Pricing in Ride Sharing Platforms: Static vs Dynamic Strategies

Vivek Nagraj Pandit, School of Computer Science, Carnegie Mellon University, USA
Datar Mandar, INRIA & LIA, University of Avignon, France
Manjesh K. Hanawal, Dept. of IEOR, IIT Bombay, India
Sharayu Moharir, Dept. of EE, IIT Bombay, India

***Abstract*—We study pricing strategies in two-sided ride-sharing platforms that facilitate transactions between drivers and customers. In the setting we consider, the platform announces the price of service and customers/drivers react to this price. We model the two-sided market using a queuing framework where customers and drivers arrive into two separate queues and wait to be matched by the platform. The arrival process is stochastic and the rate of arrival depends on the instantaneous price set by the platform. On the completion of service, the platform keeps a fraction of the price paid by the customer as its commission and gives the rest of it to the worker who served this customer. Since the arrival processes as well as the platform's commission depends on the price per transaction set by the platform, the platform's revenue is a function of the pricing strategy. Our goal is to characterize pricing strategies which maximize the platform's revenue.**

We focus on two classes of pricing strategies, namely, static and dynamic. Static strategies are those where the price per transaction is independent of the number of customers/drivers waiting in the queue. Dynamic strategies allow for the price per transaction to be a function of the number of customers/drivers queued in the system. We characterize and compare the platform's revenue under optimal static and dynamic pricing strategies.

I. INTRODUCTION

In recent years, the sharing economy has undergone extraordinary growth in many fields, including ride-sharing with services such as Lyft, Ola and Uber. The main role of ride-sharing platforms is to facilitate transactions between drivers, who are willing to give rides, and customers, who are looking for rides. Typically, the drivers using these platforms work freelance, i.e., they are not employed by the platform. The platform pays a fixed fraction of the money paid by the customer, to the driver for providing the service, and retains the remaining fraction as its commission for facilitating the transaction. The price per transaction is determined by the platform and in this work, we focus on designing pricing strategies for such ride-sharing platforms.

For simplicity, we focus on customers and drivers arriving in a specific geographical area and assume all rides are of the same type (no class differentiation). We model the ride-sharing service using a pair of queues – a customer queue and a driver queue. The arrivals to these queues are modeled as exogenous stochastic processes whose rates depend on the instantaneous price per transaction set by the platform. Whenever possible, the platform matches a queued customer with a queued driver for service and the two matched users immediately leave the queue. Higher prices attract more drivers whereas lower prices attract more customers, and therefore, the revenue generated by the service depends heavily on the price set by the platform.

Ride-sharing platforms in most geographical areas are typically oligopolies with a small number of firms holding a large chunk of the market share. Often, demand curves in oligopolies are concave [1], [2], [3]. Motivated by this, we consider a single ride-sharing platform and work in a setting where the rate of arrival of customers into the system is a decreasing and concave function of the price set by the platform while it is increasing for the drivers. Similar setting has also been studied in [4].

The main focus of this work is on comparing the performance of two types of pricing strategies, namely static and dynamic. We call a pricing strategy static if the price per transaction is independent of the state of the system, i.e., the number of customers/drivers currently queued. In contrast, a strategy in which the price per transaction is a function of the current state of the system is referred to as a dynamic pricing strategy.

While it is reasonable to expect that dynamic pricing strategies outperform static pricing strategies due to the added flexibility, [5] shows that this is not always the case. In their ride-sharing model, static pricing policies perform just as well as dynamic pricing polices with respect to the goal of revenue maximization. We believe that this is a consequence of their modeling assumption that drivers do not react to instantaneous prices set by the platform and are instead focused only on their long term time-average earnings. In contrast, in our setting, the rate of arrival of drivers into the system increases with the price set by the platform. This is a key modeling difference between our work and [5]. The key takeaway from this work is that, dynamic pricing can indeed strictly outperform static pricing strategies if both the customers and drivers decisions are based on the instantaneous price. In practice it is often the case that the drivers and customer react to the instantaneous price as observed in the case study of UBER's surge pricing. [6].

A. Related Work

The design of sharing platforms with two-sided markets is a challenging task as it requires optimal combination of economics and engineering. This challenging task has been addressed by many researchers [7], [8]. Ride-sharing platform

978-1-5386-7902-9/19/$31.00

is a type of two-sided market and its increasing popularity has led to an increased amount of research activity in this field.

In [5], [9], [10] a queuing model is proposed to study optimal platform pricing. The design of pricing and subsidies for improving the trade-off between revenue and social welfare is discussed in [11], where the authors employ game theory to prove the existence of a unique Nash equilibrium regardless of prices imposed by the ride-sharing platform.

Several works [8], [12], [7] focus on the economics of two-sided markets and discuss the challenges in such markets using various examples. The challenges and open ended problems in matchmaking, in multi-sided markets are discussed in [13]. A graph theoretic model is used in [14], where each node represents a particular region in a city. The authors in [15] study matching of supply and demand in both space and time using a convex programming formulation to find the optimal pricing policy.

Our work focuses on state dependent optimal pricing. We derive the optimal pricing for both the static and dynamic setting. For the dynamic case, we characterized the optimal price using an Markov Decision Process approach.

B. Organization

The rest of this paper is organized as follows. In Section I we introduce the Markov model and define the related state space, action space and the reward function. Models for both discrete and continuous setting are introduced. Static pricing and dynamic pricing policies for Poisson as well as Bernoulli distributed arrivals, of drivers and customers, are discussed in Section III and IV respectively. In Section V we analyze and compare the revenue obtained from the static and dynamic pricing. In Section VI we discuss performance metrics the platform would like to consider in addition to revenue. Conclusions are given in Section VII.

II. Setting

We focus on a system consisting of a platform and two types of users. We refer to these users as drivers and customers. The platform facilitates transactions between the drivers and the customers by matching them.

A. System State and Evolution

The platform maintains two queues, a customer queue and a driver queue. The customer and driver queues have finite buffers of size N_c and N_d respectively. Any customer or driver arriving to the queue when the corresponding buffer is full are dropped from the system.

1) Customer and Driver Arrivals: Customers and drivers arrive into the system according to a stochastic process. The rate of arrival of drivers and customers to their respective queues is a function of the price of transaction set by the platform at that time. We consider the following two arrival models.

Model 1 (Bernoulli Arrivals): Time is divided into slots. In each time-slot, the number of arrivals to both queues are Bernoulli random variables. At the end of each time-slot, drivers and customers are matched for the service(ride).

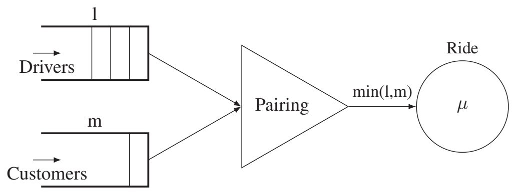

Fig. 1: Queuing model of the platform for Bernoulli arrival. l and m represents current length of driver and customers queues and $min(l,m)$ are the matched in each slot. System state is $l-m$.

Matched customers and drivers depart from their respective queues and any unmatched drivers/customers remain queued up for the next time-slot.

Model 2 (Poisson Arrivals): Customers and drivers arrive to the respective queues according to a Poisson process. At any instant, if there is at least one customer and at least one driver in the two queues, the customer is immediately matched to the driver and they depart from their respective queues.

2) State Space: Given that queued customers and drivers are matched as soon as possible, at most one of the two queues is non-empty at a given time (at the end of the time-slot in the discrete-time system). The state of the system can then be uniquely represented by the difference of the customer queue-length and the driver queue-length. Since the customer and driver queues have finite buffers of size N_c and N_d respectively, this difference can take on all integer values from $-N_d$ to N_c. The set of states is denoted by S and a generic state by s.

B. Pricing and Revenue

The platform decides the price of a transaction. The price is constrained to lie in the interval $[p_{mn}, p_{mx}]$. As mentioned before, the rate of arrival of drivers and customers to their respective queues is a function of the price set by the platform at that time. The arrival rate of customers is a decreasing and concave function of the price and the arrival rate of drivers is an increasing function of the price. Once matched, the driver provides a ride to the customer. We assume that all rides are equal. At the end of the ride, the customer pays the platform for the ride. The platform keeps a fixed fraction of this price and gives away the rest of it to the driver. The money collected by the platform is called its revenue and the platform's goal is to maximize its long term average revenue. We consider two pricing strategies, namely static and dynamic pricing, defined as follows.

Definition 1: Static Pricing Strategies: The price set by the platform is fixed, i.e., independent of the system state.

Definition 2: Dynamic Pricing Strategies: The price set by the platform is a function of the state of the system.

Let s denote the state, $p(s)$ denote the price per unit distance at state s, γ denote the fraction of the money paid by the customers, that is given to the driver by the platform, $M(s)$ denote the expected rate of matching a customer to a driver when the system is in state s. The distance traveled by a customer is denoted by the random variable X which is assumed to be independent of the price.

For a given pricing policy, the expected rate of revenue when the system is in state s is given by

$$r(s,p) = (1-\gamma)M(s)p(s)\mathbb{E}[X]. \quad (1)$$

Without loss of generality we set $\mathbb{E}[X] = 1$ and suppress the constant factor $(1-\gamma)$ for notational convenience. The platform aims to maximize the expected time-averaged revenue given by

$$R_{N_c,N_d}(p) = \mathbb{E}[r(s,p)] = \sum_{-N_d}^{N_c} r(s,p)\pi(s). \quad (2)$$

where $\pi(\cdot)$ denotes the stationary distribution of the Markov chain under the given pricing strategy.

III. Static Pricing

In this section, we focus on static pricing policies where the price of service(ride) is independent of the state of the system. The goal is to find the price which maximizes the long term expected (rate of) revenue. We study the two arrival models (Models 1 and 2) separately.

A. Model 1: Bernoulli Arrivals

In this subsection we assume that the platform matches queued drivers and customers at discrete time intervals. The length of the intervals are small and constant. The customer and driver arrivals in each time-slot are Bernoulli distributed. For a given price p, let $q_c(p)$ and $q_d(p)$ denote the probability of a customer and driver arrival, respectively, in a time-slot. For notational convenience, we drop the dependence of probability of arrival(of both customer and driver) on p and simply write them as q_c and q_d. Recall that the state of the system, denoted by s, takes value in the set S. The transition probabilities of the Markov chain is as shown below.

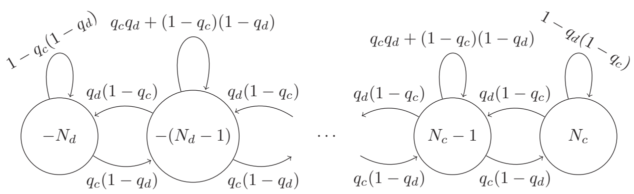

Fig. 2: Transition probabilities of the Markov Chain

We make the following assumptions on the arrival rates.

Assumption 1: $q_d(p)$ and $q_c(p)$ are continuous functions in p, taking values in $(0,\ 1)$. $q_d(p)$ is an increasing function, and $q_c(p)$ is a decreasing and concave function, in p.

One can compute the transition probabilities of the Markov chain explicitly and find the the expected revenue given as follows. Proof details are given in the appendix.

Lemma 1: $R_{N_c,N_d}(p)$ for model 1 is given by

$$\frac{p}{1-\alpha^{N_c+N_d+1}}\Big((1-\alpha^{N_d})q_c + \alpha^{N_d+1}(1-\alpha^{N_c})q_d + \alpha^{N_d}(1-\alpha)q_cq_d\Big), \quad (3)$$

where $\alpha := \alpha(p) = \frac{q_c(1-q_d)}{q_d(1-q_c)}$.

Though $R_{N_c,N_d}(p)$ is in a closed form, its dependence on p is not clear. We plot it in Figure 3 as a function of p for the case $N = N_c = N_d$. As seen, the value of $R_{N,N}(p)$ is bounded by the values of both q_cp and q_dp and closely follows $\min\{q_cp, q_dp\}$, which is unimodal in p. The unimodal property can be easily verified by taking derivative and noticing that the root of the derivative is unique. To simplify notations we write $R_N(p) := R_{N,N}(p)$. The following results give tight bounds on $R_N(p)$ in terms of its unimodal approximation.

Lemma 2: Under Model 1, $R_N(p)$ is bounded as follows:

$$\min\{q_c, q_d\}p - \frac{\alpha^N(1-\alpha)p}{4(1-\alpha^{2N+1})}) \le R_N(p) \le \min\{q_c, q_d\}p.$$

Further, $\lim_{N\to\infty} R_N(p) = \min\{q_c, q_d\}p$ and

$$\lim_{N\to\infty} R_N(p) = \begin{cases} q_dp & \text{for } \alpha > 1 \\ q_cp & \text{for } \alpha \le 1. \end{cases}$$

Theorem 3: Let $N_d = N_c = N$. For all $\epsilon > 0$, there exists $N_\epsilon > 0$ such that

$$\min\{q_c, q_d\}p - \epsilon \le R_N(p) \le \min\{q_c, q_d\}p$$

for Model 1. Specifically, we can set $N_\epsilon = \frac{p_{mx}}{8\epsilon} - \frac{1}{2}$ for a given $\epsilon > 0$.

N_ϵ indicates the value of N at which $R_N(p)$ is within ϵ approximation of its maximum possible value. In the following we assume that N is sufficiently large such that $R_N(p)$ satisfies the unimodal property, in accordance with its asymptotic approximation, i.e., $p\min\{q_c, q_d\}$. The value of N indicates the capacity of the platform to manage large queues. Increasing N results in marginal increase in optimal revenue generated but also increases operational costs. Hence a trade off is to be maintained. Next, we compute the optimal static price.

Let $\bar{p}$ be the price at which $\alpha = 1$ and denote $\bar{q} = q_c(\bar{p}) = q_d(\bar{p})$. As $\alpha \to 1$, one can simplify the expected reward in Lemma 1 to show that

$$R_N(\bar{p}) > (\bar{p}\bar{q} - \frac{\bar{p}}{4}). \quad (4)$$

By the unimodal property of $R_N(p)$, its maxima, denoted by p^*, is such that $R_N(p^*) \ge \bar{p}(\bar{q} - \frac{1}{4})$. Hence, $p^* \in [p_1, p_2]$ where p_1, p_2 satisfy $q_cp_1 = \frac{\bar{p}}{4}$ and $q_dp_2 = \frac{\bar{p}}{4}$. Then, we can then do a binary search in the interval $[p_1, p_2] \subseteq [p_{mn}, p_{mx}]$ to find p^*. An algorithm to find the optimal price is given below.

The algorithm is easy to understand for the case of linear functions of arrival rates. For the purpose of understanding, assume that $q_c = a - bp$ and $q_d = c + dp$ for some positive constants a, b, c, d. Then q_cp achieves a unique maxima at

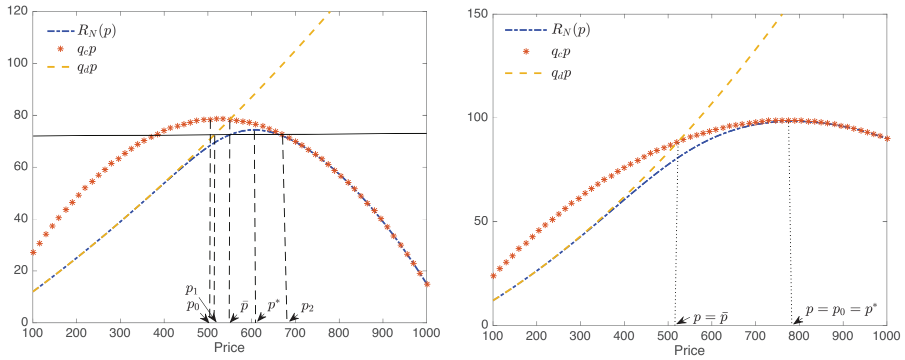

Fig. 3: $R_N(p)$ vs price. Left figure depicts the case $\bar{p} < p_0$ and the right figure the case $\bar{p} < p_0$. $N = 2$ and arrival rate of drivers and customers is set as a linear function in price.

Result: p^* –optimal price of $R_N(p)$
Initialization: Let $p_o = \max_p q_c p$, Find $\bar{p}$ such that $q_c(\bar{p}) = q_d(\bar{p})$ and set $\bar{h} = \bar{p}\bar{q} - \frac{\bar{p}}{4}$
if $p_o > \bar{p}$ **then**
  $p^* = p_o$;
else
  Solve for $q_c(p)p = \bar{h}$(take the larger root) and $q_d(p)p = \bar{h}$ to get p_1 and p_2 respectively. ;
  Do a binary search on $[p_1$,$p_2]$ to obtain p^*
end

Algorithm 1: Find optimal price in static Bernoulli case

$p_0 = a/2b$. $\bar{p}$ is simplified to $\bar{p} = (a-c)/(b+d)$. Consider the two cases corresponding to $p_0 > \bar{p}$ and $p_0 \leq \bar{p}$.

1) $p_0 > \bar{p}$ (see left Fig. 3): Under this condition $p^\star = p_0$. This is because when $\alpha > 1$, the function $q_d p$ is monotonically increasing with p (as both q_d and p are individually increasing functions of p and both are positive). When $\alpha \leq 1$, the function $q_c p$, is still increasing from $p = \bar{p}$ to $p = p_0$, and then decreasing. Since $R_N(p)$ follows exactly the nature of these functions for every value of α (for large N), $R_N(p)$ will also have its maxima at $p = p_0$.
2) $p_0 \leq \bar{p}$ (see right Fig. 3): Under this case $p^\star \in [p_1, p_2]$. This is because when $\alpha > 1$, function $q_d p$ is increasing monotonically and reaches its maxima at $p = \bar{p}$. Hence for $p \leq \bar{p}, R_N(p)$ is monotonically increasing. When $\alpha \leq 1$, $q_c p$ is a decreasing function of p because it has already reached its unique maxima at $p = p_0 < \bar{p}$. Hence for $p > \bar{p}$, the function $R_N(p)$ is decreasing. This results in $R_N(p)$ having its maxima between p_1 and p_2.

The above procedure works as long as maxima of $q_c p$ is unique. Under our assumptions that q_c is concave and decreasing, $q_c p$ has a unique maxima and the above algorithm finds the optimal price.

B. Model 2: Poisson Arrivals

In this section we model the system as a continuous time queuing process. Potential drivers and customers arrive according to Poisson processes with rates λ_{d0} and λ_{c0} respectively. Out of these potential drivers and customers, only some choose to join the system, based on the current price of transaction set by the platform. Let $q_d(p)$ and $q_c(p)$ denote the probability that an incoming driver and customer, respectively, joins the queue when the price is set to p. The effective rate of driver and customer arrivals into the queue is then given by $\lambda_c(p) = q_c(p)\lambda_{c0}$ and $\lambda_d(p) = q_d(p)\lambda_{d0}$.

Transition Probability: The state of the system changes as soon as there is an arrival of either a customer or a driver. As inter arrival time in the Poisson process is exponentially distributed, the probability that the platform transitions from a state s to $s+1$ is same as the probability that a customer arrives before driver, i.e., $\frac{\lambda_c(p)}{\lambda_c(p)+\lambda_d(p)}$. Similarly, the probability that the platform transitions form state s to state $s-1$ is $\frac{\lambda_d(p)}{\lambda_c(p)+\lambda_d(p)}$. Then one can calculate all the transition probabilities of the embedded Markov chain, as shown in Figure 4. Under Assumptions 1, the embedded Markov chain is irreducible.

Reward: In state $s > 0$, the platform earns revenue if a driver arrives before a customer, otherwise platform moves to the next state $s+1$. Similarly, in state $s < 0$, the platform earns revenue if a customer arrives before a driver. Hence the 'rate' of reward as a function of state s at price p is given by

$$r(s,p) = \begin{cases} \frac{\lambda_d(p}{\lambda_c(p)+\lambda_d(p)} \times p & \text{if } s > 0 \\ \frac{\lambda_c(p)}{\lambda_c(p)+\lambda_d(p)} \times p & \text{if } s < 0 \\ 0 & \text{if } s = 0. \end{cases} \quad (5)$$

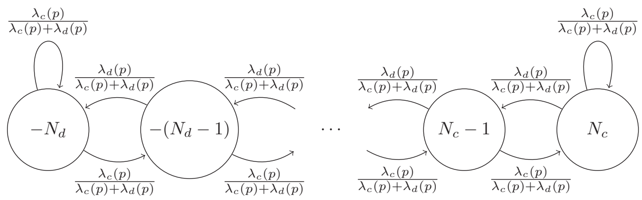

Fig. 4: Transition probability of the embedded Markov Chain for Poisson arrivals

The expected reward in (2) can then be computed as

$$R_{N_c,N_d}(p) = \frac{(1-\lambda_c(p)/\lambda_d(p))^{N_d+N_c}}{(1-\lambda_c(p)/\lambda_d(p))^{N_d+N_c+1}} \times \frac{\lambda_c(p)/\lambda_d(p)}{1+\lambda_c(p)/\lambda_d(p)}p. \tag{6}$$

Similar to the case of Model 1, $R_{N_c,N_d}(p)$ for Model 2 also asymptotically converges to $\frac{\min(\lambda_c,\lambda_d)}{\lambda_c+\lambda_d}p$ as N_c and N_d tend to infinity. Specifically, we have

Theorem 4: Let $N_d = N_c = N$, for all $\epsilon > 0$, there exists $N_\epsilon > 0$ such that

$$\frac{\min(\lambda_c,\lambda_d)}{\lambda_c+\lambda_d}p - \epsilon \le R_N(p) \le \frac{\min(\lambda_c,\lambda_d)}{\lambda_c+\lambda_d}p$$

for Model 2. Specifically,using linear rate functions, $q_c = a - bp$ and $q_d = c + dp$, we get

- If $c > a, N_\epsilon = \frac{1}{2}\left(\frac{ln(1+\frac{\sigma_2 p_{mx}}{\epsilon})}{ln(\frac{c}{a})} - 1\right)$
- If $c < a, N_\epsilon = \frac{1}{2}\left(\frac{ln(1-\frac{\sigma_1 p_{mx}}{\epsilon})}{ln(\frac{c}{a})} - 1\right)$
- If $c = a, N_\epsilon = \frac{1}{2}\left(\frac{p_{mx}(1-\gamma)}{2\epsilon} - 1\right)$,

where σ_1 and σ_2 are some fixed constants.

We can then use the same algorithm as in 1 to find the optimal price for the case with Poisson arrivals.

IV. Dynamic Pricing

In dynamic pricing, the price set by the platform depends on the state of the system. In this section we only focus on the discrete time setting with Bernoulli arrivals. Continuous time setting with Poisson arrivals can be analyzed similarly (see appendix). We assume that the price is set at the beginning of each time interval based on the state reached in the previous time slot. We model the problem as a discounted infinite horizon Markov Decision Problem (MDP) and use the value iteration algorithm to find the optimal policy [16]. We also find the optimal price for the average total cost criteria and compare the revenues from the optimal static and dynamic pricing strategies.

The state space, action set and transition probabilities remain the same as in the previous section. Let s_t denote the state of the system at time t. We define $D : [-N_d, N_c] \to [p_{mn}\ p_{mx}]$ as the decision rule which determines the price in a given state. Let $p_t = D(s_t)$ denote the price set at time t. We next study optimal policies for the total discounted and the average reward criteria.

A. Total discounted reward

For a given policy D and discount factor $0 < \beta < 1$, the total discounted reward starting from state $s_0 = s$ is,

$$v(s) := v^D_\beta(s) = \mathbb{E}^D\left\{\sum_{t=1}^{\infty} \beta^{t-1} r(s_t, p_t) | s_0 = s\right\}. \tag{7}$$

Let $P(s'|s,p)$ denote the transition probability from state s to s' when the platform sets a price p. The values of these probabilities are given in Fig. 2. Substituting the values of transition probabilities and reward, the Bellman's equation for the discounted reward are as follows:

$$v(s) = \begin{cases} \max\limits_{p\in[p_{mn}\ p_{mx}]} q_d p + \beta(q_d(1-q_c)v(s-1) \\ \quad +q_c(1-q_d)v(s+1) + (q_d q_c \\ \quad +(1-q_c)(1-q_d))v(s)) \qquad \text{for } 0 < s < N_c \\ \max\limits_{p\in[p_{mn}\ p_{mx}]} q_c p + \beta(q_d(1-q_c)v(s-1) \\ \quad +q_c(1-q_d)v(s+1) + (q_d q_c \\ \quad +(1-q_c)(1-q_d))v(s)) \qquad \text{for } -N_d < s < 0 \end{cases}$$

It is difficult to derive any definite structure of the optimal policy from the above value functions. But, one can use the standard value iteration algorithm[16] given below to find an optimal policy.

Result: $D^*(\cdot)$ – optimal policy for discounted reward
Initialization: $v^0(s) = 0, \forall s \in S, n = 0, \epsilon > 0$
1. For each $s \in S$ compute $v^{n+1}(s) = \max\limits_{p\in[p_{mn}\ p_{mx}]}\left\{r(s,p) + \sum_{s'\in S}\beta p(s'/s,p)v^n(s')\right\}$
if $|v^{n+1} - v^n| < \epsilon$ **then**
  For each $s \in S$ choose $d(s) \in \underset{p\in[p_{min}p_{max}]}{arg\ max}\left\{r(s,p) + \sum_{s'\in S}\beta p(s'/s,a)v^n(s')\right\}$
else
  $n = n + 1$ and go to step 1.
end

Algorithm 2: Value iteration algorithm

It is easy observe that optimal price in state $s = N_c$ is $p = p_{mx}$. However, no such structural claim is straightforward for other states due to complex relations of the value functions.

B. Average reward

For any policy D, the average reward is given by

$$R_N(D) = \lim_{T\to\infty} \frac{1}{T}\mathbb{E}^D\left\{\sum_{t=0}^{T-1} r(s_t, D(s_t))\right\}. \tag{8}$$

Let $\pi^D := \{\pi^D(s), s \in S\}$ denote the stationary policy associated with the dynamic policy D, then the average reward is given by

$$R_N(D) = \sum_{s\in S} \pi^D(s) r(s, D(s)).$$

The stationary probabilities for a given dynamic policy can be computed by writing a set of balance equations and solving them simultaneously. Note that any policy D can be represented by a vector $D := \{p_s : s \in S\}$, where $p_s = D(s) \in [p_{mn}\ p_{mx}]\ \forall\ s$. Then an optimal policy D^* can be obtained by solving

$$D^* = \arg \max_{D=\{p_s:s\in S\}} R(D) \tag{9}$$

$$s.t. \quad p_{mn} \le p_s \le p_{mx} \quad \forall \tag{10}$$

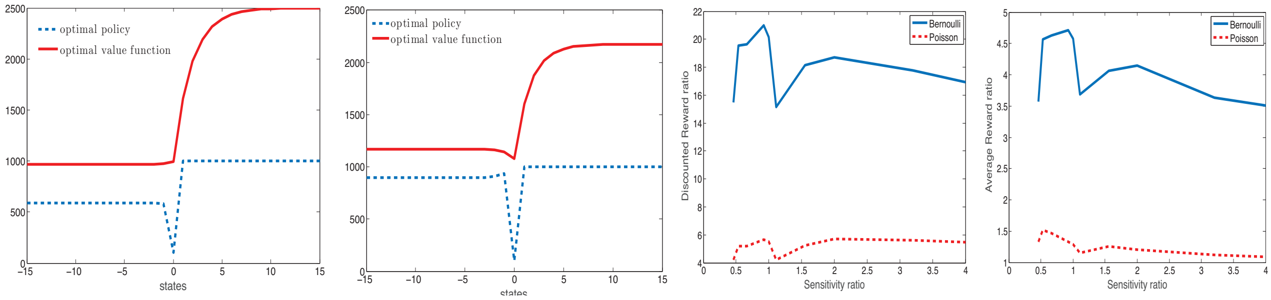

Fig. 5: Left to Right: First two plots show at optimal value function for a) Bernoulli and b) Poisson arrivals. In the last two plots show ratio optimal revenue with static and dynamic price vs sensitivity ratio for c) discounted and d) average rewards. We set $N = 15, p_{mn} = 100, p_{mx} = 1000, \beta = 0.7$. The arrival rates are linear in price.

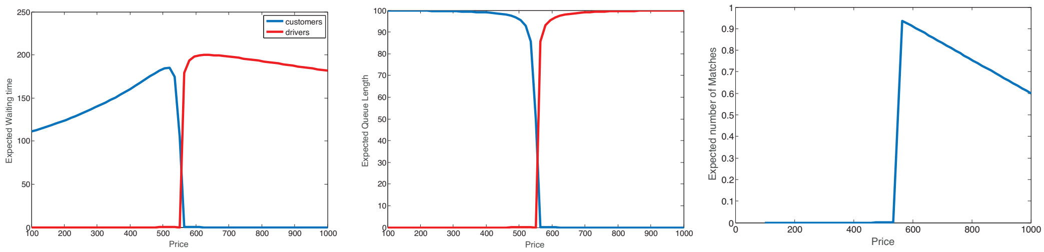

Fig. 6: First plot shows mean waiting time, second mean queue length and third mean number of matches vs price.

V. Numerical Results

In this section we compare the effects of static and dynamic pricing strategies on the expected revenue of the platform. For both Bernoulli and Poisson arrival models, optimal static price is computed as discussed in Section III, and the optimal dynamic policy as discussed in Section IV. The first two plots in Fig 5 show optimal prices in different states for linear arrival rates. As shown, under dynamic pricing, the price is higher when customers are in excess compared to the case when drivers are in excess. Thus the platform increases the price when number of customers exceed the number of drivers. This is analogous to the 'surge' pricing mechanism used by many ride-sharing platforms. As seen in the figures, the behavior of the optimal policy and corresponding revenue earned (value function) is similar for both Bernoulli and Poisson models.

The last two plots in Fig. 5 show the revenue improvements due to dynamic pricing strategy in comparison to that of static pricing strategy as a function of the price sensitivity of customers and drivers. Price sensitivity is defined in terms of the rate of change of q_c and q_d in p. Specifically, we set price sensitivity as the ratio $(|\delta q_c/\delta p|)/(|\delta q_d/\delta p|)$. The plots show the behavior for the linear arrival rates for which the sensitivity ratio is b/d. As sensitivity ratio becomes large i.e q_c falls at a rapid rate compared to increase in q_d, a change in unit price will result in drastic reduction in customer arrivals compared to increase in driver arrivals. This will result in reduced number of matches. Hence to prevent such a situation, the prices are kept low so as to increase customer arrivals relatively. This results in decrease in revenue and the reward ratio reduces when sensitivity ratio is high. This trend is reflected in the plots given above. For all values of sensitivity ratio, the revenue gain due to dynamic pricing is higher and the gain becomes significant when the price sensitivity is higher.

VI. Waiting Times and Matchings

In this section we discuss various other performance metrics like, expected waiting time for customers and drivers, and expected number of matches to study the effect of price.

Mean waiting time: Recall that whenever $s > 0$, the driver queue will be empty, and whenever $s < 0$, the customer queue will be empty. Then, the expected queue length of the drivers and customers denoted L_c and L_d, respectively, can be expressed as $L_c = \sum_{s>0} s\pi(s)$ and $L_d = \sum_{s<0} s\pi(s)$. Substituting the values of stationary distribution, we get expected queue lengths as

$$L_c = \frac{\alpha^{N+1}}{1-\alpha^{2N+1}}\left(\frac{1-\alpha^N}{1-\alpha} - N\alpha^N\right) \quad (11)$$

$$L_d = \frac{\alpha^{N+1} + N - \alpha - N\alpha}{(1-\alpha)^2}\left(\frac{1-\alpha}{1-\alpha^{2N+1}}\right). \quad (12)$$

By Little's law, the expected waiting times for customers and drivers denoted by W_c and W_d, respectively, can be computed as $W_c = L_c/q_d$ and $W_d = L_d/q_c$.

Mean rate of matching: When $s > 0$, $s = 0$ and $s < 0$, matching occurs at the rate of q_d,q_c and $q_c q_d$ respectively. Hence the mean rate of matching, denoted as M, is

$$M = q_c \sum_{s<0} \pi(s) + q_d \sum_{s>0} \pi(s) + q_c q_d \pi(0)$$
$$= \left(q_c + q_d\alpha^{N+1} + \frac{q_c q_d \alpha^N(1-\alpha)}{1-\alpha^N}\right)\frac{1-\alpha^N}{1-\alpha^{2N+1}} \quad (13)$$

As price increases, the arrival rate of drivers increases, resulting in increased queue length and waiting time of drivers. The customer arrival rate decreases because of which there is decreased queue length and waiting time for customers. These trends are reflected in the plots in Figure 6. When price is low, probability of staying in positive states i.e customers in queue is higher and driver arrival rates is lower. Hence the expected number of matching is quite low. The case is similar when price is very high. .

VII. Conclusions

In this work, we modeled a ride-sharing service in a single region using a queuing model. The service is run by a central platform which facilitates transactions between drivers and customers. The platform fixes the price of the service. Potential drivers and customers decide whether they are interested in using this service at that price. The goal of the platform is to maximize its revenue.

We characterized the performance of two classes of pricing policies, namely, static and dynamic policies. Through theoretical and numerical results, we conclude that if both customers and drivers react to the instantaneous price of service, dynamic pricing strategies outperform static pricing strategies. Our model can also be used to optimize other metrics such as the average waiting time/delay of customers and/or drivers.

VIII. Appendix

A. Proof of Lemma 1

Under Assumption 1 the Markov Chain is irreducible and hence a unique stationary distribution exists. We first find the stationary distribution of the Markov chain induced by the transition probabilities given in 2. For state $i = -N_d$, we have

$$\pi(-N_d) = \pi(-N_d+1)(q_d(1-q_c)+ \\ \pi(-N_d)(q_d q_c + (1-q_d)(1-q_c) + q_d(1-q_c))$$

Upon simplification, the above equation reduces to $\pi(-N_d+1) = \alpha\pi(-N_d)$ where $\alpha = \frac{q_c(1-q_d)}{q_d(1-q_c)}$. Applying the balance equations to other states, one can easily verify upon simplification that the above relation holds for all $s > -N_d$. Hence, for all $s \in S$

$$\pi(s) = \alpha\pi(s-1)$$

Using the fact that $\sum_{s\in S}\pi(s) = 1$ we get

$$\pi(-N_d) = \frac{1-\alpha}{1-\alpha^{N_c+N_d+1}},$$

or in more general terms

$$\pi(s) = \alpha^{s+N_d}\frac{1-\alpha}{1-\alpha^{N_c+N_d+1}} \qquad \forall s \in [-N_d, N_c]. \quad (14)$$

As expected number of matched at given price is $M(s) = q_d$ for $s > 0$, $M(s) = q_c$ for $s < 0$, and $M(s) = q_c q_d$ for $s = 0$, the reward is given by

$$r(s,p) = \begin{cases} pq_d & \text{if } s > 0 \\ pq_c & \text{if } s < 0 \\ pq_c q_d & \text{if } s = 0 \end{cases}$$

The expected reward in 2 simplifies to

$$R_{N_c,N_d}(p) = \sum_{s=-N_d}^{-1}\pi(s)q_c p + \sum_{s=1}^{N_c}\pi(s)q_d p + \pi(0)q_c q_d p$$

Substituting the values of the stationary distribution π and simplifying we get

$$R_{N_c,N_d}(p) = \frac{p}{1-\alpha^{N_c+N_d+1}}\Big((1-\alpha^{N_d})q_c + \\ \alpha^{N_d+1}(1-\alpha^{N_c})q_d + \alpha^{N_d}(1-\alpha)q_c q_d\Big) \quad (15)$$

B. Proof for Lemma 2

We have

$$R_N(p) = \frac{p}{1-\alpha^{2N+1}} \\ ((1-\alpha^N)q_c + \alpha^{N+1}(1-\alpha^N)q_d + \alpha^N(1-\alpha)q_c q_d))$$

For $\alpha < 1$, $\lim_{N\to\infty}\alpha^N = 0$ and hence $\lim_{N\to\infty} R_N(p) = pq_c$. For $\alpha > 1$, $\lim_{N\to\infty}\frac{\alpha^{N+1}(1-\alpha^N)}{1-\alpha^{2N+1}} = 1$ and hence $\lim_{N\to\infty} R_N(p) = pq_d$. For the case $\alpha = 1$, we can apply L'Hospital's rule to check that $\lim_{N\to\infty} R_N(p) = pq_c = pq_d$.

We first upper bound $R_N(p)$. Consider the following cases.

1) Let $\alpha > 1$, i.e., $q_c > q_d$. Subtracting both sides of Eqn. (15) from pq_d, we have

$$pq_d - R_N(p) = \frac{p}{1-\alpha^{2N+1}}(q_d(1-\alpha^{2N+1}) - (1-\alpha^N)q_c \\ - \alpha^{N+1}(1-\alpha^N)q_d - \alpha^N(1-\alpha)q_c q_d) \\ = p\alpha^N\left(\frac{(q_d-q_c)}{1-\alpha^{2N+1}}\right) > 0$$

Hence, $R_N(p) < pq_d \quad \forall\, \alpha > 1$

2) Let $\alpha < 1$, i.e., $q_c < q_d$, Subtracting both sides of 15 from pq_c we have

$$pq_c - R_N(p) = \frac{p}{1-\alpha^{2N+1}}(q_c(1-\alpha^{2N+1}) - (1-\alpha^N)q_c \\ - \alpha^{N+1}(1-\alpha^N)q_d - \alpha^N(1-\alpha)q_c q_d) \\ = p\alpha^N\left(\frac{\alpha^{N+1}(q_d-q_c)}{1-\alpha^{2N+1}}\right) > 0$$

Hence, $R_N(p) < pq_c \quad \forall\, \alpha < 1$.

3) Let $\alpha = 1$. By L'Hospital's rule it is easy to check that $\lim_{\alpha\to1} R_N(p) \le pq_c = pq_d$.

Hence $R_N(p)$ is upper bounded by $min(q_c, q_d)p$. To lower bound $R_N(p)$, again consider the following cases.

1) Let $\alpha \geq 1$

$$
\begin{aligned}
R_N(p) &= \frac{p}{1-\alpha^{2N+1}}((1-\alpha^N)q_c \\
&\quad + \alpha^{N+1}(1-\alpha^N)q_d + \alpha^N(1-\alpha)q_c q_d) \\
&\geq \frac{p}{1-\alpha^{2N+1}}((1-\alpha^N)q_d \\
&\quad + \alpha^{N+1}(1-\alpha^N)q_d + \alpha^N(1-\alpha)q_d^2) \\
&= pq_d - \frac{p\alpha^N(\alpha-1)q_d(1-q_d)}{\alpha^{2N+1}-1}
\end{aligned}
$$

where the inequality follows as $q_c \geq q_d$. Since the maximum value of $q_d(1-q_d)$ is $\frac{1}{4}$, we get

$$R_N(p) \geq pq_d - \frac{p\alpha^N(\alpha-1)}{4(\alpha^{2N+1}-1)} \qquad (16)$$

2) Let $\alpha \leq 1$. Following similar arguments we get

$$R_N(p) \geq pq_c - \frac{p\alpha^N(\alpha-1)}{4(\alpha^{2N+1}-1)} \qquad (17)$$

Combining Eqs (16) and(17), we get

$$R_N(p) \geq \min(q_c, q_d)p - \frac{\alpha^N(1-\alpha)p}{4(1-\alpha^{2N+1})})$$

To further lower bound $R_N(p)$, we find the maximum value of the second term $f(\alpha, p) := \frac{\alpha^N(1-\alpha)p}{4(1-\alpha^{2N+1})}$. $f(\alpha, p)$ can be simplified as

$$f(\alpha, p) = \frac{\alpha^N(1-\alpha)p}{4(1-\alpha^{2N+1})} = \frac{p/4}{\sum_{k=1}^{N}(\alpha^k + \frac{1}{\alpha^k}) + 1}$$

The function $(\alpha^k + \frac{1}{\alpha^k})$ is convex in α and attains minimum at $\alpha = 1 \quad \forall k \in [1, N]$ Hence $f(\alpha, p)$ attains maximum at $\alpha = 1$ and its value is given by

$$\lim_{\alpha \to 1} f(\alpha, p) = \frac{p}{4(2N+1)}$$

Hence we get $R_N(p) \geq min(q_c, q_d)p - \frac{p_{max}}{4(2N+1)}$

Proof for Theorem 3

To find the value of N_ϵ, note that $\frac{p_{max}}{4(2N+1)} < \epsilon$. Hence we set $N_\epsilon = \frac{p_{max}}{8\epsilon} - \frac{1}{2}$.

C. Dynamic pricing

Poisson Arrivals

In this case , the arrival of drivers and customers are Poisson distributed. Transition probabilities are same as in static case equations, as shown in Fig. 4. The expected reward is defined by equation (5). Putting this together in the value iteration function gives

$$v(s) = \max_{p \in [p_{min}, p_{max}]} \left\{ r(s,p) + \beta \sum_{s'} P(s'/s,p)v(s') \right\}$$

$$
\begin{aligned}
v(s) = \max_{p \in [p_{min}, p_{max}]} \Bigg\{ & \frac{\lambda_d(p)}{\lambda_d(p) + \lambda_c(p)} p \\
& + \beta \left(\frac{\lambda_d(p)}{\lambda_d v + \lambda_c(p)} v(s-1) \right. \\
& \left. + \frac{\lambda_c(p)}{\lambda_d(p) + \lambda_c(p)} v(s+1) \right) \Bigg\} \text{ for } 0 < s < N_c
\end{aligned} \qquad (18)
$$

$$
\begin{aligned}
v(s) = \max_{p \in [p_{min}, p_{max}]} \Bigg\{ & \frac{\lambda_c(p)}{\lambda_d(p) + \lambda_c(p)} p \\
& + \beta \left(\frac{\lambda_d(p)}{\lambda_d v + \lambda_c(p)} v(s-1) \right. \\
& \left. + \frac{\lambda_c(p)}{\lambda_d(p) + \lambda_c(p)} v(s+1) \right) \Bigg\} \text{ for } -N_d < s < 0
\end{aligned}
$$

Acknowledgments

The authors would like to thank support from INSPIRE faculty fellowships from DST, Government of India and SEED grant (16IRCCSG010) from IIT Bombay.

References

[1] G. J. Stigler, "The literature of economics: the case of the kinked oligopoly demand curve," *Economic Inquiry*, vol. 16, no. 2, pp. 185–204, 1978.
[2] E. Maskin and J. Tirole, "A theory of dynamic oligopoly, ii: Price competition, kinked demand curves, and edgeworth cycles," *Econometrica: Journal of the Econometric Society*, pp. 571–599, 1988.
[3] C. W. Efroymson, "A note on kinked demand curves," *The American Economic Review*, vol. 33, no. 1, pp. 98–109, 1943.
[4] M. Sood, S. Moharir, and A. A. Kulkarni, "Pricing in two-sided markets in the presence of free upgrades," in *Proceedings of the tenth IEEE International Conference on Communication Systems & Networks (COMSNETS'18)*, 2018.
[5] S. Banerjee, C. Riquelme, and R. Johari, "Pricing in ride-share platforms: A queueing-theoretic approach," in *Proceedings of the Sixteenth ACM Conference on Economics and Computation (EC'15)*, 2015.
[6] J. Hall, C. Kendrick, and C. Nosko, "The effects of ubers surge pricing: A case study," in *A research collaboration report*, 2016.
[7] J.-C. Rochet and J. Tirole, "Two-sided markets: A progress report," vol. 37, no. 3, pp. 645–667, 2006.
[8] M. Rysman, "The economics of two-sided markets," *Journal of Economic Perspectives*, vol. 23, no. 3, pp. 125–43, 2009.
[9] S. Banerjee, D. Freund, and T. Lykouris, "Pricing and optimization in shared vehicle systems: An approximation framework," in *Available online on Arxiv*, 2017.
[10] M. Hu and Y. Zhou, "Dynamic matching in a two-sided market," in *Available on SSRN*, 2015.
[11] Z. Fang, L. Huang, and A. Wierman, "Prices and subsidies in the sharing economy," in *Proceedings of the 26th International Conference on World Wide Web (WWW'17)*, 2017.
[12] E. G. Weyl, "A price theory of multi-sided platforms," in *American Economic Review*, 2009.
[13] T. Holzmann, K. Sailer, and B. R. Katzy, "Matchmaking as multi-sided market for open innovation," *Technology Analysis and Strategic Management*, vol. 26, no. 6, pp. 601–615, 2014.
[14] M. Chen, W. Shen, P. Tang, and S. Zuo, "An economic approach to vehicle dispatching for ride sharing," in *Available online on Arxiv*, 2018.
[15] B. Kostas, C. Ozan, and S. Daniela, "Spatial pricing in ride-sharing networks," in *Available online on SSRN*, 2016.
[16] M. L. Puterman. John Wiley & Sons, Inc., 1994.