(1), 131–142 (1963)
32. Kallenberg, O.: Foundations of modern probability. Springer Science & Business Media (2006)
33. Kelly, F., Yudovina, E.: Stochastic networks, vol. 2. Cambridge University Press (2014)
34. Kelly, F.P.: Reversibility and stochastic networks. Cambridge University Press (1979)
35. Kojima, F., Pathak, P.A.: Incentives and stability in large two-sided matching markets. The American Economic Review pp. 608–627 (2009)
36. Levi, R., Radovanovic, A.: Provably near-optimal lp-based policies for revenue management in systems with reusable resources. Operations Research **58**(2), 503–507 (2010)
37. Moyal, P., Busic, A., Mairesse, J.: A product form and a sub-additive theorem for the general stochastic matching model. arXiv preprint arXiv:1711.02620 (2017)
38. Naor, P.: The regulation of queue size by levying tolls. Econometrica: journal of the Econometric Society pp. 15–24 (1969)
39. Nazari, M., Stolyar, A.L.: Optimal control of general dynamic matching systems. arXiv preprint arXiv:1608.01646 (2016)
40. Ozkan, E., Ward, A.R.: Dynamic matching for real-time ridesharing. Available at SSRN 2844451 (2016)
41. Ramsey, F.P.: A contribution to the theory of taxation. The Economic Journal **37**(145), 47–61 (1927)
42. Reiser, M., Lavenberg, S.S.: Mean-value analysis of closed multichain queuing networks. Journal of the ACM (JACM) **27**(2), 313–322 (1980)
43. Rochet, J.C., Tirole, J.: Two-sided markets: a progress report. The RAND Journal of Economics **37**(3), 645–667 (2006)
44. Rysman, M.: The economics of two-sided markets. The Journal of Economic Perspectives pp. 125–143 (2009)
45. Santi, P., Resta, G., Szell, M., Sobolevsky, S., Strogatz, S.H., Ratti, C.: Quantifying the benefits of vehicle pooling with shareability networks. Proceedings of the National Academy of Sciences **111**(37), 13,290–13,294 (2014)
46. Séjourné, T., Samaranayake, S., Banerjee, S.: The price of fragmentation in mobility-on-demand services. arXiv preprint arXiv:1711.10963 (2017)
47. Serfozo, R.: Introduction to Stochastic Networks, vol. 44. Springer Science & Business Media (1999)
48. Spieser, K., Samaranayake, S., Gruel, W., Frazzoli, E.: Shared-vehicle mobility-on-demand systems: a fleet operators guide to rebalancing empty vehicles. In: Transportation Research Board 95th Annual Meeting, 16-5987 (2016)
49. Srikant, R., Ying, L.: Communication networks: an optimization, control, and stochastic networks perspective. Cambridge University Press (2013)
50. Talluri, K.T., Van Ryzin, G.J.: The theory and practice of revenue management, vol. 68. Springer Science & Business Media (2006)

51. Visschers, J., Adan, I., Weiss, G.: A product form solution to a system with multi-type jobs and multi-type servers. Queueing Systems **70**(3), 269–298 (2012)
52. Waserhole, A., Jost, V.: Pricing in vehicle sharing systems: optimization in queuing networks with product forms. EURO Journal on Transportation and Logistics pp. 1–28 (2014)
53. Weyl, E.G.: A price theory of multi-sided platforms. The American Economic Review pp. 1642–1672 (2010)
54. Whittle, P.: Scheduling and characterization problems for stochastic networks. Journal of the Royal Statistical Society. Series B (Methodological) pp. 407–428 (1985)
55. Zhang, R., Pavone, M.: Control of robotic mobility-on-demand systems: a queueing-theoretical perspective. The International Journal of Robotics Research **35**(1-3), 186–203 (2016)