Learning-based Optimal Admission Control in a Single Server Queuing System

Yili Zhang
Dept. of Mathematics
University of Michigan
Ann Arbor, MI 48109, USA
zhyili@umich.edu

Asaf Cohen
Dept. of Mathematics
University of Michigan
Ann Arbor, MI 48109, USA
asafc@umich.edu

Vijay G. Subramanian
EECS Department
University of Michigan
Ann Arbor, MI 48109, USA
vgsubram@umich.edu

2022 58th Annual Allerton Conference on Communication, Control, and Computing (Allerton) | 979-8-3503-9998-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/ALLERTON49937.2022.9929406

I. Extended Abstract

We consider admission control for a first-in first-out single class single server queueing model with Poisson arrivals and exponential service times. Specifically, there is a dispatcher that decides on admitting arrivals with the aim to maximize total net profit - each admitted arrival yields a positive reward R (obtained after customer finishes service) that needs to be balanced by a holding cost for the (homogeneous) customers in the queue. Whereas the capacity of this queue is infinite, the dispatcher may decide to reject any customers joining the queue with the profit objective in mind. When the arrival and service rates are known, this model was studied by Naor [1], but in our investigation the dispatcher is assumed to know the arrival rate but not the service rate.

a) Model and the Learning problem: Naor [1] studied the social-welfare maximization problems for the following model. Homogeneous customers arrive at a single server queue according to a Poisson process with rate $0 < \lambda < \infty$. When a customer arrives, the dispatcher decides whether to admit this customer to the queue or not. A customer that is not admitted (i.e., rejected) leaves and does not return. An admitted customer remains in the queue until being served. Upon service completion, the dispatcher receives a reward $R > 0$. Once the service is completed, the customer leaves the queue. The dispatcher suffers from a waiting/holding cost at the rate of $C > 0$ per time unit for each customer in the queue until service completion. The service requirements for the customers are *i.i.d.* EXP(μ) (i.e. exponential random variables with rate $0 < \mu < \infty$). The dispatcher's goal is maximizing the population average over all arrivals of the net expected gains/profits.

The optimal admission policy of the dispatcher in [1] is a threshold policy: the dispatcher admits an arriving customer if and only if the queue-length upon arrival is strictly below a threshold. This threshold is characterized via the function $V : \mathbb{N} \times (0, \infty) \to [0, \infty)$, given by:

$$V(x, y) = \frac{x(y - \lambda) - \lambda(1 - (\lambda/y)^x)}{(y - \lambda)^2},$$

Funding agency: A.C. is partially supported by the NSF grant DMS-2006305; V.S. is supported in part by NSF grants CCF-2008130, ECCS-2038416 and CNS-1955777.

where the singularities of V at $y = \lambda$ are removable. Naor [1] showed that the optimal admittance threshold $\bar{K}$ is the only integer satisfying:

$$V\left(\bar{K}, \mu\right) \leq \frac{R}{C} < V\left(\bar{K} + 1, \mu\right). \tag{1}$$

When $V\left(\bar{K}, \mu\right) < R/C$, the optimal threshold is unique. However, when $V\left(\bar{K}, \mu\right) = R/C$, both thresholds $\bar{K}$ and $\bar{K} - 1$ are optimal; hence, so is any randomization between the two thresholds. In any case, when the dispatcher uses a threshold policy with a threshold K, what results is an $M/M/1/K$ queuing system.

We assume that the reward R, the cost per time unit C, and the arrival rate of customers λ are known to the learning dispatcher, but not the service rate μ. In our model the dispatcher continuously observes the queue length. Hence, we restrict the dispatcher to admission controls that at the time of a new arrival, admit or reject based on the entire history of the queue length until the arrival time. Therefore, when a new customer arrives, the dispatcher can estimate the mean service time (hence, the service rate) using the service times of the customers that have departed the queue before the new arrival.

We use the marker $\bar{}$ to denote processes associated with the *genie-aided system*. The processes without a marker are associated with the *learning system*. We let

- $\bar{Q}(t)$ and $Q(t)$ denote the queue length at time t,
- $\bar{Q}_i$ and Q_i denote the queue length right before the arrival of the i^{th} customer,
- $\bar{N}_J(t)$ and $N_J(t)$ denote the number of customers that joined the queue until and including time t.

We measure the performance of a policy chosen by the learning dispatcher by the regret it incurs. Namely, we take the difference between the expected net profit under the given control/policy and the best expected net profit the dispatcher could have obtained had it known the parameter μ. The regret $G(t)$ is given by

$$G(t) := \mathbb{E}\left[R\bar{N}_J(t) - C\int_0^t \bar{Q}(u)du - \left(RN_J(t) - C\int_0^t Q(u)du\right)\right].$$

We couple genie-aided and learning system by two independent Poisson processes $(P(t))_{t\geq 0}$ and $(L(t))_{t\geq 0}$ with rates μ and λ, respectively. Set the arrival processes of both systems to be L, and the head of the line customer of each system (assuming not empty) completes her service at the time of the

next jump of $P(t)$. Under the coupling, we can give an upper bound of $G(t)$ as:

$$G(t) \leq (R + \frac{C}{\lambda})\mathbb{E}\left[\sum_{i=1}^{N_A(t)} \left|\mathbb{1}_{\{\bar{Q}_i < \bar{K}\}} - \mathbb{1}_{\{Q_i < K_i\}}\right| + |\bar{Q}_i - Q_i|\right].$$

Following this bound, we can analyze the systems at the arrival epochs, and characterize the regret in terms of the total number of arrivals N. We use $\tilde{G}(N) := G(T_N^A)$ to denote the total regret accumulated up to the arrival of the N^{th} customer.

b) The Learning algorithm and main results: We propose (and study) an learning algorithm for learning-based social-welfare maximizing dispatch that consists of a sequence of batches, where each batch has two phases: phase 1 for exploration and phase 2 for exploitation. Specifically, every arriving customer will join the queue during phase 1 (assuming that a phase 1 is used); hence, the exploration title. At the end of this exploration phase, the algorithm determines an admittance threshold level that will be held fixed through the proceeding phase 2. During phase 2, the algorithm will use a threshold policy given by the threshold determined at the end of either phase 1 of the same batch or the previous batch. As the batch number increases, our algorithm will extend the length of the exploitation phase and reduce the occurrences of the exploration phases. The length of the exploration phase (if used) is fixed for all batches.

Theorem I.1. *Assume that the initial queue length for the learning and genie-aided systems are the same, and the threshold used in the genie-aided system is not 0, the proposed algorithm achieves $O(1)$ regret as $N \to \infty$, where N is the total number of arrivals. When threshold used in the genie-aided system can be 0, the proposed algorithm achieves $O(\ln^2(N))$ regret as $N \to \infty$.*

c) Related work: Learning unknown parameters to operate optimally in queuing systems, and analyzing queuing systems with model uncertainly have both been studied under various settings – see tutorial [2] for a recent overview. Our paper focuses on regret analysis in comparison to the optimal algorithm when the parameters are known. Under this framework, there is a growing literature considering different models and various types of regret. [3] considered a Erlang-B blocking system with unknown service rates, where a customer is either blocked or receives service immediately. They proposed an algorithm which observes the system upon arrivals, and converges to the optimal policy that either accept all customers when there is a free server or block all customers. In our setting, the queue has infinite capacity, customers may wait in the queue, and the dispatcher observes the whole history of the queue-length when making a decision. The net gain of accepting an customer in both works is only realized in the future, and the expected net gain needs knowledge of the service rate. Stability is always assured in [3] since the maximum system occupancy is bounded (finite number of servers with no queuing). In our problem, however, whereas the system is stable under the optimal policy, the maximum queue-length over the unknown parameter regime is unbounded. [4] considered a discrete-time parallel multi-server queuing system with multi-class customers and unknown service rates. They proposed a $c\mu$ rule based algorithm that achieves constant regret. For a discrete-time multi-class parallel-server system, when comparing to the algorithm which matches the queue to servers with the highest service probability, [5] used a multi-armed bandit view point and proposed Q-UCB and Q-Thompson sampling algorithms that achieve $O(\text{poly}(\log t)/t)$ queue-regret as the horizon t goes to ∞. [6] focused on a single-server discrete-time queue, and showed the existence of queue-length based polices that can achieve $O(1)$ regret. When each server has its own queue, [7] studied the discrete-time routing problem when service rate and queue-length are not known. Taking an MDP viewpoint, [8] proposed a learning algorithm that achieves $O(\sqrt{t})$ regret for an inventory control problem. [9] considered a queuing system with random network attributes and proposed a Max-Weight learning algorithm. Ojeda et al. [10] proposed a non-parametric model for service times and experimented with their algorithms in various settings. With model uncertainty, [11], [12], [13], [14] studied the heavy traffic regime for multi-class queue.

REFERENCES

[1] P. Naor, "The regulation of queue size by levying tolls," *Econometrica*, vol. 37, no. 1, p. 15–24, 1969.

[2] N. Walton and K. Xu, "Learning and information in stochastic networks and queues," in *TutORials in Operations Research*. INFORMS, 2021, pp. 161–198.

[3] S. Adler, M. Moharrami, and V. Subramanian, "Learning a discrete set of optimal allocation rules in queueing systems with unknown service rates," 2022. [Online]. Available: https://arxiv.org/abs/2202.02419

[4] S. Krishnasamy, A. Arapostathis, R. Johari, and S. Shakkottai, "On learning the cμ rule in single and parallel server networks," in *Allerton Conference*. IEEE, 2018, pp. 153–154.

[5] S. Krishnasamy, R. Sen, R. Johari, and S. Shakkottai, "Learning unknown service rates in queues: A multiarmed bandit approach," *Operations research*, vol. 69, no. 1, p. 315–330.

[6] T. Stahlbuhk, B. Shrader, and E. Modiano, "Learning algorithms for minimizing queue length regret," 2020. [Online]. Available: https://arxiv.org/abs/2005.05206

[7] T. Choudhury, G. Joshi, W. Wang, and S. Shakkottai, "Job dispatching policies for queueing systems with unknown service rates," in *ACM Mobihoc*. ACM, Jul 2021.

[8] S. Agrawal and R. Jia, "Learning in structured MDPs with convex cost functions: Improved regret bounds for inventory management," in *ACM EC*, ser. EC '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 743–744.

[9] M. J. Neely, S. T. Rager, and T. F. La Porta, "Max-Weight learning algorithms for scheduling in unknown environments," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1179–1191, 2012.

[10] C. Ojeda, K. Cvejoski, B. Georgiev, C. Bauckhage, J. Schuecker, and R. J. Sanchez, "Learning deep generative models for queuing systems," in *AAAI*, vol. 35, no. 10, May 2021, pp. 9214–9222.

[11] R. Atar, E. Castiel, and Y. Shadmi, "Scheduling in the high uncertainty heavy traffic regime," 2022.

[12] A. Cohen, "Asymptotic analysis of a multiclass queueing control problem under heavy traffic with model uncertainty," *Stochastic Systems*, vol. 9, no. 4, pp. 359–391, 2019.

[13] ——, "Brownian control problems for a multiclass M/M/1 queueing problem with model uncertainty," *Mathematics of Operations Research*, vol. 44, no. 2, pp. 739–766, 2019.

[14] A. Cohen and S. Saha, "Asymptotic optimality of the generalized cμ rule under model uncertainty," *Stochastic Processes and their Applications*, vol. 136, pp. 206–236, 2021.