Hybrid Priority Queue and its Applications

Zhouzi Li Mor Harchol-Balter* Alan Scheller-Wolf

ABSTRACT

Priority queues are well understood in queueing theory. However, they are somewhat restrictive in that the low-priority customers suffer far greater waiting times than the high-priority customers. In this short paper, we introduce a novel generalization of a two-class priority queue, which we call Hybrid. We prove that Hybrid has a much broader achievability region than strict priority, allowing for a much greater range of waiting time pairs. We demonstrate settings where this new flexibility can increase the revenue obtained by a service system (like airport TSA) selling priority.

1. INTRODUCTION

1.1 Non-Preemptive Priority Queue

This paper focuses on non-preemptive priority in a two-class system, where class 1 has priority over class 2, and the waiting times are denoted by random variables W_1 and W_2. It is well understood that $\mathbf{E}[W_2] > \mathbf{E}[W_1]$ and in fact it is often the case that $\mathbf{E}[W_2] \gg \mathbf{E}[W_1]$. This fact has been exploited to charge class 1 customers more money in exchange for offering them lower waiting time([8]).

1.2 Achievability Region of Priority Queue

We define the *achievability region* of two-class priority as the set of expected waiting time pairs $(\mathbf{E}[W_1], \mathbf{E}[W_2])$ which can be achieved as we consider all feasible arrival rates (λ_1, λ_2) of class 1 and 2. Note that this definition differs slightly from that in [3], because we allow for different arrival rates. In this way, we extend a stream of previous works using Achievability Region for optimal queue control (see [2] for a survey) to cases when arrival rates can also be controlled, e.g. by setting prices.

Figure 1 shows that the achievability region for strict non-preemptive priority consists of two narrow green "tornado"-shaped regions. This fact holds for any service time distributions of class 1 and class 2 customers, although in this work we only focus on the special case when both classes have service time drawn from the same distribution. While this fact is easy to prove, it is not prominent in the literature.

Figure 1 highlights the significant separation between the achievable waiting times in class 1 versus class 2. However what if one doesn't want such large separation? Consider a setting where a service provider charges customers to join queue 1. Could reducing the separation between $\mathbf{E}[W_1]$ and $\mathbf{E}[W_2]$ enable the service provider to make a larger profit? We next introduce a generalization of the strict priority setting which allows this to happen.

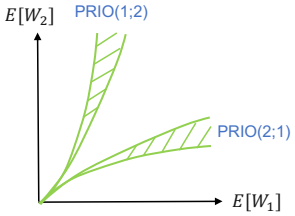

Figure 1: Achievability region of Strict Priority. Under PRIO(1;2), class 1 customers have stict priority. Under PRIO(2;1), class 2 customers have priority.

1.3 Hybrid Priority Queue

We introduce *Hybrid Priority*. Under Hybrid Priority (or Hybrid for short), there are still two queues, but whenever there are customers in both queues, the server serves a customer in queue 1 with probability q, say 70%, and serves a customer from queue 2 otherwise (see Figure 2). Obviously, Strict Priority is a special case of Hybrid where $q = 1$.

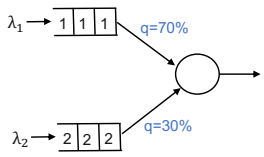

Figure 2: Hybrid queue system.

Under Hybrid, the service provider can adjust the level of priority of queue 1 customers over queue 2 customers. This allows for a much greater achievability region. In fact, Hybrid's achievability region includes the entire region between the achievability regions of Prio(1;2) and Prio(2;1), as shown in Figure 3 (proof omitted for lack of space). This gives the service provider much more flexibility in choosing the expected waiting times provided to customers.

1.4 A Simple Example: TSA Precheck

We now demonstrate the power of Hybrid in revenue maximization.

*All authors are at Carnegie Mellon University. This work is supported by NSF-IIS-2322973 and NSF-CMMI-1938909.

Copyright is held by author/owner(s).

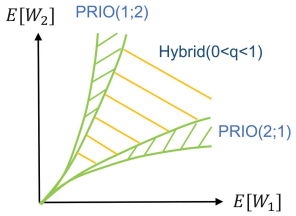

Figure 3: Achievability region for Hybrid.

Imagine two types of customers who need to go through airport security (TSA checks). Type A customers are *time-sensitive*, and are willing to pay for shorter waiting times, while type B customers are more patient, and don't have the money to pay for shorter waiting times.

The TSA knows that there are these two types of people and wants to leverage this fact to make some money. One common solution is that the TSA sells strict class 1 priority to type A customers (precheck) and puts type B customers in class 2. Note that the decision to buy priority is made before the customers come to the airport – hence, they decide based on an *unobservable* system, where only mean waiting times are known.

The service provider picks a price \$ for entering queue 1. Let λ_1 be the rate of type A customers who buy priority. Then the revenue rate made by the service provider is

$$\text{Revenue rate} = \lambda_1 \cdot \$.$$

The goal of the service provider is to choose \$ to maximize their revenue rate.

However in practice, the TSA cannot simply choose whatever price it wants, and it furthermore cannot simply delay class 2 customers indefinitely. In this paper we assume that the expected waiting time of any customer should not exceed $\overline{W}$ (say 30 minutes). To obey this constraint, the TSA must limit the number of priority (class 1) tickets that it sells; otherwise the waiting time for class 2 customers will exceed $\overline{W}$. Additionally, there is some upper limit (cap) on the price \$ that the TSA can charge. Thus there is a limit on the potential revenue rate for the TSA because there is a cap on the number of class 1 tickets that can be sold before conditions become intolerable for class 2 customers, and also a cap on the price.

Our paper shows that Hybrid may help the TSA increase the revenue. The intuition is as follows. In a Hybrid system, type A customers are still willing to pay (maybe less) for the partial priority. However, under partial priority, class 2 customers suffer less than they would have under Strict Priority. In this way, under Hybrid, more class 1 tickets can be sold before the waiting time of class 2 customers exceeds $\overline{W}$, hence potentially increasing revenue.

2. ANALYZING HYBRID

It is not known how to derive the waiting time under Hybrid(q) for a particular value of q. What makes analyzing Hybrid(q) difficult is that the state space for Hybrid(q) is infinite in 2 dimensions (one needs to track both the number of jobs in queue 1 and in queue 2). While all priority systems have a 2D-infinite state space, in the case of Prio(1;2) or Prio(2;1), we can use a "tagged job method" to derive the mean waiting time for each queue. Unfortunately, Hybrid(q) does not lend itself to such tagged analysis.

Fortunately, it turns out that we do not need to know the mean waiting time for Hybrid(q). It suffices to understand the range of waiting times spanned by Hybrid(q). In short, as shown in Figure 4, Hybrid(q) spans the full range from Prio(1;2) to Prio(2;1) as q runs from 0 to 1. Thus we know that there exists an $\alpha \in [0, 1]$ such that:

$$\mathbf{E}[W_1]^{Hybrid(q)} = \alpha \cdot \mathbf{E}[W_1]^{\text{Prio}(1;2)} + (1-\alpha)\mathbf{E}[W_1]^{\text{Prio}(2;1)},$$

$$\mathbf{E}[W_2]^{Hybrid(q)} = \alpha \cdot \mathbf{E}[W_2]^{\text{Prio}(1;2)} + (1-\alpha)\mathbf{E}[W_2]^{\text{Prio}(2;1)}.$$

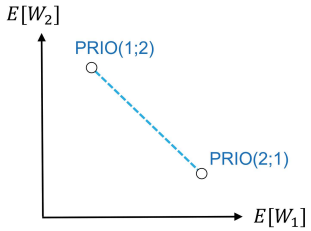

Figure 4: Hybrid spans the whole segment as q ranges from 0 to 1.

3. APPLICATION TO TSA

In this section, we describe our TSA model in more detail.

System: We assume for simplicity that there is a single server which serves customers. Customers are divided into two queues (queue 1 and queue 2). Customers have to pay a price, \$, to enter queue 1. Entering queue 2 is free. When the server is free, the server picks a queue according to the Hybrid(q) policy.

Customers: All customers have i.i.d. service requirement (service time need) drawn from the distribution denoted by random variable S, where the mean of S is $\mathbf{E}[S] = 1$. We further assume that customers arrive according to a Poisson process with rate $\overline{\lambda} < 1$.

Customers consist of two types, called type A and type B. Both types arrive according to a Poisson process, where type A customers arrive with rate λ_A, and type B customers arrive with rate λ_B, and $\lambda_A + \lambda_B = \overline{\lambda}$. Importantly, *all* customers are served and do not renege.

Type A customers are *time-sensitive*, meaning that they are willing to pay for shorter waiting time. Specifically, there is an impatience factor c associated with type A customers, where c is specified in dollars per unit waiting time. Thus if a type A customer experiences W waiting time, they will experience a cost of $c \cdot W$ dollars. By contrast, type B customers are not time-sensitive, meaning that they *never* pay for shortening their waiting time.

We say that a customer is *class 1* if she decides to buy priority (i.e., join queue 1). Those customers who choose not to buy priority are *class 2*. Note that class 1 consists of only type A customers (since they're the only ones willing to pay). However class 2 consists of both type A and type B customers. We denote by $\lambda_1 \leqslant \lambda_A$ the arrival rate of class 1 customers. We denote by λ_2 the arrival rate of class 2 customers.

Waiting Times: The waiting time of a customer is the time from when the customer arrives to the system until the customer first receives service. We use the r.v. W_1 to

denote the *waiting time* of class 1 customers, namely the time from when a customer joins queue 1 until they get served. Likewise r.v. W_2 will denote the waiting time of class 2 customers.

Type A customers are willing to pay for priority if and if only if the expected value of the reduction in their waiting time from buying priority is at least \$. Mathematically, a type A customer is willing to buy priority iff

$$c(\mathbf{E}[W_2] - \mathbf{E}[W_1]) \geqslant \$. \tag{1}$$

We assume that the government has placed a restriction of at most $\overline{W}$ on the mean waiting time of any customer. Thus we are restricted to:

$$\mathbf{E}[W_1] < \mathbf{E}[W_2] \leqslant \overline{W}. \tag{2}$$

Revenue: The *revenue* that the TSA brings in per unit time is defined as

$$\text{Revenue} := \lambda_1 \cdot \$.$$

There are some limitations on the revenue that the TSA can make. First, we assume the government has capped the price (\$) that we can charge at \$, i.e.,

$$\$ \leqslant \overline{\$}. \tag{3}$$

Second, because of the existence of (2), the TSA may limit the rate of sale of priority tickets (hence limiting the possible λ_1). Thus even if a type A customer is willing to buy priority, the TSA may not allow it.

Optimization Problem: The TSA's aim is to maximize its revenue. There are three variables that the TSA can optimize. First, TSA can choose a price $\$ \leqslant \overline{\$}$. Second, the TSA can set the rate of sale of queue 1 tickets λ_1. Finally, the TSA can choose its queueing policy, namely choose the parameter q of the Hybrid(q) policy.

However, the TSA is constrained in choosing parameters that satisfy all of the aforementioned constraints: (1),(2), (3). Thus in short, the TSA's optimization problem can be formulated as follows:

$$\begin{aligned}
\underset{\lambda_1, q, \$}{\text{maximize}} \quad & \$ \cdot \lambda_1 \\
\text{s.t.} \quad & \lambda_1 + \lambda_2 = \overline{\lambda}, \\
& c(\mathbf{E}[W_2 \mid \lambda_1, \lambda_2, q] - \mathbf{E}[W_1 \mid \lambda_1, \lambda_2, q]) \geqslant \$, \\
& \mathbf{E}[W_2 \mid \lambda_1, \lambda_2, q] \leqslant \overline{W}, \\
& \$ \leqslant \overline{\$}, \\
& \lambda_A \geqslant \lambda_1 \geqslant 0, \\
& 1 \geqslant q > 0.
\end{aligned}$$

To eliminate uninteresting cases, we assume that $\overline{W}$ is small enough that not all type A customers can be admitted to queue 1 (if this were not true, then Strict Priority is optimal). Likewise, we assume that $\overline{W}$ is large enough that there is a feasible solution to the optimization problem.

4. OUR RESULTS

Our paper derives the necessary and sufficient condition under which Hybrid outperforms Strict Priority. In short, when the cap on the price $\overline{\$}$ is high, Hybrid does not increase revenue over Strict Priority. However when the price cap, $\overline{\$}$, is low, Hybrid can increase revenue significantly compared to Strict Priority.

THEOREM 1 (STRICT PRIORITY WINS). *If $\overline{\$} \geqslant \overline{\lambda} \cdot \overline{W} \cdot c$, Prio(1;2) maximizes the revenue.*

THEOREM 2. *If $\overline{\$} < \overline{\lambda} \cdot \overline{W} \cdot c$, Hybrid increases revenue compared with Strict Priority. Moreover, the ratio of the optimal revenue, Rev^{Hybrid}, earned by Hybrid to the optimal revenue, Rev^{Prio}, earned by Strict Priority is:*

$$\frac{Rev^{Hybrid}}{Rev^{Prio}} = \min\left\{ \frac{\overline{\lambda} t_2}{t_2 - t_1} \ , \ \frac{\lambda_A}{1 - \frac{\overline{\lambda}\mathbf{E}[S_e]}{t_2(1-\overline{\lambda})}} \right\}.$$

Note that the ratio given in Theorem 2 can approach infinity under some specific parameters. As a practical example, suppose that $\overline{W} = 30$ minutes, the price cap $\overline{\$} = 15$ dollars, the impatience factor is $c = 1$, $\lambda_A = 0.6$, $\lambda_B = 0.35$, and the customer service times are exponentially distributed with rate 1. Then the improvement factor of Hybrid over Strict Priority is about 60%.

5. RELATED WORK

The book [4] thoroughly surveys how to maximize revenue in queueing systems. For this short paper, we focus on just prior work related to "partial" (hybrid-like) priority.

While we believe that the concept of Hybrid priority as we've defined it is novel, there are other related notions of priority in the literature. One example is Discriminatory Processor Sharing (DPS), where the server is time-shared between the two queues *preemptively*, with each queue getting some fraction of the server, see [6, 7]. DPS is different from Hybrid because it is preemptive. The one paper that we've found that uses DPS to maximize revenue is [5]. The authors provide numerical examples showing that DPS can be helpful in maximizing revenue.

There are also a few papers looking at using some kind of partial priority to obtain closer waiting times between two classes ([1, 9]). These papers deal only with waiting time targets and do not talk about explicit queueing policies to achieve them.

6. REFERENCES

[1] P. Afèche. Incentive-compatible revenue management in queueing systems: Optimal strategic delay. *Manufacturing & Service Operations Management*, 15(3):423–443, 2013.

[2] D. Bertsimas. The achievable region method in the optimal control of queueing systems; formulations, bounds and policies. *Queueing systems*, 21:337–389, 1995.

[3] E. G. Coffman Jr and I. Mitrani. A characterization of waiting time performance realizable by single-server queues. *Operations Research*, 28(3-part-ii):810–821, 1980.

[4] R. Hassin and M. Haviv. *To queue or not to queue: Equilibrium behavior in queueing systems*, volume 59. Springer Science & Business Media, 2003.

[5] R. Hassin and M. Haviv. Who should be given priority in a queue? *Operations Research Letters*, 34(2):191–198, 2006.

[6] R. Hassin, J. Puerto, and F. R. Fernández. The use of relative priorities in optimizing the performance of a queueing system. *European Journal of Operational Research*, 193(2):476–483, 2009.

[7] M. Haviv and J. van der Wal. Equilibrium strategies for processor sharing and random queues with relative priorities. *Probability in the Engineering and Informational Sciences*, 11(4):403–412, 1997.

[8] H. Mendelson and S. Whang. Optimal incentive-compatible priority pricing for the M/M/1 queue. *Operations Research*, 38(5):870–883, 1990.

[9] L. Yang, S. Cui, and Z. Wang. Design of covid-19 testing queues. *Production and Operations Management*, 31(5):2204–2221, 2022.