Non-asymptotic near optimal algorithms for two sided matchings

Rahul Vaze
School of Technology and Computer Science
Tata Institute of Fundamental Research

Jayakrishnan Nair
Department of Electrical Engineering
IIT Bombay

Abstract—**A two-sided matching system is considered, where servers are assumed to arrive at a fixed rate, while the arrival rate of customers is modulated via a price-control mechanism. We analyse a loss model, wherein customers who are not served immediately upon arrival get blocked, as well as a queueing model, wherein customers wait in a queue until they receive service. The objective is to maximize the platform profit generated from matching servers and customers, subject to quality of service constraints, such as the expected wait time of servers in the loss system model, and the stability of the customer queue in the queuing model. For the loss system, subject to a certain relaxation, we show that the optimal policy has a *bang-bang* structure. We also derive approximation guarantees for simple pricing policies. For the queueing system, we propose a simple bimodal matching strategy and show that it achieves near optimal profit.**

I. Introduction

Two-sided queues, where customers and servers both arrive to a platform/aggregator and then wait to be matched, have been made fairly popular by ride hailing applications like Uber and Lyft that match passengers with drivers, meal delivery couriers like Grubhub and DoorDash that match diners with delivery couriers, and crowdsourcing platforms like Amazon MTurk, where tasks are matched to workers/volunteers. The objective of the platform is to maximize profit, while improving the market efficiency.

To earn revenue, the platform sets a two sided or one sided price. With two sided pricing, both the customers and the servers are advertised a (possibly different) price, and customers willing to pay the quoted price and servers who are willing to serve for the quoted price enter the system. The platform profit is then the difference between the two prices. In a one sided price model, only the customers are quoted a price, and customers who are willing to pay enter the system. The server arrival rate is assumed to be fixed, and insensitive to price. In this model, the platform keeps a fraction of the revenue (made from customers) to itself and distributes the rest uniformly across all servers. In many practical systems, server payoffs are better modelled as long-term rewards [1], and thus considering one sided pricing is reasonable. In both models, the price is dynamically adjusted to maximize profit, and to efficiently match supply and demand.

With two-sided queues, we consider two well studied models: the loss model, and the queueing model. For both models, we assume one sided pricing, where the platform advertises a state dependent price that determines the rate of arrival of customers, while the rate of arrival of servers is fixed and insensitive to the platform price.

Under the loss model, servers arrive into a queue and wait, while arriving customers are instantaneously matched to the head of the line server, if any. A customer arriving when no servers are present is lost. The objective we consider with the loss model is to maximize a linear combination of the platform's profit and the expected delay experienced by servers, and the goal is to find the optimal dynamic price to maximize this objective function.

With the queuing model, both the servers and the customers wait in their respective queues, and are matched appropriately. In this case, we consider the objective of maximizing the platform's profit subject to the stability of the customer queue. The goal is to design a dynamic pricing and matching strategy to achieve optimal platform profit while ensuring the stability of the customer queue.

Prior Work: With a single sided queue (where server is fixed and only customers arrive), [2] considered a loss-system with finite resource capacity. A customer arriving when all resources are occupied is lost, and the problem is to decide the price of admission given the state of the system at each time to maximize the expected payoff. Similar pricing models have been considered for a queuing system, where customers wait in a queue, and server's service distribution is exponential with a tuneable parameter. Optimal admission control, with and without pricing, and service parameter selection, so as to maximize a linear combination of expected payoff and expected waiting time has been considered in [3]–[8], assuming Poisson arrivals. In particular, [6] exactly characterizes the optimal policy, though not in closed-form, while [8] derives an asymptotically optimal dynamic pricing policy.

Two sided queues have been considered in [1], [9]–[15], where both customers and servers arrive over time, and wait to be matched. In particular, most of these papers are motivated from ride-hailing applications such as Uber, Lyft, or meal delivery services Grubhub, DoorDash, etc. Some of the results [1], [13]–[15] in this area focus on the importance of dynamic pricing over static pricing. Under a suitable fluid scaled limit of this system, [1] showed that static pricing is sufficient to optimize the objective function, and dynamic pricing only helps in improving the robustness of the system.

An extension of this two-sided queue model with multiple

types of servers and customers has also been considered in [10], [16]–[19], where an additional bipartite matching decision has to be made. In [16], limiting results of matching rates between certain customer and server types with FCFS scheduling have been analyzed. Without pricing, the optimal matching to minimize the queueing cost has been analyzed in [18] under a suitably scaled large system limit. With the objective of minimizing the discounted reward obtained by matching customers and servers over a finite horizon, while accounting for the waiting costs, [19] analyzed fixed pricing strategies, while dynamic pricing strategies were considered in [10], [20]. Most of these analyses are presented using a large system scaling regime.

To summarize, in prior work, either exactly optimal but non-closed form policies have been characterized, or structural properties of the optimal policy have been established, or performance analysis of simple strategies has been performed. Importantly, to the best of our knowledge, any analysis of an explicit policy has been via an asymptotic (large system) scaling.

Our Contributions: Compared to prior work, the focus of this work is to consider simple (one-sided) pricing strategies for two-sided queues without considering any large system limits, e.g., fluid limits or asymptotics, with near-optimal performance. Specifically, we model server arrivals as an exogenous random process (independent of the system state and the matching/pricing policy). On the other hand, customer arrivals are modulated by the price p posted by the platform. Formally, we model the customer arrival rate $\mu(p)$ as a non-increasing concave function of the platform price p. We consider two variants of our model: a loss model where customers not served upon arrival are blocked, and a queueing model, where customers wait in a FCFS queue until they are matched with a server. Our contributions for these two models are as follows.

1) Loss System: For the loss system, we consider that the customer and server arrival processes are Poisson, and the objective is to maximize a linear combination of the platform's profit and the expected delay experienced by servers.

When function $\mu(p)$ is linear in p (which is an important case [21] in pricing/profit models), we consider a suitable relaxation of the objective, for which we show that the optimal pricing strategy has a simple bang-bang style threshold structure. This means that the price is set to be the maximum feasible value when the number of servers is less than a threshold, and to the minimum feasible value when the number of servers exceeds the same threshold. This bang-bang structure of the optimal policy implies that computing the optimal policy boils down to a simple one-dimensional search. Moreover, the same structure can also be exploited to speed up online learning of the optimal policy via reinforcement learning (see [22]). While the relaxation we consider is quite similar to models studied in the literature, that a linear price sensitivity function admits such a simple optimal policy has not been observed before, to the best of our knowledge. Additionally, we show that a simple static pricing model can be near-optimal under certain conditions.

2) Queueing System: Compared to the loss system, with the queueing system, we consider general customer and server arrival distributions, and a discrete time setup. The objective function is to maximize the platform's profit subject to the customer queue being stable. Towards that end, we propose an algorithm that charges a static price, and uses a bi-modal matching strategy with a single threshold U, to decide the number of customer-server pairs that should be matched at each time slot. The main idea of the algorithm is to choose a static price thereby making the customer arrival rate constant over time. The threshold U is chosen in such way that the rate of customer departures (number of customer-server matched pairs) in each time slot is close to the revenue-optimal departure rate. Small perturbations in the departure rate, required to maintain the customer queue at a steady level, result in near-optimal performance.

In particular, we show that the proposed strategy has an additive sub-optimality gap of $O\left(\frac{\log U}{U}\right)$. Moreover, the expected delay experienced by customers with the proposed strategy is $O(U)$. Thus, even though we do not explicitly include the expected delay experienced by the customers in the objective function and only enforce the constraint that the customer queue should be stable, a by-product of the proposed algorithm and its analysis is that the expected delay experienced by the customers is also controlled by the threshold U. Thus, given additional QoS constraints on the expected delay experienced by the customers, we can choose the threshold U to achieve a tradeoff between the sub-optimality gap and the expected delay.

Due to space constraints, several proofs have been omitted; these can be found in [23].

II. LOSS MODEL

In this section, we consider the case where customers do not wait, i.e., customers that arrive when there are no servers in the system get lost/blocked.

A. Model and Preliminaries

We model the loss system in continuous time. Servers arrive to the platform as per a Poisson process with rate λ. Upon arrival, servers wait in an infinite buffer queue, until they are matched to a customer. The customer arrival rate $\mu = g(p)$ is a function of the platform's posted price p, where g is a strictly decreasing and concave function of p. Specifically, when the platform posts price p, the time until the next customer arrival is assumed to be exponentially distributed with mean $1/g(p)$. Thus, the function g captures the price sensitivity of the customer base.

We assume that the price p is constrained to lie in $[p_{\min}, p_{\max}]$, where $0 < p_{\min} < p_{\max}$, and that the price defaults to $p_{\max}$ when there are no servers available. Customers that arrive when there are no servers are blocked/lost, i.e., customers do not queue.[1]

---

[1]The case where customers also can get queued will be considered in Section III.

Under the above model, given the memorylessness in server and customer interarrival times, the state of the system is captured by the number of servers in the queue. Denoting the platform's price when there are i servers in the queue by p_i, the state evolves as per the (controlled) birth-death Markov chain depicted in Figure 1. Here, $\mu_i := g(p_i)$ and $\mu_i \in [g(p_{\max}), g(p_{\min})]$.
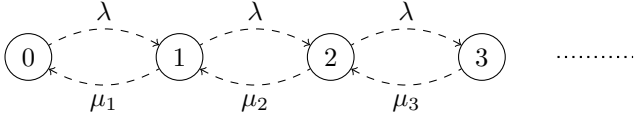

Fig. 1. Birth death chain for server evolution

The goal of the platform is to set prices $(p_i, \ i \in \mathbb{N})$ so as to maximize $\tilde{C} = \lambda E[P] - \tilde{w}E[N]$, where P denotes the stationary price seen by a (matched) customer, N denotes the stationary number of servers in the system, and $\tilde{w}$ is a positive weight. The impicit constraint here is of course that the server queue is stable, i.e., the Markov chain describing the temporal evolution of the number of waiting servers is positive recurrent. The first term in $\tilde{C}$ is the revenue rate for the platform, the second may be interpreted as the (holding) cost associated with servers idling. By Little's law, maximizing $\tilde{C}$ is equivalent to maximizing

$$C = \frac{\tilde{C}}{\lambda} = E[P] - \tilde{w}E[T] = E[P] - wE[N]$$

where T denotes the stationary server sojourn time, $w = \frac{\tilde{w}}{\lambda}$.

The optimal policy can be computed numerically using standard machinery from the theory of Markov decision process (see, for example, [24]). Related problem formulations are also analysed in [2], [6]; these references characterize structural properties of the optimal policy. However, our goal here is to consider a relaxed version of the above objective, which admits a more explicit analysis.

We now describe our relaxed objective. For a given policy $(p_i, \ i \in \mathbb{N})$, let the stationary distribution associated with the server occupancy be denoted by $\pi = (\pi_i, \ i \in \mathbb{N})$. Our objective C may be expressed in terms of π as follows:

$$C = \underbrace{\sum_{i=1}^{\infty} \pi_{i-1} p_i}_{E[P]} - w \underbrace{\sum_{i=1}^{\infty} i\pi_i}_{E[N]}.$$

It is important to note that in the expression for $E[P]$, we have π_{i-1} multiplying p_i because the long run fraction of (server) departures out of state i (i.e., departures that *leave behind* $i-1$ servers in the system) equals the long run fraction of server arrivals that *see* $i-1$ servers in the system, which, by PASTA, equals π_{i-1}. Our relaxed objective is now stated as follows.

$$C_{\mathrm{rel}} = \sum_{i=1}^{\infty} \pi_{i-1} p_{i-1} - w \sum_{i=1}^{\infty} i\pi_i = \sum_{i=0}^{\infty} \pi_i p_i - w \sum_{i=1}^{\infty} i\pi_i \quad (1)$$

The relaxation replaces the term $\sum_{i=1}^{\infty} \pi_{i-1} p_i$ in the objective by $\sum_{i=1}^{\infty} \pi_{i-1} p_{i-1}$. The motivation for this relaxation is of

course to align the distribution used to average price with the stationary distribution.[2] Clearly, C_{rel} and C would be close when the price varies slowly with state. Specifically, for static pricing policies, the two objectives are identical. However, even for the bang-bang type policies we consider later, we find that two objectives are aligned, as we demonstrate as part of our numerical experiments in Section II-E.

Most of our results will be derived for the case where the price sensitivity function g is linear:

Assumption 1. $g(p)$ *depends linearly on* p, *i.e.,* $g(p) = \beta - \alpha p$, *where* $\beta, \alpha > 0$. *Moreover,* $\mu_{\max} := \beta - \alpha p_{\min} > \lambda$, *and* $\mu_{\min} := \beta - \alpha p_{\max} > 0$.

We also consider a more general class of concave price sensitivity functions, and prove bounds on the sub-optimality of simple static pricing policies for this class.

Assumption 2. $g(p) = (\beta - \alpha p)^\theta$, *where* $\theta \in (0,1]$, *and* $\beta, \alpha > 0$. *Moreover,* $\mu_{\max} := (\beta - \alpha p_{\min})^\theta > \lambda$, *and* $\mu_{\min} := (\beta - \alpha p_{\max})^\theta > 0$.

Under both assumptions, note that $p_{\max} < \frac{\beta}{\alpha}$. Thus, $\frac{\beta}{\alpha}$ is a trivial upper bound on the objective value achievable. More refined bounds will be derived in Section II-D.

B. Linear price sensitivity & relaxed objective: Optimal policy

Throughout this section, we make Assumption 1, and consider the relaxed objective C_{rel} (see (1)). Recall that the number of servers evolves as per the birth death Markov chain shown in Figure 1. Here, p_i, or equivalently $\mu_i = g(p_i)$, can be interpreted as the 'action' taken in state i. Recall also that $p_0 = p_{\max}$; let $\mu_0 := g(p_0) = \mu_{\min}$.

We begin by rewriting C_{rel} as follows.

Lemma 1. *Under Assumption 1,* $C_{rel} = \frac{1}{\alpha}\left(\beta - \pi_0\mu_0 - \lambda\right) - w\sum_{i=1}^{\infty} i\pi_i$.

Lemma 1 states that under linear price sensivity, the first term in the relaxed objective decreases linearly in π_0. In other words, maximizing the first term boils down to minimizing π_0. We now exploit this property to show that the optimal policy is of bang-bang type.

Theorem 3. *Under Assumption 1 (wherein the price sensitivity function g is linear), there exists a policy that optimizes C_{rel} of the form*

$$p_i = \begin{cases} p_{\max} & \text{for } i < \ell^* \\ p^* & \text{for } i = \ell^* \\ p_{\min} & \text{for } i > \ell^* \end{cases},$$

where $\ell^* \in \mathbb{N} \cup \{\infty\}$, $p^* \in [p_{\min}, p_{\max}]$.

The bang-bang policy stated in Theorem 3 has the following interpretation: When there are fewer than ℓ^* servers in the

---

[2]This 'mis-alignment' arises in C because we control the *left* transition rates in this model. In an alternative model wherein the control is on the *rightward* transition rates (this would arise if customers were to queue, and servers do not; think of an airport taxi lot), the 'mis-alignment' would not occur, and we would not need to relax the objective in this manner.

system, the platform sets the maximum price, thereby limiting customer arrivals as far as possible so as to provide the maximum upward drift to the server queue. On the other hand, when the number of servers exceeds ℓ^*, the platform sets the minimum price, thereby maximizing the rate of customer arrivals and providing the maximum downward drift to the server queue. Finally, when there are exactly ℓ^* servers, the platform sets an intermediate price p^*. Thus, the platform in effect seeks to maintain the number of servers at around ℓ^* to the extent that its control over customer arrivals permits. As will be apparent from the proof of Theorem 3, the aforementioned policy effectively minimizes the probability that the server queue becomes empty, subject to an upper bound on the expected stationary number of servers in the system.

Remark 4. *It is important to note that the bang-bang structure of the optimal policy as stated in Theorem 3 is significantly stronger than the monotonicity properties that are typically established in structured MDPs (see, for example, [25]). Moreover, note that Theorem 3 is proved using the representation of the relaxed objective in Lemma 1, which in turn relies heavily on the linearly of the price sensitivity function g.*

Remark 5. *While the optimal policy as stated in Theorem 3 is parameterized by (ℓ^*, p^*), it can also be parameterized via a single parameter $x^* \in \mathbb{R}_+$. The two parameterizations are related as follows: $\ell^* = \lceil x^* \rceil$, and $p^* = p_{\max} - (\lceil x^* \rceil - x^*)(p_{\max} - p_{\min})$. This one-dimensional parameterization simplifies the task of computing the optimal policy as a function of the system parameters, and is amenable to efficient online learning via standard stochastic approximation techniques (see [22]).*

Proof of Theorem 3. It suffices to show that an optimal policy of the specified form exists for any *constrained* optimization of the following form, parameterized by $\mathcal{C} > 0$.

$$\max \ \frac{1}{\alpha}\left(\beta - \pi_0 \mu_0 - \lambda\right) \qquad s.t. \ \sum_{i=1}^{\infty} i\pi_i \leq \mathcal{C}$$

Equivalently, it suffices to show that an optimal policy of the specified form exists for any *constrained* optimization of the following form:

$$\min \ \pi_0 \qquad s.t. \ \sum_{i=1}^{\infty} i\pi_i \leq \mathcal{C} \qquad (2)$$

That this problem, which is well motivated in its own right, has an optimal solution of the bang-bang form specified, is proved as Lemma 2. □

The optimization (2) is a natural and well-motivated problem in the context of controlled birth death Markov chains over $\mathbb{Z}_+$, where the goal is to minimize the stationary probability of state 0, subject to an upper bound on the expectation of the steady state distribution. Since this problem is interesting in its own right, we study it (independently of its application to our dynamic pricing model) in the following section.

C. A bang-bang lemma for controlled Markov chains

Consider a birth death Markov chain over state space $\mathbb{Z}_+$. The transition rate from state i to $i+1$ is denoted by λ_i for $i \geq 0$ and the transition rate from state i to $i-1$ is denoted by μ_i for $i \geq 1$. For $i \geq 1$ $\rho_i := \frac{\lambda_{i-1}}{\mu_i}$.

Assuming the chain is positive recurrent, the stationary distribution is given by $\pi_i = \frac{h_i}{\sum_{j=0}^{\infty} h_j}$, where $h_0 = 1$, $h_i = \prod_{j=1}^{i} \rho_j$ for $i \geq 1$.

With ρ_i constrained to lie in $[\underline{\rho}, \bar{\rho}]$, where $0 < \underline{\rho} < \bar{\rho}$, our goal is to minimize π_0 subject to a moment condition on the stationary distribution. Formally, this is posed as:

$$\begin{aligned} \min. \quad & \pi_0 \\ s.t. \quad & \sum_{i=0}^{\infty} i\pi_i \leq C \\ & \rho_i \in [\underline{\rho}, \bar{\rho}] \quad (i \geq 1) \end{aligned} \qquad (3)$$

An implicit constraint here is of course that the chain is positive recurrent. Clearly, $\underline{\rho} < 1$ is necessary and sufficient for the feasibility of positive recurrence of the chain. Note also that if $\underline{\rho} < 1$ and $\frac{\underline{\rho}}{1-\underline{\rho}} > C$, then the optimization (3) is infeasible (since the expected steady state value of the chain is easily seen to be at least $\frac{\underline{\rho}}{1-\underline{\rho}}$). Thus, the optimization (3) is well posed if $\underline{\rho} < 1$ and $\frac{\underline{\rho}}{1-\underline{\rho}} \leq C$. Under these conditions, the following lemma shows that the optimal control is of bang-bang type.

Lemma 2. *Assuming $\underline{\rho} < 1$ and $\frac{\underline{\rho}}{1-\underline{\rho}} \leq C$, the optimal solution of (3) is of the form*

$$\rho_i = \begin{cases} \bar{\rho} & \text{for } i < \ell^* \\ r^* & \text{for } i = \ell^* \\ \underline{\rho} & \text{for } i > \ell^* \end{cases},$$

where $\ell^* \in \mathbb{N} \cup \{\infty\}$, $r^* \in [\underline{\rho}, \bar{\rho}]$.

This lemma is proved by showing that any feasible policy that does not have the above bang bang structure can be improved upon via a perturbation towards this structure (details can be found in [23]). The proof arguments are similar to those in [26], where a bang-bang style policy is shown to be optimal in a different context.

D. Single (static) price policies

Next, we consider the simplest pricing policy: a constant, state-independent price, under the relaxed objective C_{rel}.[3] While static pricing might seem naive, we show that under certain conditions, if the static price is chosen carefully, the suboptimality relative to the (unknown) optimal policy can be bounded. This is somewhat analogous to what happens with the classical server speed scaling problem, where it is known that a suitably chosen static speed choice is constant competitive under a stochastic workload model [27], [28].

We begin by deriving some universal upper bounds on the objective value under *any* policy (not necessarily static).

---

[3]Note that under static pricing, C and C_{rel} are equal. However, we since one of our universal upper bounds (Lemma 3) is only proved for the relaxed objective, we persist with the use of C_{rel} throughout this section.

1) Universal upper bounds:

Lemma 3. *Under Assumption 2, for any policy, we have $C_{rel} \leq g^{-1}(\lambda)$.*

Lemma 4 (Light traffic bound)**.** *Under Assumption 2, for any policy, $C_{rel} \leq \max_{p \in [p_{\min}, p_{\max}]} \left(p - \frac{\tilde{w}}{g(p)} \right)$.*

Next, we apply these upper bounds to study the competitiveness of static pricing for the case of linear price sensitivity.

2) Competitiveness under linear model: We begin by specializing the above universal bounds to Assumption 1, i.e., $g(p) = \beta - \alpha p$, where $\beta, \alpha > 0$. Combining the two bounds above together, under Assumption 1, for any policy,

$$C_{\text{rel}} \leq \frac{\beta}{\alpha} - \max \left(\frac{\lambda}{\alpha}, \frac{2\sqrt{\tilde{w}\alpha}}{\alpha} \right). \qquad (4)$$

Clearly, the condition $\beta > 2\sqrt{\tilde{w}\alpha}$ is a necessary condition for positive objective value under any policy. Now the optimal static price is given by $p^* = \arg\max_{p \in [p_{\min}, p_{\max}]} \left(p - \frac{\tilde{w}}{\beta - \alpha p - \lambda} \right)$. If $\mu_{\max} = \beta - \alpha p_{\min} \geq \lambda + \sqrt{\alpha\tilde{w}}$, then $p^* = \frac{\beta - \sqrt{\alpha\tilde{w}} - \lambda}{\alpha}$. The static policy that always chooses the price p^* would then have payoff

$$C_{\text{rel}}(p^*) = \frac{\beta - 2\sqrt{\alpha\tilde{w}} - \lambda}{\alpha}. \qquad (5)$$

Comparing (4) and (5), it follows that so long as $\mu_{\max} \geq \lambda + \sqrt{\alpha\tilde{w}}$, i.e., it is feasible to maintain a $\sqrt{\alpha\tilde{w}}$ slack in the customer arrival rate relative to the server arrival rate, the reduction in payoff from $\frac{\beta}{\alpha}$ under the optimal static pricing policy is at most twice that under any policy. Reframing the objective as a minimization of the payoff reduction from its (unattainable) upper bound $\frac{\beta}{\alpha}$, this implies a competitive ratio of at most two.

3) Competitiveness under non-linear model: We now consider the more general concave price sensitivity model specified by Assumption 2. Specializing our two upper bounds to this particular choice of $g(\cdot)$, we get that under any policy,

$$C_{\text{rel}} \leq \frac{\beta}{\alpha} - \max \left(\frac{\lambda^{1/\theta}}{\alpha}, \frac{B}{\alpha} \right), \qquad (6)$$

where $B := (\tilde{w}\alpha\theta)^{1/(\theta+1)}(1 + 1/\theta)$.

Unlike in the linear case however, this non-linear model for $g(\cdot)$ does not admit a closed form characterization of the optimal static price. We thus consider two separate cases based on which term contributes to the max in (6). For each of these cases, a different reasonable choice of static price is considered.

Case 1 (heavy traffic): $\lambda^{1/\theta} \geq B$

Consider the static policy that sets the price p such that $\mu = (\beta - \alpha p)^\theta = \gamma\lambda$, where $\gamma > 1$. Clearly, such a γ exists, given that $\mu_{\max} > \lambda$. The payoff under this policy is

$$C_{\text{rel}} = \frac{\beta}{\alpha} - \frac{(\gamma\lambda)^{1/\theta}}{\alpha} - \frac{w}{(\gamma - 1)\lambda}.$$
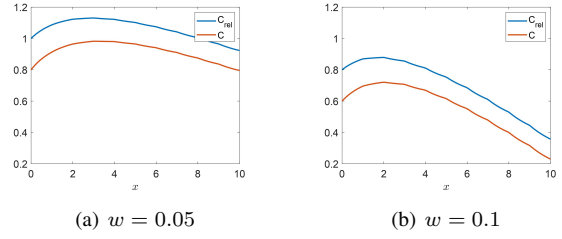

(a) $w = 0.05$ (b) $w = 0.1$

Fig. 2. Comparison of the original objective C and the relaxed objective C_{rel} for bang-bang policies. The policy parameters are related to the x-axis label as follows: $\ell = \lceil x \rceil$, $p = p_{\max} - (\lceil x \rceil - x)(p_{\max} - p_{\min})$. Chosen system parameters are as follows: $\lambda = 2$, $p_{\min} = 1$, $p_{\max} = 2$, $\beta = 3.5$, $\alpha = 1$.

For large λ, the last term above would be negligible, meaning the payoff reduction from $\frac{\beta}{\alpha}$ would be (approximately) at most a factor of $\gamma^{1/\theta}$ of that under any policy.

Case 2 (light traffic): $\lambda^{1/\theta} < B$

In this case, consider the static policy that sets the price such that $\mu = (\beta - \alpha p)^\theta = \gamma B^\theta$, where $\gamma > 1$. Such a choice is of course feasible only when $\mu_{\max} > B^\theta$. If so, the cost under this policy is bounded as:

$$C_{\text{rel}} \geq \frac{\beta}{\alpha} - \frac{\gamma^{1/\theta}B}{\alpha} - \frac{w}{(\gamma - 1)B^\theta}.$$

For large B, the last term above would be negligible, meaning the payoff reduction from $\frac{\beta}{\alpha}$ would be (approximately) at most a factor of $\gamma^{1/\theta}$ of that under any policy, as before.

E. Numerical experiments

Since we have used the relaxed objective C_{rel} in the preceding sections, we now present some numerical results illustrating the connection between C_{rel} and C. Specifically, we compare both objectives over the (one-dimensional) space of bang-bang policies of the kind we proved as optimal for C_{rel} for linear price sensitivity.

In Figure 2, we plot C_{rel} and C as a function of a single parameter x that specifies the bang bang policy, as described in Section II-B, for different values of w. We note that increasing the weight w on the holding cost decreases the objective values, and decreases the optimal choice of x, as expected. However, what is interesting to note is that the optimal choice of x under the relaxed objective matches almost perfectly with the optimal choice under the original objective. This suggests that the optimal bang-bang policy under the relaxed objective is also a near optimal choice (within the class of bang-bang policies) for our original objective.

III. QUEUEING MODEL

Consider a discrete time system, where the arrival rate for servers is insensitive to price and is equal to $\lambda(t) = \lambda$ for each time t[4]. The customers respond to price set by the platform, and the arrival rate of customers in any time slot t is $\mu(p(t))$ if the price chosen by the platform is $p(t)$ at time t. We assume the natural model, where $\mu(p)$ is a non-increasing

---

[4]With abuse of notation, we are calling λ as the arrival rate with a discrete time system similar to the continuous time loss model of Section II.

continuous function of p. In addition, we assume that $\mu(p)$ is a concave function, and $p\mu(p)$ is a unimodal function of p, following prior work [11]. Recall that for notational convenience in Section II, we assumed that the customer arrival rate is $\mu = g(p)$, while in this section, we use simply $\mu(p)$.

Under this model, let $A(t)$ and $B(t)$ be the number of servers and customers that arrive in time slot t, following i.i.d. processes, with rate λ and $\mu(p(t))$, respectively. In particular, $\lambda = \mathbb{E}\{A(t)\}$ and $\mu(p(t)) = \mathbb{E}\{B(t)\}$. Let $M(t)$ be the set of customers that depart (number of customer-server pairs that are matched) at time t. Since there are two queues corresponding to servers and customers, making both queues stable (that allows the existence of steady state distribution) simultaneously is not possible without some exogenous constraint (unless we have a separate price lever for each queue, as is done in [10]). So to keep the server queue stable, we consider an upper limit $\bar{S}$ on the number of outstanding servers, while ensuring the stability of customer queue will be part of the considered problem. Thus, a server arriving when the server queue size is $\bar{S}$ is not admitted.

The number of servers $S(t)$, and customers $C(t)$, in the system at time t, evolve as follows.

$$S(t+1) = \min\{(S(t) + A(t) - |M(t)|)^+, \bar{S}\}, \quad (7)$$
$$C(t+1) = (C(t) + B(t) - |M(t)|)^+, \quad (8)$$

where $|M(t)|$ is the number of customer-server pairs that are matched by the platform in time slot t, and $(x)^+ = \max\{0, x\}$. The evolution of $S(t)$ and $C(t)$ is hence coupled via $M(t)$.

Remark 6. *Compared to Section II, where we considered a loss model, in this section, customers wait in the queue, and depart only when they are matched to any server. Moreover, we are considering a more general system than Section II, that evolves in discrete time, and the customer and server arrival processes are not restricted to follow a Poisson distribution.*

Customer i arriving at time t sees or commits to price $p_i = p(t)$, and let the platform make profit of p_i when customer i is matched (departs) to some server at time $t' \geq t$. Thus, the profit made by the platform at time t is determined by the set of $M(t)$ customers that depart at time t, and the price they saw when they arrived $p_i, i \in M(t)$. Thus, given the price $p(t)$ chosen by the platform, its profit (that is a function of price $p(t)$) is given by

$$\mathbf{V} = \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} V\left(\sum_{i \in M(t)} p_i\right), \quad (9)$$

where V is a non-decreasing concave function.

Note that in defining the platform's profit (9), we have not directly accounted for the payment to the servers, however, that is implicitly captured by assuming that the platform keeps a constant factor of the profit to itself and distributes the rest uniformly across all servers. Thus, maximizing profit, is equivalant to maximizing the payment to the servers. Servers not being incentivised per-customer matching is well justified

following [1], which shows that servers payoffs are better modelled as long-term rewards.

The optimization problem that we consider is as follows.

$$\max_{p(t),M(t)} \quad \mathbf{V} \quad \text{s.t.} \quad \underbrace{\lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\{C(t)\} < \infty}_{\text{stability of customer queue}}. \quad (10)$$

The stability condition in (10) takes care of the fact that the delay seen by arriving customers is bounded.

Remark 7. *An alternate formulation to (10) is to maximize* $\mathbf{V}$ *subject to a constraint on the expected delay seen by the customers, or maximize a linear combination of $\mathbf{V}$ and expected delay seen by the customers. Both these alternatives are, however, more challenging to solve in the setting considered, where we are not considering any scaling limit regime, unlike [1] and similar papers. In what will follow, the algorithm we propose to solve (10), will have a parameter that will tradeoff the sub-optimality gap (Theorem 10) and the expected delay (Lemma 8) seen by the customers. Thus, given a constraint on the expected delay seen by the customers, we can tune the parameter and bound the sub-optimality gap.*

Let the **optimal price** be $p^\star(t)$ and **optimal matching decision** be $M^\star(t)$ to maximize (10) under the stability constraint, and let the optimal profit be $\mathbf{V}_{\mathsf{OPT}}$. Next, we upper bound $\mathbf{V}_{\mathsf{OPT}}$. Towards that end, we define a **critical** quantity $p^\star$, as follows.

$$p^\star = \arg \max_{p, \lambda \geq \mu(p)} p \cdot \mu(p). \quad (11)$$

Note that since $\mu(p)$ is assumed to be a non-increasing continuous concave function of p, it follows that $p\mu(p)$ is a concave function. Using this fact, we get the following upper bound on $\mathbf{V}_{\mathsf{OPT}}$.

Lemma 5. $\mathbf{V}_{\mathsf{OPT}} \leq V(p^\star \mu(p^\star))$.

Lemma 5 essentially says that the largest profit is possible if the price is set as constant $p^\star$, and the number of matched customer-server pairs $|M(t)|$ is a constant equal to the expected customer arrival rate $\mu(p^\star)$ at price $p^\star$. The proof follows from a straightforward application of Jensen's inequality, given that V is a concave function.

Next, we propose an algorithm for setting the price $p(t)$, and choosing the number of matched customers $|M(t)|$ in each time slot t, and lower bound its profit.

Algorithm $\mathcal{A}$: Following the definitions of S(t) (7) and C(t) (8), let $N(t) = \min\{S(t), C(t)\}$. Solving (11), either we have $\lambda > \mu(p^\star)$ or $\lambda = \mu(p^\star)$.

If $\lambda > \mu(p^\star)$, the algorithm chooses constant price $p(t) = p^\star$, $\forall$ t, defined in (11), and the number of customer-server pairs **matched** in slot t using FIFO schedule are

$$|M_\mathcal{A}(t)| = \begin{cases} 0 & \text{if } N(t) < \mu(p^\star) - \delta, \\ \mu(p^\star) - \delta & \text{if } \mu(p^\star) - \delta \leq N(t) \leq U/2, \\ \mu(p^\star) + \delta & \text{if } N(t) > U/2, \end{cases} \quad (12)$$

where $U < \bar{S}$ is some threshold, and $\delta > 0$ that will be chosen later. Threshold U will control both the profit made by the platform as well as the expected waiting time of any customer.

If $\lambda = \mu(p^\star)$, then $p(t) = p^\star + \epsilon$ (to ensure that the customer arrival rate is lower than the server arrival rate), while choice of $|M_\mathcal{A}(t)|$ remains unchanged as in (12).

For the rest of this section, we consider the case when $\lambda > \mu(p^\star)$. All results will go through even when $\lambda = \mu(p^\star)$ with an additional $O(\epsilon)$ penalty.

Remark 8. *Note that we are not enforcing the integrality constraint on $|M_\mathcal{A}(t)|$, similar to prior works [29], [30] on dynamic decision problems with server-customer queues, where the consideration is that the number of servers/customers is large enough at an aggregate scale, and $|M_\mathcal{A}(t)|$ can be thought of as the fraction of customers served.*

With algorithm $\mathcal{A}$, for both cases, $\lambda > \mu(p^\star)$ or $\lambda = \mu(p^\star)$, the arrival rate of servers is more than the arrival rate of customers with the algorithm, ensuring stability of the customer queue, satisfying the constraint in (10). Thus, we only need to derive a lower bound on the profit of algorithm $\mathcal{A}$, for which we need the following definition. Let $\sigma_C^2 = \lim_{T \to \infty} \frac{1}{T} \mathrm{var}\left(\sum_{t=1}^{T} B(t)\right)$ be the variance of process $B(t)$, number of arrivals of customers at time t.

Remark 9. *With the algorithm $\mathcal{A}$ that charges price $p(t) = p^\star$, the process $B(t)$ (the number of customer arrivals with mean $\mu(p^\star)$) is an i.i.d. process. Therefore, σ_C^2 is well-defined.*

The main result of this section is the following.

Theorem 10. *Choosing $\delta = \alpha\left(\frac{\log U}{U}\right)$, where $\alpha = \beta \sigma_C^2$ and $\beta \geq 2$, for algorithm $\mathcal{A}$,*

$$\mathbf{V}_\mathcal{A} \geq \mathbf{V}_{\mathsf{OPT}} - \mathcal{O}\left(\left(\frac{\log U}{U}\right)^2\right).$$

Thus, algorithm $\mathcal{A}$ is near-optimal and the sub-optimality gap is governed by the choice of threshold U.

Proof Sketch: Let N be distributed as as per the steady state distribution of $N(t)$ with algorithm $\mathcal{A}$. By definition, algorithm $\mathcal{A}$ can achieve profit close to $V(p^\star \mu(p^\star))$ (where $\mathbf{V}_{\mathsf{OPT}} \leq V(p^\star \mu(p^\star))$) as long as $N > \mu(p^\star) - \delta$. So the main result is to show that for $\mathcal{A}$, $P_{\mathrm{outage}} = \mathbb{P}(N < \mu(p^\star) - \delta) = \mathcal{O}\left(U^{-\beta}\right)$ for $\delta = \alpha\left(\frac{\log U}{U}\right)$, where $\alpha = \beta \sigma_C^2$ and $\beta \geq 2$, which we prove in Lemma 6. $\square$

Lemma 6. *For algorithm $\mathcal{A}$ (12) with price $p(t) = p^\star$, $\forall\, t$,*

$$P_{\mathrm{outage}} = \mathbb{P}(N < \mu(p^\star) - \delta) = \mathcal{O}\left(U^{-\beta}\right),$$

by choosing $\delta = \alpha\left(\frac{\log U}{U}\right)$, where $\alpha = \beta \sigma_C^2$ and $\beta \geq 2$.

Similar to the Lemma 6, we can get an upper bound on the $P(N \geq x)$ for $x \geq U$ as follows.

Lemma 7. *For algorithm $\mathcal{A}$ (12) with price $p(t) = p^\star$, $\forall\, t$,*

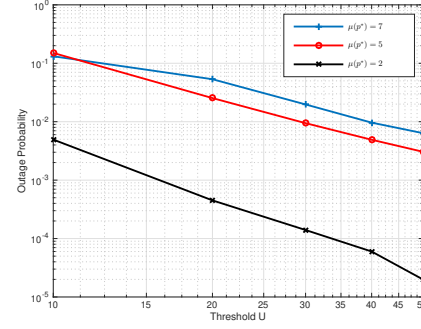$$\mathbb{P}(N \geq x) = \mathcal{O}\left(U^{-\beta}\right)$$

Fig. 3. Outage probability with Poisson driver and customer arrivals with different values of $\mu(p^\star)$ and $\Delta = .1$, respectively, as a function of U.
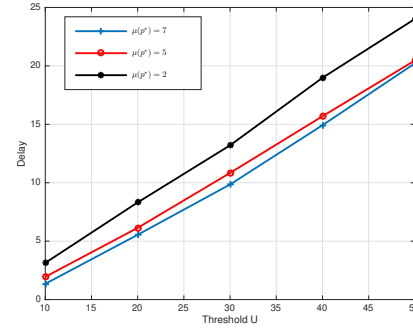

Fig. 4. Expected delay experienced by a customer with Poisson driver and customer arrivals with different value of $\mu(p^\star)$ and $\Delta = .1$, respectively, as a function of U.

by choosing $\delta = \alpha\left(\frac{\log U}{U}\right)$, where $\alpha = \beta \sigma_C^2$ and $\beta \geq 2$, and $x > U$.

Proof of Lemma 7 is identical to that of Lemma 6 and is omitted. An important consequence of Lemma 7 is a bound on the expected customer queue length, which using Little's law gives a bound on the average delay seen by the customers.

Corollary 11. $\mathbb{E}\{N\} = \mathcal{O}(U)$ *by choosing $\beta \geq 3$.*

Lemma 7 helps us in bounding the expected waiting time $\mathbb{E}\{W\}$ seen by any customer using Little's Law. Recall that with algorithm $\mathcal{A}$, the customer arrival rate is constant $\mu(p^\star)$, and the system is stable. Thus, the expected customer departure rate is also $\mu(p^\star)$. Thus, using Little's Law,

$$\mathbb{E}\{W\} = \mathbb{E}\{N\}/\mu(p^\star) = \mathcal{O}\left(\frac{U}{\mu(p^\star)}\right). \tag{13}$$

Thus, we have proved the following lemma.

Lemma 8. *With algorithm $\mathcal{A}$, the expected waiting time $\mathbb{E}\{W\}$ seen by any customer is $\mathbb{E}\{W\} = \mathcal{O}\left(\frac{U}{\mu(p^\star)}\right)$.*

Discussion: The upshot of Theorem 10 is that near optimal profit can be obtained with static pricing, and advantage of dynamic pricing is arbitrarily small. Similar conclusions have been derived in earlier papers e.g., [1], with an objective

function that is a sum of the platform profit and the expected delay seen by the servers, however, these results were shown in limiting regimes, such as scaling the number of customers and servers, and considering a fluid limit. Moreover, Lemma 8 shows that the static pricing policy that achieves near optimal profit has bounded expected delay for customers, as a function of the parameter U that also controls the sub-optimality gap. Thus, threshold level U provides a tradeoff between the sub-optimality gap from Theorem 10 and the expected delay seen by a customer (13), and can be appropriately chosen given QoS requirements.

In Figs. 3 and 4, we illustrate the tradeoff between the sub-optimality gap and the expected customer delay as a function of U, when both the customer and the server arrival distributions are Poisson. Recall that the sub-optimality gap is essentially controlled by the outage probability. Thus, we plot the outage probability and expected customer queue length (that controls the expected delay), as a function of U for different values of $\mu(p^\star)$ and $\Delta = \lambda - \mu(p^\star) = 0.1$. This way we avoid making specific choices of V and μ, while still capturing the quantities of interest. We observe that the simulated performance remains unchanged for higher values of Δ, since the minimum of the two rates λ, $\mu(p^\star)$, controls the performance. Hence to avoid cluttered plots, we only illustrate the $\Delta = 0.1$ case. For simulations, we use $\beta = 2$, $\sigma_C^2 = 2$ that completely defines algorithm $\mathcal{A}$'s choice of δ. As promised by theory, the outage probability falls off as power-law with respect to U, while the expected delay is close to $U/2$.

IV. Conclusions

In this paper, we have considered simple pricing strategies for two sided queue matching problems, that have typically been either analysed in large system limits, or for which optimal but non-closed form policies are known. We considered both the loss and the queueing systems, which are relevant for practical applications. For the loss system, we showed an important structural result that the optimal policy is of the bang bang type when the customer arrival rate is a linear function of the price. For the queueing model, we propose a simple static pricing strategy, and a bi-modal matching decision that is shown to be near-optimal, together with a bound on the expected delay seen by the customers.

References

[1] S. Banerjee, R. Johari, and C. Riquelme, "Pricing in ride-sharing platforms: A queueing-theoretic approach," in *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, 2015, pp. 639–639.

[2] I. C. Paschalidis and J. N. Tsitsiklis, "Congestion-dependent pricing of network services," *IEEE/ACM transactions on networking*, vol. 8, no. 2, pp. 171–184, 2000.

[3] D. W. Low, "Optimal pricing for an unbounded queue," *IBM Journal of research and Development*, vol. 18, no. 4, pp. 290–302, 1974.

[4] S. Stidham Jr and R. R. Weber, "Monotonic and insensitive optimal policies for control of queues with undiscounted costs," *Operations research*, vol. 37, no. 4, pp. 611–625, 1989.

[5] J. M. George and J. M. Harrison, "Dynamic control of a queue with adjustable service rate," *Operations research*, vol. 49, no. 5, pp. 720–731, 2001.

[6] B. Ata and S. Shneorson, "Dynamic control of an M/M/1 service system with adjustable arrival and service rates," *Management Science*, vol. 52, no. 11, pp. 1778–1791, 2006.

[7] K. M. Adusumilli and J. J. Hasenbein, "Dynamic admission and service rate control of a queue," *Queueing Systems*, vol. 66, no. 2, pp. 131–154, 2010.

[8] J. Kim and R. S. Randhawa, "The value of dynamic pricing in large queueing systems," *Operations Research*, vol. 66, no. 2, pp. 409–425, 2018.

[9] L. M. Nguyen and A. L. Stolyar, "A queueing system with on-demand servers: local stability of fluid limits," *Queueing Systems*, vol. 89, no. 3, pp. 243–268, 2018.

[10] S. Mahavir Varma, P. Bumpensanti, S. Theja Maguluri, and H. Wang, "Dynamic pricing and matching for two-sided queues," *ACM SIGMETRICS Performance Evaluation Review*, vol. 48, no. 1, pp. 105–106, 2020.

[11] M. Sood, S. Moharir, and A. A. Kulkarni, "Pricing in two-sided markets in the presence of free upgrades," in *COMSNETS*, 2018.

[12] Y. Kanoria and P. Qian, "Near optimal control of a ride-hailing platform via mirror backpressure," *arXiv preprint arXiv:1903.02764*, 2019.

[13] C. Yan, C. Zhu, N. Korolko, and D. Woodard, "Dynamic pricing and matching in ride-hailing platforms," *Naval Research Logistics (NRL)*, vol. 67, no. 8, pp. 705–724, 2020.

[14] O. Besbes, F. Castro, and I. Lobel, "Surge pricing and its spatial supply response," *Management Science*, 2020.

[15] B. Hu, M. Hu, and H. Zhu, "Surge pricing and two-sided temporal responses in ride hailing," *Available at SSRN 3278023*, 2020.

[16] R. Caldentey, E. H. Kaplan, and G. Weiss, "FCFS infinite bipartite matching of servers and customers," *Advances in Applied Probability*, vol. 41, no. 3, pp. 695–730, 2009.

[17] I. Adan and G. Weiss, "Exact fcfs matching rates for two infinite multitype sequences," *Operations research*, vol. 60, no. 2, pp. 475–489, 2012.

[18] I. Gurvich and A. Ward, "On the dynamic control of matching queues," *Stochastic Systems*, vol. 4, no. 2, pp. 479–523, 2015.

[19] M. Hu and Y. Zhou, "Dynamic type matching," *Rotman School of Management Working Paper*, no. 2592622, 2020.

[20] Y. Chen and M. Hu, "Pricing and matching with forward-looking buyers and sellers," *Manufacturing & Service Operations Management*, vol. 22, no. 4, pp. 717–734, 2020.

[21] M. S. Lobo and S. Boyd, "Pricing and learning with uncertain demand," in *INFORMS Revenue Management Conference*, 2003.

[22] A. Roy, V. Borkar, A. Karandikar, and P. Chaporkar, "Online reinforcement learning of optimal threshold policies for markov decision processes," *arXiv preprint arXiv:1912.10325*, 2019.

[23] R. Vaze and J. Nair, "Non-asymptotic near optimal algorithms for two sided matchings," 2022. [Online]. Available: https://arxiv.org/abs/2207.11656

[24] M. L. Puterman, *Markov decision processes: Discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[25] R. F. Serfozo, "Monotone optimal policies for markov decision processes," in *Stochastic Systems: Modeling, Identification and Optimization, II*, 1976, pp. 202–215.

[26] K. Chaudhary, V. Kavitha, and J. Nair, "Dynamic scheduling in a partially fluid, partially lossy queueing system," 2019. [Online]. Available: https://arxiv.org/abs/1904.06480

[27] A. Wierman, L. L. Andrew, and A. Tang, "Power-aware speed scaling in processor sharing systems: Optimality and robustness," *Performance Evaluation*, vol. 69, no. 12, pp. 601–622, 2012.

[28] R. Vaze and J. Nair, "Multiple server srpt with speed scaling is competitive," *IEEE/ACM Transactions on Networking*, vol. 28, no. 4, pp. 1739–1751, 2020.

[29] M. Lin, A. Wierman, L. L. Andrew, and E. Thereska, "Dynamic right-sizing for power-proportional data centers," *IEEE/ACM Transactions on Networking*, vol. 21, no. 5, pp. 1378–1391, 2012.

[30] B. Berg, R. Vesilo, and M. Harchol-Balter, "heSRPT: Optimal parallel scheduling of jobs with known sizes," *arXiv preprint arXiv:1903.09346*, 2019.