INDIAN INSTITUTE OF MANAGEMENT
AHMEDABAD • INDIA

Priority Service System Optimization under Service Level Constraints

Sachin Jayaswal

W.P. No. 2014-08-04

August 2014

INDIAN INSTITUTE OF MANAGEMENT
AHMEDABAD-380 015
INDIA

Priority Service System Optimization under Service Level Constraints

Sachin Jayaswal

Production & Quantitative Methods, Indian Institute of Management Ahmedabad
sachin@iimahd.ernet.in

Abstract

We consider a service system that serves one class of customers, which is willing to pay a premium for a faster delivery, with priority over the other class, which is more price sensitive but is willing to wait longer. The demand from one class depends not only on the price and delivery time quoted to it, but also on that offered to the other class. The service provider needs to select the price and delivery time quoted to the two classes, and the required service capacity to meet the quoted delivery times with a certain degree of reliability, so as to maximize its rate (per unit time) of earning profit. This results in a non-linear priority queue optimization model, for which the analytical expression for service level constraint for the low priority customers is unknown. We provide a cutting plane method to solve the problem, where constraints to be satisfied are identified iteratively from results of matrix geometric evaluation of the proposed system alternative, which are then added to the mathematical model for re-optimization.

Keywords: Priority queue, Waiting time distribution, Matrix geometric method, Cutting plane method

Priority Service System Optimization under Service Level Constraints

1 Introduction

We consider a server that serves two different classes of customers, called high priority (indexed by h) and low priority (indexed by l). High priority customers are valuable as they are willing to pay a comparatively higher price ($p_h > p_l$) for their service. In return, they are guaranteed a comparatively shorter delivery time ($L_h < L_l$). Of course, given the variability in the demand and the service process, the server can never meet the guaranteed service level, no matter how pessimistic, with 100% reliability. The server, therefore, uses a service level guarantee, α^h and α^l, with which it targets to meet its quoted delivery times L_h and L_l. That is, the actual waiting time, W_h or W_l, of a customer until she is served should not exceed her guaranteed delivery time, L_h or L_l, with a probability of at least α^h or α^l. High priority customers are always served in (preemptive) priority over low priority customers, irrespective of their order of arrivals. Customers from the same class are, however, served on a first-come-first-serve (FIFO) basis. In real life, such a server may be representative of an airline check-in counter serving both the business and the economy class; or a call center serving both regular and priority calls. The incentive to use such a priority scheme stems from the relatively shorter delivery time guaranteed to the high priority customers for which they are willing to pay a price premium.

Customers arrive for service according to a Poisson process with rates λ_h and λ_l. Customer demands are sensitive to their respective prices and guaranteed delivery times, and also to their relative values $(p_h - p_l)$ and $(L_l - L_h)$. The service provider can, therefore, attract new customers through price reductions or by offering shorter delivery times. Lowering the price or delivery time for one class also induces the other group of customers to switch classes. The demand rates are described using the following linear functions [4], [20]:

$$\lambda_h = a - \beta_p p_h + \theta_p(p_l - p_h) - \beta_L L_h + \theta_L(L_l - L_h) \tag{1}$$

$$\lambda_l = a - \beta_p p_l + \theta_p(p_h - p_l) - \beta_L L_l + \theta_L(L_h - L_l) \tag{2}$$

where,

$2a$: potential market size, i.e., total demand if price and delivery time for the service is zero

β_p : sensitivity of demand to price

β_L : sensitivity of demand to the guaranteed delivery time

θ_p : sensitivity of demand switchovers to the price difference $(p_h - p_l)$

θ_L : sensitivity of demand switchovers to the difference in guaranteed delivery times $(L_h - L_l)$

Customers from either class have service times that are exponentially distributed with the same rate μ since they differ only in their price and time sensitivity, and not in the work content they present to the server. In the call center example, this is tantamount to saying that the high and low priority customers have similar call durations. It costs the server $\$m$ per customer in operation and $\$A$ per customer per unit time in capacity. The objective of the service provider is to price its service for each customer class and offer them appropriate delivery times so as to maximize its rate (per unit time) of earning profit. At the same time, it needs to decide on an optimal service rate μ, in order to guarantee the service levels of at least α^h and α^l. The service provider's optimization problem can stated as the follows:

$$[PQO] : \max \quad \pi(p_h, p_l, \lambda_h, \lambda_l, \mu) = (p_h - m)\lambda_h + (p_l - m)\lambda_l - A\mu \tag{3}$$

$$\text{s.t.} \quad S_h(L_h) = P(W_h \leq L_h) = 1 - e^{(\lambda_h - \mu)L_h} \geq \alpha^h \tag{4}$$

$$S_l(L_l) = P(W_l \leq L_l) \geq \alpha^l \tag{5}$$

$$\lambda_h + \lambda_l - \mu < 0 \tag{6}$$

$$p_h, p_l, \lambda_h, \lambda_l, \mu > 0 \tag{7}$$

Constraints (4) and (5) are delivery time reliability constraints. (4) uses the result that the tail of the sojourn time distribution for high priority customers in a preemptive priority queue is known to be exponential. However, there is no exact closed form expression for the sojourn time distribution for the low priority customers, appearing in (5) . Constraint (6) is the stability condition for the queuing system. Constraint set (7) is needed to define realistic parameter values. We note that other objective functions are also be plausible. For example, [21] minimizes the total cost per unit time incurred by the server, which is a sum of its capacity induced cost and the expected delay costs to it customers.

Variants of [PQO] are common in the literature on product pricing. [5] and [19] have earlier studied a similar problem of a shared capacity, but they have used expected performance measures instead of the entire distribution, which eliminates the challenges inherent in our model. [4], [13], [14], [16], [17], [18] use the entire distribution of waiting times, but they either consider a single class of customer or a dedicated server for each class, and can thus use closed form constraints like (4). What makes the current model challenging is the use of service level constraint (5) in a shared capacity environment, which cannot be expressed analytically.

Using the linear demand function described by (1) and (2), [PQO] can be restated as:

$$\begin{aligned}
[PQO] : \max \quad & \pi(p_h, p_l, \mu) = -(\beta_p + \theta_p)p_h^2 - (\beta_p + \theta_p)p_l^2 + 2\theta_p p_h p_l + \\
& (-\beta_L L_h + \theta_L L_l - \theta_L L_h + m\beta_p + a)p_h + \\
& (-\beta_L L_l + \theta_L L_h - \theta_L L_l + m\beta_p + a)p_l - A\mu + \beta_L L_h m + \beta_L L_l m - 2ma \qquad (8) \\
\text{s.t.} \quad & -(\beta_p + \theta_p)p_h + \theta_p p_l - \mu \leq \frac{\log(1-\alpha^h)}{L_h} - a + \beta_L L_l - \theta_L(L_h - L_l) \qquad (9) \\
& S_l(L_l) = P(W_l \leq L_l) \geq \alpha^l \qquad (10) \\
& -\beta_p p_h - \beta_p p_l - \mu < \beta_L L_h + \beta_L L_l - 2a \qquad (11) \\
& -(\beta_p + \theta_p)p_h + \theta_p p_l \geq \beta_L L_h + \theta_L L_h - \theta_L L_l - a \qquad (12) \\
& \theta_p p_h - (\beta_p + \theta_p)p_l \geq \beta_L L_l - \theta_L L_h + \theta_L L_l - a \qquad (13) \\
& p_h, p_l, \mu > 0 \qquad (14)
\end{aligned}$$

Thus, the model has a quadratic objective function with linear constraints, except constraint (10) for which the exact analytical form is not known. There are approximations proposed in the literature for $S_l(\cdot)$. However, they are complex and often not sufficiently accurate [1]. Moreover, the appropriate approximation to use depends on the relative demand rates, which may only be determined endogenously and not known in advance. The [PQO] model, therefore, turns out to be challenging, which does not lend itself easily to conventional optimization methods.

In the next section, we describe a proposed solution approach, which uses the *matrix geometric* method in conjunction with *cutting plane* optimization method. Some of the recent works (e.g., [2]) have used simulation in conjunction with *cutting plane* method to optimize complex queuing systems. However, the strength of our solution approach lies in the fact that it can find an exact solution in contrast with simulation, which at best gives point estimates. Moreover, our solution approach is computationally efficient compared to simulation. This is important especially in applications [6], [7] where one needs to run several experiments with different parameter values. The proposed approach also has its limitations in that it cannot work for some of the more complex queuing systems that simulation can handle.

2 Solution Methodology

In this section, we describe our solution approach for [PQO]. Before describing the solution procedure, we state the following important result.

Proposition 1: If $S_l(\cdot)$ is concave, [PQO] has a unique optimal solution. **proof:** The Hessian for (8) is given by:

$$\begin{pmatrix} -2(\beta_p+\theta_p) & 2\theta_p & 0 \\ 2\theta_p & -2(\beta_p+\theta_p) & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

The first principal minor of the Hessian is negative, the second principal minor is positive, while the third principal minor is 0. Therefore, the objective function (8) is concave. All the constraints except (10) are linear. If $S_l(\cdot)$ in constraint (10) is concave, then any point satisfying the Kuhn-Tucker conditions will optimally solve PQO [3].

It is important to note that the above proposition assumes the concavity of $S_l(\cdot)$. Our initial computational results (obtained using a method to be described in subsequent sections) show that this is a reasonable assumption. Plots of $S_l(\cdot)$ vs. $(p_h,\, p_l)$, and $S_l(\cdot)$ vs. μ are shown in Figure 1. These plots suggest concavity of $S_l(\cdot)$ with respect to $(p_h,\, p_l)$ and separately with respect to μ. However, this does not necessarily justify the joint concavity of $S_l(\cdot)$ with respect to $(p_h,\, p_l,\, \mu)$. We will, therefore, integrate in our solution method a mechanism to make sure that the concavity assumption is not violated.

Assuming $S_l(\cdot)$ is concave, we can approximate it by a set of tangent hyperplanes at various points $(p_h^k,\, p_l^k,\, \mu^k)$, $\forall\, k \in K$, that is

$$S_l(\cdot) = \min_{k\in K}\left\{ S_l^k(\cdot) + (p_h - p_h^k)\left(\frac{\partial S_l^k(\cdot)}{\partial p_h}\right) + (p_l - p_l^k)\left(\frac{\partial S_l^k(\cdot)}{\partial p_l}\right) + (\mu - \mu^k)\left(\frac{\partial S_l^k(\cdot)}{\partial \mu}\right)\right\}$$

where $S_l^k(\cdot)$ denotes the cumulative distribution function of W_l, while $\frac{\partial S_l^k(\cdot)}{\partial p_h}$, $\frac{\partial S_l^k(\cdot)}{\partial p_l}$ and $\frac{\partial S_l^k(\cdot)}{\partial \mu}$ are the partial gradients of $S_l(\cdot)$ at a fixed point (p_h^k, p_l^k, μ^k). This means constraint (10) can be replaced by the following set of linear constraints:

$$S_l^k(\cdot) + (p_h - p_h^k)\left(\frac{\partial S_l^k(\cdot)}{\partial p_h}\right) + (p_l - p_l^k)\left(\frac{\partial S_l^k(\cdot)}{\partial p_l}\right) + (\mu - \mu^k)\left(\frac{\partial S_l^k(\cdot)}{\partial \mu}\right) \geq \alpha^l \qquad \forall k \in K \tag{15}$$

Let us denote the resulting mathematical model by $[PQO_{(K)}]$. We propose to use the *matrix geometric* method to numerically evaluate $S_l(\cdot)$ and its partial gradients, which is described next.

2.1 Estimation of $S_l(\cdot)$ and its Gradient

In this section, we describe the use of *matrix geometric method* to numerically evaluate $S_l^k(\cdot)$ at a given point $(p_h^k,\, p_l^k,\, \mu^k)$, and show how to use this method to obtain its gradient (we refer our

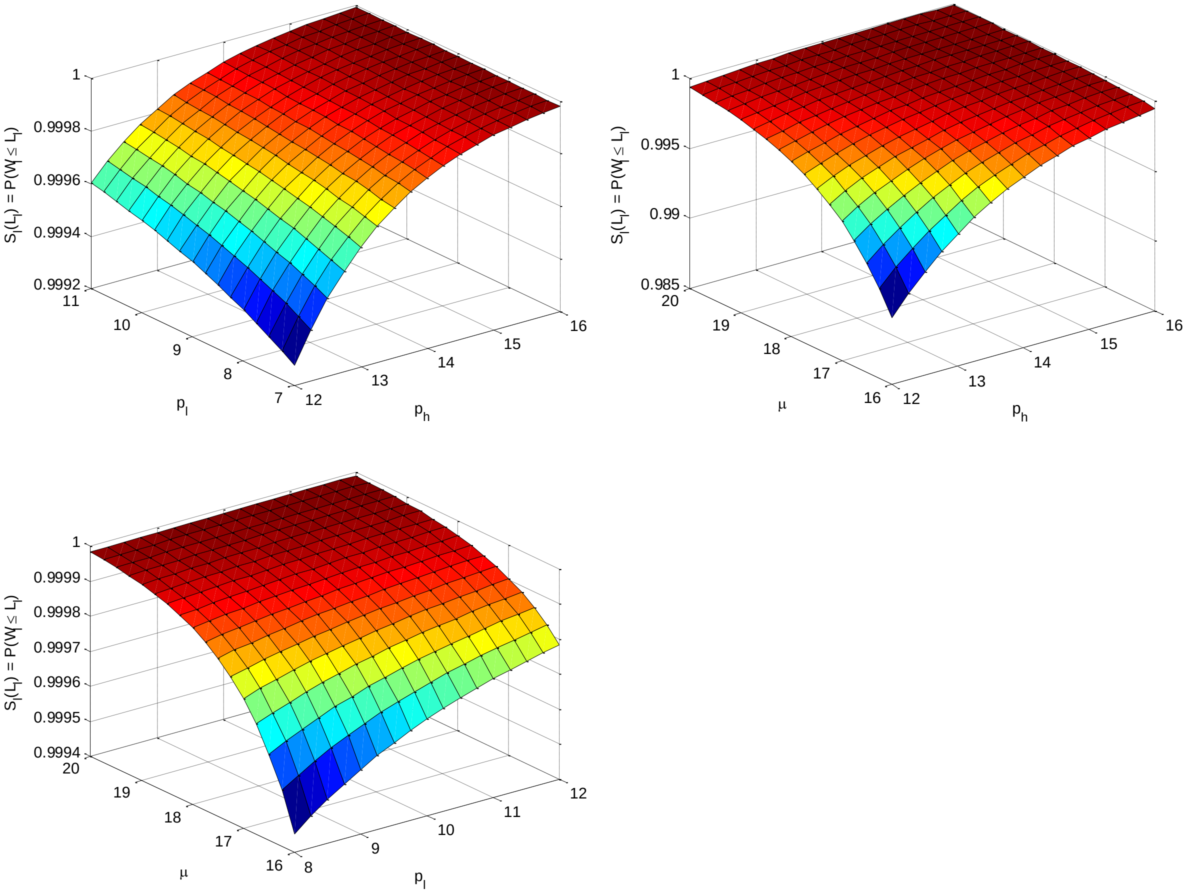

Figure 1: service level vs. prices and capacity

reader to [12] for a detailed discussion, and to [11] for an excellent tutorial on the subject). To evaluate $S_l^k(\cdot)$, we first need to obtain the joint stationary distribution of high and low priority customers in the queue.

2.1.1 The Matrix Geometric Method for Joint Stationary Distribution

If we choose $N_h(t)$ and $N_l(t)$ as state variables representing the number of high and low priority customers in the system, then $\{\mathbf{N}(t)\} = \{N_l(t), N_h(t), t \geq 0\}$ is a continuous-time two-dimensional Markov chain with state space $\{\mathbf{n} = (n_l, n_h) | n_i \geq 0, n_h \leq M; i = l, h\}$. The key idea we employ here is that the Markov process $\{\mathbf{N}(t)\}$ is *quasi-birth-and-death* (QBD), which allows us to develop a *matrix geometric* solution for the joint distribution of the number of customers of each class in the system. A simple implementation of the *matrix geometric* method, however, requires the number of states in the QBD process to be finite. For this, we treat the queue length of high priority customers (including the one in service) to be of finite size M, but of size large enough for the desired accuracy of our results. Since high priority customers are always served in priority over low priority customers, its queue size will always be bounded by some large number.

A transition can occur only if a customer of either class arrives or a customer of either class is served. The possible transitions are:

From	To	Rate	Condition
(n_l, n_h)	$(n_l, n_h + 1)$	λ_h	for $n_l \geq 0$, $0 \leq n_h < M$
(n_l, n_h)	$(n_l + 1, n_h)$	λ_l	for $n_l \geq 0$, $0 \leq n_h \leq M$
(n_l, n_h)	$(n_l, n_h - 1)$	μ	for $n_l \geq 0$, $0 \leq n_h \leq M$
(n_l, n_h)	$(n_l - 1, n_h)$	μ	for $n_l > 0$, $n_h = 0$

The infinitesimal generator associated with our system description is block-tridiagonal, represented as:

$$Q = \begin{pmatrix} B_0 & A_0 & & & \\ A_2 & A_1 & A_0 & & \\ & A_2 & A_1 & A_0 & \\ & & \ddots & \ddots & \ddots \end{pmatrix}$$

where B_0, A_0, A_1, A_2 are square matrices of order $M+1$. These matrices can be easily constructed using the transition rates described above.

$$A_0 = \begin{pmatrix} \lambda_l & & & & \\ & \lambda_l & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & \lambda_l \end{pmatrix} ; \quad A_2 = \begin{pmatrix} \mu & & & & \\ & 0 & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & 0 \end{pmatrix} ; \quad B_0 = \begin{pmatrix} * & \lambda_h & & & \\ \mu & * & \lambda_h & & \\ & \mu & * & \lambda_h & \\ & & \ddots & \ddots & \ddots \\ & & & \mu & * \end{pmatrix}$$

where $*$ is such that $A_0\mathbf{1} + B_0\mathbf{1} = \mathbf{0}$, and $\mathbf{1}$ is a column vector of ones of size $M+1$. $A_1 = B_0 - A_2$.

We denote by $\mathbf{x}$ the stationary probability vector of $\{\mathbf{N}(t)\}$, where $\mathbf{x}$ is represented as:

$$\mathbf{x} = [x_{00}, x_{01}, \ldots, x_{0M}, x_{10}, x_{11}, \ldots, x_{1M}, \ldots, \ldots, x_{i0}, x_{i1}, \ldots, x_{iM}, \ldots, \ldots]$$

The vector $\mathbf{x}$ can be partitioned by levels into sub vectors $\mathbf{x}_i$, $i \geq 0$, where $\mathbf{x}_i = [x_{i0}, x_{i1}, \ldots, x_{iM}]$ is the stationary probability of states in level i ($n_l = i$). Thus, $\mathbf{x} = [\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \ldots, \ldots]$ can be obtained using a set of balance equations, given in matrix form by the following standard relations [9], [11], [12]:

$$\mathbf{x}Q = \mathbf{0} \quad ; \qquad\qquad \mathbf{x}_{i+1} = \mathbf{x}_i R$$

where R is the minimal non-negative solution to the matrix quadratic equation:

$$A_0 + RA_1 + R^2 A_2 = \mathbf{0}$$

The matrix R can be computed using well known methods [9], [11], [12]. A simple iterative procedure often used is:

$$R(n+1) = -\left[A_0 + R^2(n)A_2\right] A_1^{-1} \; ; \qquad\qquad R(0) = 0$$

The probabilities $\mathbf{x}_0$ are determined from:

$$\mathbf{x}_0(B_0 + RA_2) = \mathbf{0}$$

subject to the normalization equation:

$$\sum \mathbf{x}_k \mathbf{1} = \mathbf{x}_0 (I - R)^{-1} \mathbf{1} = 1$$

where $\mathbf{1}$ is a column vector of ones and is of size $M + 1$.

2.1.2 Matrix Geometric Method for Estimation of $S_l(\cdot)$

The delivery time, W_l, of a low priority customer is the time between its arrival to the system till it completes its service. It may, however, be preempted by one or more high priority customers for service. It is, therefore, difficult in general to characterize the CDF, $S_l(\cdot)$, of the time spent in system by low priority customers. [15] presents an efficient algorithm based on *unifromization* to derive the complimentary distribution of waiting times in phase-type and QBD processes. The same approach is used in [10] to derive the complimentary distribution of waiting times in a more complex queuing system. We adopt their algorithm to derive $S_l(\cdot)$.

Consider a tagged low priority customer entering the system. The time spent by the tagged customer depends on the number of customers of either class already present in the system ahead of it, and also on the number of high priority arrivals before it completes its service. All further low priority arrivals, however, have no influence on its time spent in the system. The tagged customer's time in the system is, therefore, simply the time until absorption in a modified Markov process $\{\tilde{\mathbf{N}}(t)\}$, obtained by setting $\lambda_l = 0$. Consequently, matrix $\tilde{A}_0$, representing transitions to a higher level, becomes a zero matrix. We define an *absorbing* state, call it state $0^{'}$, as the state in which the tagged customer has finished its service. The infinitesimal generator for this process can be represented as:

$$\tilde{Q} = \left(\begin{array}{c|cccccc} 0 & 0 & 0 & 0 & 0 & \cdots \\ \hline b_0 & \tilde{B}_0 & 0 & & & \\ 0 & A_2 & \tilde{A}_1 & 0 & & \\ 0 & & A_2 & \tilde{A}_1 & 0 & \\ \vdots & & & \ddots & \ddots & \ddots \end{array}\right)$$

where, $\tilde{B}_0 = B_0 + A_0$; $\tilde{A}_1 = A_1 + A_0$; and $b_0 = [\mu \quad 0 \quad \cdots \quad 0]^T_{M+1}$. The first row and column in $\tilde{Q}$ corresponds to the absorbing state $\acute{0}$. The time spent in system by the tagged customer, which is the time until absorption in the modified Markov process with rate matrix $\tilde{Q}$, depends on the prices (p_h and p_l), through the arrival rates (λ_h and λ_l), and the service rate μ. For given prices (p_h^k, p_l^k) and service rate μ^k, the CDF of the time spent by a low priority customer in the system is $S_l^k(y) = 1 - \overline{S_l^k}(y)$, where $\overline{S_l^k}(y)$ is the stationary probability that a low priority customer spends more than y units of time in the system. Further, let $\overline{S_{li}^k}(y)$ denote the conditional probability that a tagged customer, who finds i low priority customers ahead of it, has to spend a time exceeding y in the system. The probability that a tagged customer finds i low priority customers is given, using the PASTA property, by $\mathbf{x}_i = \mathbf{x}_0 R^i$, as derived in the previous section. $\overline{S_l^k}(y)$ can be expressed as:

$$\overline{S_l^k}(y) = \sum_{i=0}^{\infty} \mathbf{x}_i \overline{S_{li}^k}(y) \tag{16}$$

$\overline{S_{li}^k}(y)$ can be computed more conveniently by uniformizing the Markov process $\{\tilde{\mathbf{N}}(t)\}$ with a

Poisson process with rate γ, where

$$\gamma = \max_{0 \le j \le M} (-\tilde{Q})_{jj} = \max_{0 \le j \le M} (-\tilde{A}_1)_{jj} = \max_{0 \le j \le M} -(A_0 + A_1)_{jj}$$

so that the rate matrix $\tilde{Q}$ is transformed into the discrete-time probability matrix:

$$\hat{Q} = \frac{1}{\gamma}\tilde{Q} + I = \left(\begin{array}{c|cccccc} 1 & 0 & 0 & 0 & 0 & \cdots \\ \hline \hat{b_0} & \hat{B}_0 & 0 & & & \\ 0 & \hat{A}_2 & \hat{A}_1 & 0 & & \\ 0 & & \hat{A}_2 & \hat{A}_1 & 0 & \\ \vdots & & & \ddots & \ddots & \ddots \end{array} \right)$$

where $\hat{A}_2 = \frac{A_2}{\gamma}$, $\hat{A}_1 = \frac{\tilde{A}_1}{\gamma} + I$, $\hat{b_0} = \frac{b_0}{\gamma}$. In this uniformized process, points of a Poisson process are generated with rate γ and transitions occur at these epochs only. The probability that n Poisson events are generated in time y equals $e^{-\gamma y}\frac{(\gamma y)^n}{n!}$. Suppose the tagged customer finds i low priority customers ahead of it. Then, for its time in system to exceed y, at most i of the n Poisson points may correspond to transitions to lower levels (i.e., service completions of low priority customers). Therefore,

$$\overline{S^k_{li}}(y) = \sum_{n=0}^{\infty} e^{-\gamma y}\frac{(\gamma y)^n}{n!} \sum_{v=0}^{i} G_v^{(n)}\mathbf{1}, \qquad i \ge 0 \tag{17}$$

where, $G_v^{(n)}$ is a matrix such that its entries are the conditional probabilities, given that the system has made n transitions in the discrete-time Markov process with rate matrix $\hat{Q}$, that v of those transitions correspond to lower levels (i.e., service completions of low priority customers). Substituting the expression for $\overline{S^k_l}_i(y)$ from (17) in (16), we obtain:

$$\overline{S^k_l}(y) = \sum_{n=0}^{\infty} d_n e^{-\gamma y}\frac{(\gamma y)^n}{n!} \tag{18}$$

where, d_n is given by:

$$d_n = \sum_{i=0}^{\infty} \mathbf{x}_0 R^i \sum_{v=0}^{i} G_v^{(n)} \mathbf{1}, \qquad n \geq 0 \qquad (19)$$

Now,

$$\begin{aligned}
\sum_{i=0}^{\infty} R^i \sum_{v=0}^{i} G_v^{(n)} \mathbf{1} &= \sum_{i=0}^{n+1} R^i \sum_{v=0}^{i} G_v^{(n)} \mathbf{1} + \sum_{i=n+2}^{\infty} R^i \sum_{v=0}^{n} G_v^{(n)} \mathbf{1} && (\text{since} \quad K_v^n = 0 \quad \text{for} \quad v > n) \\
&= \sum_{v=0}^{n+1} \sum_{i=v}^{n+1} R^i G_v^{(n)} \mathbf{1} + (I-R)^{-1} R^{n+2} \mathbf{1} && \left(\text{since} \quad \sum_{v=0}^{n} G_v^{(n)} \mathbf{1} = \mathbf{1}\right) \\
&= \sum_{v=0}^{n+1} (I-R)^{-1} (R^v - R^{n+2}) G_v^{(n)} \mathbf{1} + (I-R)^{-1} R^{n+2} \mathbf{1} && \left(\text{since} \quad \sum_{v=0}^{n+1} G_v^{(n)} \mathbf{1} = \mathbf{1}\right) \\
&= \sum_{v=0}^{n} (I-R)^{-1} R^v G_v^{(n)} \mathbf{1} + (I-R)^{-1} R^{n+1} G_{n+1}^{(n)} \\
&= \sum_{v=0}^{n} (I-R)^{-1} R^v G_v^{(n)} \mathbf{1} && \left(\text{since} \quad G_v^{(n)} = 0 \quad \text{for} \quad v > n\right) \\
&= (I-R)^{-1} H_n \mathbf{1} && n \geq 0
\end{aligned}$$

where, $H_n = \sum_{v=0}^{n} R^v G_v^{(n)}$. Therefore,

$$S_l^k(L_l) = 1 - \overline{S_l^k}(L_l) = 1 - \sum_{n=0}^{\infty} e^{-\gamma L_l} \frac{(\gamma L_l)^n}{n!} \mathbf{x}_0 (I-R)^{-1} H_n \mathbf{1} \qquad (20)$$

H_n can be computed recursively as:

$$H_{n+1} = H_n \hat{A}_1 + R H_n \hat{A}_2; \quad H_0 = I$$

where, I is an identity matrix of size $M+1$. Therefore, for given prices $(p_h^k,\, p_l^k)$ and service rate (μ^k), CDF, $S_l^k(\cdot)$ of W_l in (15) can be computed using (20). We next describe the procedure to numerically estimate the gradients of $S_l(\cdot)$ used in (15).

2.1.3 Estimation of Gradient of $S_l(\cdot)$

There are several methods available in the literature to compute the gradients of the CDF $S_l(\cdot)$ of W_l. We use a *finite difference* method as it is probably the simplest and most intuitive, and can be easily explained (e.g. [2]). Using the *finite difference* method, the gradients can be computed as:

$$\frac{\partial S_l^k(\cdot)}{\partial p_h} = \frac{S_l^{(p_h^k+dp_h,p_l,\mu)}(\cdot) - S_l^{(p_h^k-dp_h,p_l,\mu)}(\cdot)}{2dp_h}$$
$$\frac{\partial S_l^k(\cdot)}{\partial p_l} = \frac{S_l^{(p_h,p_l^k+dp_l,\mu)}(\cdot) - S_l^{(p_h,p_l^k-dp_l,\mu)}(\cdot)}{2dp_l}$$
$$\frac{\partial S_l^k(\cdot)}{\partial \mu} = \frac{S_l^{(p_h,p_l,\mu^k+d\mu)}(\cdot) - S_l^{(p_h,p_l,\mu^k-d\mu)}(\cdot)}{2d\mu}$$

where dp_h, dp_l and $d\mu$ (referred to as step sizes) are infinitesimal changes in the respective variables. These estimates of the gradients are used in the *cutting plane* algorithm to generate constraints/cuts of the form (15).

2.2 The Cutting Plane Algorithm

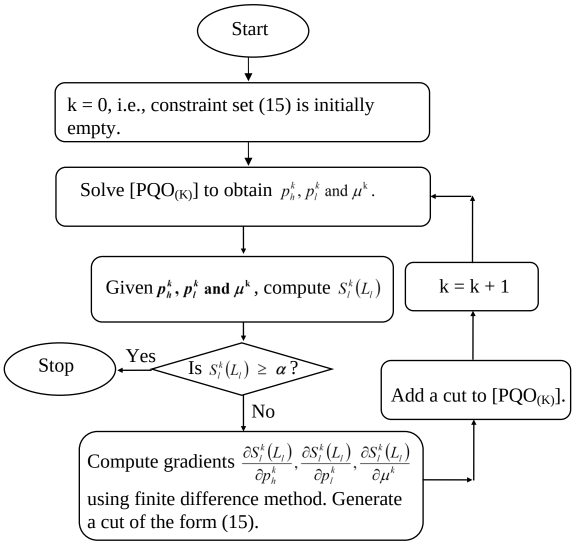
Start
k = 0, i.e., constraint set (15) is initially empty.
Solve [PQO$_{(K)}$] to obtain p_h^k, p_l^k and μ^k.
Given p_h^k, p_l^k and μ^k, compute $S_l^k(L_l)$
k = k + 1
Stop
Yes
Is $S_l^k(L_l) \geq \alpha$?
Add a cut to [PQO$_{(K)}$].
No
Compute gradients $\frac{\partial S_l^k(L_l)}{\partial p_h^k}, \frac{\partial S_l^k(L_l)}{\partial p_l^k}, \frac{\partial S_l^k(L_l)}{\partial \mu^k}$ using finite difference method. Generate a cut of the form (15).

Figure 2: Cutting Plane Algorithm

In this section, we describe the *cutting plane* algorithm to solve $[PQO_{(K)}]$. The algorithm fits the framework of Kelley's *cutting plane* method [8]. It differs from the traditional description of the algorithm in that we use *matrix geometric method* to generate the cuts and evaluate the function values instead of having an algebraic form for the function and using analytically determined gradients to generate the cuts. Figure 2 shows a flowchart of the *cutting plane* algorithm. The algorithm works as follows: We start with an empty constraint set (15), which results in a simple quadratic programming problem (QPP), and obtain an initial solution (p_h^0, p_l^0, μ^0). We use the *matrix geometric* method to compute the CDF, $S_l^{(p_h^0, p_l^0, \mu^0)}(\cdot)$, of W_l. If $S_l^{(p_h^0, p_l^0, \mu^0)}(\cdot)$ meets the minimum required service level α^l, we stop with an optimal solution to $[PQO_{(K)}]$, else we add to (15) a linear constraint/cut generated using the *finite difference* method. The new cut eliminates the current solution but does not eliminate any feasible solution to $[PQO_{(K)}]$. This procedure repeats until the delivery reliability constraint is satisfied within a sufficiently small tolerance limit ϵ ($\left|S_l - \alpha^l\right| \leq \epsilon$). The method has been proved to converge [2].

The success of the cutting plane algorithm relies on the concavity of $S_l(\cdot)$. We have already demonstrated, using computational results obtained by the *matrix-geometric* method, that $S_l(\cdot)$ is concave in (p_h, p_l) and separately concave in μ. However, it is difficult to establish the joint concavity of $S_l(\cdot)$ in (p_h, p_l, μ). If the concavity assumption is violated, then the algorithm may cut off parts of the feasible region and terminate with a solution that is suboptimal. We, therefore, need to include a test to ensure the concavity assumption is not violated. This is done by ensuring that a new point, visited by the *cutting plane* algorithm after each iteration, lies below all the previously defined cuts, and that all previous points lie below the newly added cut. The test, however, cannot ensure that $S_l(\cdot)$ is concave unless it examines all the points in the feasible region. Still, it does help ensure that the concavity assumption is not violated at least in the region visited by the algorithm. Details of the test can be found in [2].

3 Illustrative Example & Computational Experience

We now illustrate the solution approach using the following model parameter values: a = 10, m = 3, $\alpha^h = \alpha^l = 0.99$, $L_l = 1$, $L_h = 0.5$, $A = 0.5$. The demand parameters are specified as: $\beta_p = 0.50$, $\theta_L = 0.25$, $\theta_p = 0.10$, $\beta_L = 0.25$. For the specified values of the parameters, the model [PQO] reduces to:

$$
\begin{aligned}
[PQO]:\max \quad & \pi(p_h, p_l, \mu) = -0.60p_h^2 - 0.60p_l^2 + 0.20p_h p_l + 11.50p_h + 11.25p_l - 0.50\mu - 58.875 \\
\text{s.t.} \quad & -0.60p_h + 0.10p_l - \mu \leq -19.2103 \\
& S_l(L_l) = P(W_l \leq L_l) \geq 0.99 \\
& -0.50p_h - 0.50p_l - \mu < -19.6250 \\
& -0.60p_h + 0.10p_l \geq -10 \\
& \; 0.10p_h - 0.60p_l \geq -9.6250 \\
& \; p_h, p_l, \mu > 0
\end{aligned}
$$

For the solution algorithm, a bound (M) on the high priority queue size needs to be specified to facilitate use of the $m$atrix geometric method. To find an appropriate value for M requires some experimentation. Computational experiments of a priority queue with a reasonable range of parameter values suggested $M = 100$ to be a good choice with little effect on the accuracy of results. For the cutting plane algorithm, we set the tolerance limit (ϵ) at 10^{-6}, and the step sizes ($dp_h, dp_l, d\mu$) for gradient estimation at 0.01.

The algorithm works by relaxing the service level constraint ($S_l(L_l) = P(W_l \leq L_l) \geq 0.99$) at iteration 0, and successively adding linear cuts of the form (15) until the stopping criteria is met. At iteration 0, the model reduces to a simple QPP, which gives the following solution: $p_h = 11.696429$, $p_l = 11.178571$ and $\mu = 13.310340$. the corresponding values for the demand rates are: $\lambda_h = 4.100000$ and $\lambda_l = 4.087500$. With these values, the service levels achieved for the two classes are: $S_h(L_h) = 0.990000$ and $S_l(L_l) = 0.957852$. Since $S_l(L_l) < 0.99$, following cut is appended to the model: $0.0203p_h + 0.0087p_l + 0.0266\mu \geq 0.7212$. Table **??** shows the results for successive iterations of the algorithm. The algorithm terminates after iteration 4 once $S_l(L_l)$ approaches α^l within the tolerance limit. The optimum solution, reported to six decimal digits, is $(p_h, p_l, \mu) = (11.836961, 11.355344, 15.399650)$ with an objective function value, $\pi = 61.326491$. Computational results, showing the number of cuts used and the time (in seconds) taken by the algorithm for a range of parameters values, are reported in Table 2. All computations are performed on a Pentium IV (3.06 GHz, 512 MB RAM) machine. The results suggest that the proposed algorithm is very efficient, taking only a few seconds.

4 Conclusion

We presented a priority queue optimization problem under service level constraints, which is challenging to solve due to the absence of analytical expression for one of its constraints, corresponding to the service level for low priority customers. We resolved the problem by iteratively generating the violated low priority service level constraint using *matrix geometric* method, which are then added to the model for re-optimization in a *cutting plane* framework. The algorithm works very efficiently, solving the problem to optimality in a few iterations. The proposed method, we hope, will allow researchers to revisit some of the seemingly intractable queuing optimization problems. The problem of product differentiation has been studied in [4] in which each customer class is served by a dedicated server. The proposed method allows us to obtain managerial insights into the effect of capacity sharing in a monopolistic market [6] and in a competitive market [7].

Table 1: Results for the Illustrative Example

Iter.	(p_h, p_l, μ)	(λ_h, λ_l)	$(S_h(L_h), S_l(L_l))$	Cut generated
0	(11.696429, 11.178571, 13.310340)	(4.100000, 4.087500)	(0.990000, 0.957852)	$0.0203p_h + 0.0087p_l + 0.0266\mu \geq 0.7212$
1	(11.796510, 11.373709, 14.378047)	(4.059465, 3.980425)	(0.994254, 0.980403)	$0.0102p_h + 0.0037p_l + 0.0125\mu \geq 0.3516$
2	(11.819103, 11.362923, 15.131496)	(4.044831, 3.989156)	(0.996087, 0.988016)	$0.0066p_h + 0.0021p_l + 0.0077\mu \geq 0.2200$
3	(11.832500, 11.357150, 15.379658)	(4.036215, 3.993960)	(0.996558, 0.989847)	$0.0057p_h + 0.0018p_l + 0.0065\mu \geq 0.1875$
4	(11.836961, 11.355344, 15.399650)	(4.033358, 3.995489)	(0.996597, 0.989999)	not needed

Table 2: Computational Results

$A \rightarrow$	0.1		0.25		0.5		0.75		1	
$L_h \downarrow$	Cuts	Time	Cuts	Time	Cuts	Time	Cuts	Time	Cuts	Time
0.1	0	0.08	0	0.09	0	0.08	0	0.05	0	0.09
0.2	0	0.08	0	0.11	0	0.08	0	0.09	0	0.08
0.3	0	0.09	0	0.08	0	0.08	0	0.09	0	0.09
0.4	1	0.8	0	0.17	0	0.16	0	0.13	0	0.16
0.5	4	3.81	4	3.16	4	2.42	4	2.39	4	2.38
0.6	5	3.17	5	3.16	5	3.14	5	3.13	5	3.14
0.7	6	4.02	6	4.03	6	4.02	6	3.95	6	3.94
0.8	6	4.48	6	4.47	6	4.39	6	4.42	6	4.41
0.9	7	5.7	7	5.7	7	5.67	7	5.63	7	5.59

References

[1] Abate, J. and W. Whitt. Asymptotics for M/G/1 low-priority waiting-time tail probabilities. *Queuing Systems*, 25, 1997, pp. 173–233.

[2] Atlason, J., M.A. Epelman and S.G. Henderson. Call center staffing with simulation and cutting plane methods. *Annals of Operations Research*, 127, 2004, pp. 333–358.

[3] Bazaraa, M.S., H.D. Sherali and C.M. Shetty. Nonlinear programming: Theory and algorithms. *John Wiley and Sons*, USA, 1993.

[4] Boyaci, T. and S. Ray. Product differentiation and capacity cost interaction in time and price sensitive markets. *Manufacturing and Service Operations Management*, 5 (1), 2003, pp. 18–36.

[5] Dewan, S. and H. Mendelson. User delay costs and internal pricing for a service facility. *Management Science*, 36, 1990, pp. 1502–1507.

[6] Jayaswal, S., E.M. Jewkes and S. Ray. Product differentiation and operations strategy in a capacitated environment *European Journal of Operational Research*, 210(3), 2011, pp. 716–728

[7] Jayaswal, S. Price and leadtime differentiation and operations strategy in a competitive market. *Working paper, Indian Institute of Management Ahmedabad*, 2013.

[8] Kelley, JE. Jr. The cutting plane method for solving convex programs. *Journal Society of Industrial Applied Math*, 8(4), December 1960, pp. 703–711.

[9] Latouche, V. and A. Ramaswami. An introduction to matrix analytic methods in stochastic modeling. *Society For Industrial And Applied Mathematics*, Philadelphia, PA, USA, 1999.

[10] Leemans, H. T. Waiting time distribution in a two-class two-server heterogeneous priority queue. *Performance Evaluation*, 43, 2001, pp. 133–150.

[11] Nelson, R. Matrix Geometric Solutions in Markov Models: A Mathematical Tutorial. *IBM T.J. Watson Research Center*, Yorktown Heights, NY, 1991.

[12] Neuts, F.M. Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach. *Dover Publications*, Mineola, NY, 1981.

[13] Palaka, K.S., D. Erlebacher and H. Kropp. Lead time setting, capacity utilization, and pricing decisions under lead time demand. *IIE Transactions*, 30, 1998, pp. 151–163.

[14] Pekgun, P., P.M. Griffin and P. Keskinocak. Coordination of marketing and production for price and leadtime decisions. *IIE Transactions*, 40(1), 2008, pp. 12–30.

[15] Ramaswami, A. and V. Lucantoni. Stationary waiting time distribution in queues with phase type service and quasi-birth-and death processes. *Communication in Statistics - Stochastic Models*, 1(2), 1985, pp. 125–136.

[16] Ray, S. and E.M. Jewkes. Customer lead time management when both demand and price are lead time sensitive. *European Journal of Operational Research*, 153, 2004, pp. 769–781.

[17] So, K.C. Price and time competition for service delivery. *Manufacturing and Service Operations Management*, Fall 2000, pp. 392–409.

[18] So, K.C. and J.S. Song. Price, delivery time guarantees and capacity selection. *European Journal of Operational Research*, 111, 1998, pp. 28–49.

[19] Stidham, S. Pricing and capacity decisions for a service facility: Stability and multiple local optima. *Management Science*, 38, 1992, pp. 1121–1139.

[20] Tsay, A.A. and N. Agrawal. Channel dynamics under price and service competition, *Manufacturing and Serice Operations Management*, 2(4), 2000, pp. 372–391.

[21] Yu, Y., S. Benjaafar and Y. Gerchak. Capacity pooling and cost sharing among independent firms in the presence of congestion. *Working paper*, 2007.