Pricing strategies and service differentiation in queues — A profit maximization perspective *

Akshay-Kumar Katta[†] Jay Sethuraman [‡]

March 2005

Abstract

We consider the problem of pricing and scheduling the customers arriving at a service facility, with the objective of maximizing the profits of the facility, when the value of service and time-sensitivity of a customer are his private information. First we consider the 'discrete types' problem where each customer belongs to one of N types, *type* i being characterized by its value for service R_i and cost of waiting per unit time c_i. For the special case when $\frac{R_i}{c_i}$ is decreasing in c_i, we characterize the structure of the optimal pricing-scheduling policy and design a polynomial-time algorithm to find it. We then analyze the same problem under the additional restriction of at most m different levels of service, characterize the optimal pricing-scheduling policy, and provide an efficient way to find it. Finally, we consider the case where the types of customers form a continuum and the customers have a generalized delay cost structure. Using the insights from the discrete types case, we characterize the conditions under which the optimal mechanism schedules the customers according to the $c\mu$ rule.

[*] This research was supported by an NSF grant DMI-0093981

[†] Department of Industrial Engineering and Operations Research, Columbia University, New York, NY; email: ark2001@columbia.edu

[‡] Department of Industrial Engineering and Operations Research, Columbia University, New York, NY; email: jay@ieor.columbia.edu

1 Introduction

Consider a service facility serving a heterogenous pool of customers with private information about their value for service and time sensitivity. What pricing-scheduling strategy should the service facility follow to maximize its profits? This question is important in situations where delay is a key component in the customer's perception of service quality and desirability, and together with price, completely determines whether a customer will do business with the firm. This situation applies to many service and manufacturing systems, telecommunications, transportation systems like railways, and postal services. A common strategy used by such providers facing heterogenous customers is to offer many classes of service differentiated by price, and letting the customers choose their service class for themselves. For example, the postal system offers priority and regular mail and a range of other services; Railways offer express and regular freight services; manufacturing firms routinely charge a price depending on the delivery date, etc. Priority pricing improve revenues by segmenting the market and extracting greater profit from those customers who are willing to pay more for faster service (i.e. customers with greater time sensitivity). If the service provider has perfect information regarding each of his customers, then he can price the customers so that the net benefit from obtaining service is zero for all of them. Thus, the problem of determining the optimal pricing-scheduling policy is equivalent to solving a related queuing control problem. However, the problem of determining the optimal policy becomes much more difficult if the service provider only has general information regarding his customer base, but cannot tell his customers apart. In this scenario, a customer will act in a self-interested way and choose the service class most beneficial to him; the service provider, therefore, has to take this behavior into account when determining his policy. Our main goal in this paper is to understand how the service provider should segment the customers and price them so as to maximize his profit. We analyze this problem by modeling the service facility as a queueing system and by making some simplifying assumptions on the problem parameters.

Our work was inspired by the recent papers by Afeche [1] and Afeche and Mendelson [2]. Afeche [1] considers the problem of designing a profit maximizing pricing-scheduling policy for a service facility that serves *two* types of customers. All the customers within a type have the same waiting cost, processing requirement, and a value for service based on a probability distribution. In this setting Afeche shows that the $c\mu$ priority rule need not be optimal and that the optimal policy might involve 'idling' the server even when there are some customers in the system. By strategically delaying the less impatient customers, the more impatient customers can be made to pay more money (at the expense of losing some of the lower-end customers), leading to an increase in revenue under certain conditions. In fact, when the service requirements of the two types are different, serving the customers in the reverse $c\mu$ rule (or) appropriately randomizing priority assignments might be optimal. These results show that the delay-cost might not be minimized under the optimal profit maximizing-policy, drawing a sharp distinction between the objectives of maximizing profit and maximizing overall system utility. Moreover, Afeche [1] outlines a stepwise solution methodology to identify the profit maximizing mechanisms under his setting; our analysis in this paper draws on this methodology.

Afeche and Mendelson [2] propose a generalized delay cost structure to capture the dependence between the delay cost of a customer and his service valuation. Under this cost structure, the utility derived by a customer who finishes service t time units after entering the system is given by $u(t) = r.D(t) - C(t) - p$; where r is his value of service, p is the price paid for service, $C(t)$ is the cost of waiting (increasing in t) and $D(t)$ is the delay discount function (decreasing in t). In their model, the customers have a generalized delay cost structure, and their service valuation is drawn from a probability distribution Φ, the actual service valuation of a customer is his private information. Let $V'(\lambda) = \overline{\Phi}^{-1}(\frac{\lambda}{\Lambda})$, where Λ is the potential arrival rate into the system. For this setting, assuming that $\lambda V'(\lambda)$ is strictly concave, they show that priority auction mechanisms, which serve the customers according to the $c\mu$ rule, perform better than list pricing under both social optimization as well as profit maximization criteria. However, they do not focus on the question of finding the optimal pricing-scheduling mechanism; we address this issue in our paper.

Content and Contributions. We model the service facility as an M/M/1 queue with service preemptions allowed. We assume that arriving customers cannot observe the state of the queue. However, they know all the average system statistics and make their decision solely based on these statistics. First, we analyze the problem where each customer belongs to one of N types. Each type is characterized by its value for service R_i and cost of waiting per unit time c_i; the potential arrival rate into type i is Λ_i. This scenario might arise in real life situations where the potential customer information is based on market research and hence is segmented into convenient types. For the special case when $\frac{R_i}{c_i}$ is decreasing in c_i, we characterize the structure of an optimal pricing-scheduling policy and design an efficient algorithm to find one. It turns out that in the optimal policy, the customer types may not be scheduled according to the $c\mu$ rule; we may have to "pool" some customer types together and treat them as if they belong to a single type for scheduling purposes. Note that this result is different from the result in Afeche [1] where the reverse $c\mu$ order (or) appropriately randomizing priority assignments might be optimal: The suboptimality of the $c\mu$ rule observed there is because customer types have *different* service requirements, whereas in our setting all the customers have the *same* service requirement.

Next, we consider the case where the service provider is restricted to use at most m different service levels. This arises in situations where it is expensive to have too many service classes. In many firms the number of service levels to offer is a strategic decision made by taking into account the operational and logistical difficulties in implementing it. (For example, UPS only has 5 or 6 kinds of services levels.) For this setting, we again characterize the optimal optimal pricing-scheduling policy and provide an efficient way of finding one.

Then, we consider the case where customer value for service is drawn from a continuous distribution Φ. Let $V'(\lambda) = \overline{\Phi}^{-1}(\frac{\lambda}{\Lambda})$, where Λ is the potential arrival rate into the system. We assume that the customers have the generalized delay cost structure proposed in [2] and that the waiting cost and the deflation of the value due to delay are linear in time. The generalized delay cost structure can be viewed as a continuous version of our 'discrete model' assumption that the ratio $\frac{R_i}{c_i}$ be decreasing in c_i. Our main goal here is to understand the structure of the optimal mechanism in this 'continuum' customer types setting. Using the insights gained from solving the N customer types case, we show that the optimal profit-maximizing mechanism need not always schedule the customers according to the $c\mu$ rule. We construct an example where 'pooling' some customer types together and offering them the same service level leads to a higher revenue. Furthermore, we characterize the conditions under which the optimal mechanism schedules the customers according to the $c\mu$ rule. We show that the priority auction, which serves the customers according to the $c\mu$ rule, is optimal when the function $\lambda V'(\lambda)$ is concave. In particular, our result implies that, for the setting in Afeche and Mendelson [2], the priority auction they analyze is in fact the profit-maximizing mechanism.

Finally, for the continuous case, we investigate the tradeoff between restricting the number of different service levels and the loss in optimality by conducting numerical experiments. We observe the change in the pricing and segmentation of the customers as the number of service classes goes up. We end this section with a review of other related work.

Related Work. There is a vast literature analyzing the design and control of queuing systems, where the system manager has full information regarding each of his customers and determines his policies based on this information. Stidham [36] provides a comprehensive recent survey of this field.

Kleinrock [21] first introduced priority pricing in queues in terms of 'bribing' for priority and studied the tradeoff between the delay cost and magnitude of the bribe under an overall system objective. But in his model the customers are assumed to be non-strategic. Since then numerous papers have studied the pricing and scheduling of queuing systems with strategic customers, Hassin and Haviv [20] provide a comprehensive review of this literature. Naor [30] considers an observable FIFO M/M/1 queue and shows that individual optimization does not lead to social optimality and that social optimality can be achieved by levying a fixed entry fee. Furthermore, he shows that the profit maximizing entry fee is greater than the socially optimal entry fee. Balachandran [3] considers an M/M/1 queue model where identical customers observe the length of the queue and choose from discrete, infinite set of possible payments; customers are given priority based on their payment. Conditions under which it is an equilibrium to buy the lowest priority that ensures being placed at the head of the queue are characterized. Edelson and Hilderbrand [12] consider an unobservable FIFO M/M/1 queue and show that the profit maximizing fee is same as the socially optimal one.

Ghanem [14] considers an unobservable M/G/1 system where the time value of the customers is a continuous random variable and the server is restricted to having m priority classes. He derives the structure of the optimal pricing-scheduling mechanism and calculates explicit solutions for $m = 2$ and uniform or exponential time-value distributions. Dolan [11] considers an observable queue model in which the customers have different time value, the service times are deterministic, and all the customers enter the queue for service. He proposes a mechanism that induces users to reveal their true delay costs, thus leading to an efficient ordering. Dewan and Mendelson [10] consider a queuing problem with homogeneous customers and nonlinear waiting costs. They show that, to achieve social optimality, the price charged to a customer should be proportional to the cost inflicted on all other users due to the increase in their overall delay. Mendelson and Whang [28] extend this study to an M/M/1 non preemptive priority queue with multiple customer classes and with linear waiting costs. Bradford [5] extends their results to a setting where, instead of assigning priorities, the system manager controls the processing speed of the various customers by sending them to different machines (with varying service rates). Van Mieghem [37] studies the social optimization problem when the customer waiting costs are convex and shows that the generalized $c\mu$ rule can be implemented. Ha [17] considers a system in which each customer chooses his service rate and derives IC and socially optimal prices.

A limited line of research explores the role of auctions in a queuing setting (as opposed to the centralized pricing explored in most of the literature). Glazer and Hassin [15] and Liu [23] consider the model where customers have heterogenous marginal costs of delay. They show that auctions lead to an equilibrium where higher marginal cost leads to a higher bid, thus leading to the implementation of the $c\mu$ *rule*. Hassin [19] shows that an auction mechanism always leads to a socially optimal outcome in the short run in any system with exponential service rate and preemptive service. He also shows that a profit maximizer may choose a service speed which is slower than the socially optimal speed. Afeche and Mendelson [2] extend the auction model by allowing for a minimum price.

Recently, the issue of revenue maximization in queuing settings has been receiving increasing attention. Rao and Petersen [34] consider a generic congestion model with a fixed number of priority classes, they assume that the customers belong to one of n types and that there is a single decision maker for all the customers of a particular type. Decision makers are free to send any amount of traffic to a service class, but the service provider knows the type of each customer and can charge a price based on this information; they provide prices that maximize the revenue in this setting. Lederer and Li [22] study the price-delay equilibrium under perfect competition, they *assume* that each firm uses the $c\mu$ rule for scheduling its customers and show the existence of a competitive equilibrium. Plambeck [32] considers an M/M/1 queue with two customer types who have equal service times but different delay costs. Using diffusion approximations, she studies the joint problem of dynamic lead-time quotation, static pricing and capacity sizing for this queue. Maglaras [24] studies the problem of maximizing the revenue of a make-to-order firm that offers multiple products to price sensitive users; he assumes that the demand rate for each product is a function of the prices for all the products and that the firm incurs quadratic holding costs. Maglaras and Zeevi [25] consider the revenue maximization problem in a service system in which two classes of service are offered: the first one is a *Guaranteed* service in which customers are guaranteed a certain processing capacity; and the second class is a Best-effort service in which the residual capacity is shared among the customers who opt for this class. Caldentey and Wein [6] study the problem of revenue maximization for a make to stock queue that serves a long term market and spot market demand. Gupta and Wang [16] study the revenue management in a system where the long term market is served on a make-to-order basis and the spot market is served either on a make-to-order or on a make-to-stock basis. Printezis and Burnetas [33] consider a two customer types model and study the profit maximizing pricing policies when the service provider is obliged to offer the same level of service to all customers.

2 Discrete customer types

In this section we discuss the problem of profit-maximization in a queueing setting when the customers belong to one of N types. A type i customer has a value of service R_i and incurs a cost c_i for each unit of time spent in the system. The customer types are indexed in decreasing order of the waiting costs, i.e., $c_1 > c_2 > \ldots > c_N$. Type i customers arrive to the service facility following a Poisson process of rate Λ_i; the arrival process of any customer type is independent of the arrival process of the remaining types and the service process. The service requirement of any customer is independent of his type. The service facility is modelled as a single server queue with an exponential service rate μ and with service preemptions allowed. We assume that the individual customers are infinitesimal, so the decision of a single customer has a negligible effect on the system statistics. We also assume that there is no cost to the firm of serving a customer. We analyze this system under the simplifying assumption of $\frac{R_1}{c_1} > \frac{R_2}{c_2} > \ldots > \frac{R_N}{c_N}$. This assumption makes sense in certain economic settings, for instance when the perceived waiting cost is mainly a consequence of the discounting of the value of service due to delay. The rest of this section is organized as follows. First, we show how the profit-maximization problem can be converted to an optimal scheduling problem. Then, we design an efficient algorithm to solve this scheduling problem, assuming a fixed arrival rate vector. Finally, we show how to determine the optimal arrival rate vector efficiently. Before going further, we define the class of feasible pricing-scheduling policies and show that we can restrict our attention to a subset of these called *Incentive Compatible* (IC) policies.

Pricing-Scheduling policies. To maximize revenue, what pricing-scheduling policy should the service provider follow? Remember that the service provider does not know the type of a particular customer, unless the customer himself reveals this information. Also, since the service requirement of all the customers is (statistically) identical, there is no loss of generality in restricting attention to policies that are independent of the actual processing time realized by the customers[1]. Hence it is enough to consider policies where the service provider offers a menu of *options*, denote the set of all possible menus by T. A customer who chooses option $i \in t$, where $t \in T$ is the menu being offered, pays a price P_i. The service provider also announces the scheduling policy to be

[1] Because even if we charge an amount $f(x)$ when x is the service time, the customers would behave as if they were paying $E(f(x))$.

employed in serving the customers choosing among the options in T. In this paper we only consider scheduling policies that belong to the set of *admissible* scheduling policies (denoted by A) defined as follows:

Admissible scheduling policies: Let $\overline{A}$ denote scheduling policies that are (a) stationary; (b) non-idling; (c) do not affect the arrival process or the service requirements; and (d) non-anticipating, i.e. they only make use of the past history and the current state of the system (but do not use knowledge of actual remaining service times). A scheduling policy r is *admissible* if and only if there exists an $r' \in \overline{A}$ such that r differs from r' only in that it delays completed class i jobs on average by $d_i \geq 0$ units of time. Denote the set of *admissible policies* by A.

The properties $(a) - (d)$ are intuitive and one would expect any reasonable scheduling policy to satisfy them. Hence in the queuing literature, attention is typically restricted to the scheduling policies in $\overline{A}$. However, as shown by Afeche [1], the optimal scheduling policy may be forced to idle some of the customers to achieve incentive compatibility. Hence we expand our attention to the policies in A. We will denote any pricing-scheduling policy by (t, r), where $t \in T$ and $r \in A$. The prices of the different options are denoted by the vector $(P_1^t, P_2^t, \ldots, P_{|t|}^t)$; the Nash-equilibrium arrival rates of the various customers types by $\lambda^r = (\lambda_1^r, \ldots, \lambda_N^r)$ and the induced expected waiting time vector of the customers choosing among the options in t by $W^r = (W_1^r, \ldots, W_{|t|}^r)$. Note that for all the scheduling policies in A, the expected steady state delays of all the options in t are well defined. This, coupled with the infinitesmal nature of the individual customer, ensures the existence of a Nash-equilibrium. Wherever it is clear which pricing-scheduling policy is being considered, we will drop the superscripts t and r. We now define the class of Incentive Compatible (IC) policies.

IC policies: A policy (t, r) is Incentive Compatible (IC) if there are exactly N options in t, and customers of type i always choose option i.

We can think of option i in an IC policy as being tailor-made for type i customers. Due to a fundamental result in economics, known as the *Revelation principle* (see Myerson [29]), we can restrict our attention to the class of IC policies while searching for an optimal pricing-scheduling policy. (Harris and Townsend [18] prove the revelation principle in a rather general setting.) The revelation principle holds in our setting because for any given mechanism, it is possible to construct an IC mechanism that performs equally well. In the rest of this paper, we shall restrict our attention to IC policies.

Consider any IC policy (t, r). Individual rationality (IR) implies that, in equilibrium,

$$\begin{aligned} P_i^t &= R_i - c_i W_i^t, \quad \text{if} \quad 0 < \lambda_i < \Lambda_i, \\ P_i^t &\leq R_i - c_i W_i^t, \quad \text{if} \quad \lambda_i = \Lambda_i, \\ P_i^t &\geq R_i - c_i W_i^t, \quad \text{if} \quad \lambda_i = 0. \end{aligned} \tag{1}$$

Note that, if type i customers enter the system partially, each of them will have a zero surplus (as $P_i = R_i - c_i W_i \Rightarrow$ net benefit $= R_i - c_i W_i - P_i = 0$). Also, if type i customers do not enter the system, then they do not incur any costs and get no benefits. Hence, they will also have a zero surplus. On the other hand, if type i customers enter the system in full, then they *may* have a positive surplus. We now prove the intuitive result that in any IC policy, the waiting time of a customer with a higher waiting cost will be at most the waiting time of a customer with a lower waiting cost.

Lemma 1 *Suppose that all the customer types in* $\{1, 2, \ldots, N\}$ *enter the system (at least partially) in the equilibrium of the IC policy* (t, r)*. Then we will have* $W_1 \leq W_2 \leq \ldots \leq W_N$*.*

Proof. We prove the lemma by showing that if both type i and $i-1$ enter, then we will have $W_{i-1} \leq W_i$. Incentive compatibility requires that a type i customer does not benefit from pretending to be a type $i-1$ customer, that is, $P_i + c_i W_i \leq P_{i-1} + c_i W_{i-1}$. Incentive compatibility also requires that a type $i-1$ customer does not benefit from pretending to be a type i customer, and so $P_{i-1} + c_{i-1} W_{i-1} \leq P_i + c_{i-1} W_i$. Adding these inequalities, we get

$$c_i W_i + c_{i-1} W_{i-1} \leq c_i W_{i-1} + c_{i-1} W_i \implies W_{i-1}(c_{i-1} - c_i) \leq W_i(c_{i-1} - c_i) \implies W_{i-1} \leq W_i,$$

where the last implication is because $c_{i-1} > c_i$. ■

Checking incentive compatibility. Consider any pricing-scheduling policy (not necessarily IC) with N options. Let the prices charged be $(P_1, \ldots, P_N)$. Let $(W_1, \ldots, W_N)$ be the waiting time vector when all the customers report their type truthfully (i.e., type i customers choose option i), and suppose $W_1 \leq W_2 \leq \ldots \; W_N$ (this condition is necessary due to Lemma 1). Is the given policy incentive compatible? To answer this, we would have to check the following IC conditions

$$P_k + c_k W_k \quad \leq \quad P_l + c_k W_l \Rightarrow P_k - P_l \leq c_k(W_l - W_k), \tag{2}$$

$$P_l + c_l W_l \quad \leq \quad P_k + c_l W_k \Rightarrow P_k - P_l \geq c_l(W_l - W_k), \tag{3}$$

for all pairs of customer types $\{(k, l) : k \neq l\}$. In the special case we consider, we show next that it is enough to check these conditions for "adjacent" customer types.

Lemma 2 *Consider a policy* (t, r)*; let* $(W_1, \ldots, W_N)$ *be the equilibrium average waiting times when the service provider knows the customer types and assigns a type* i *customer to option* i*. Suppose* $W_1 \leq W_2 \leq \ldots \leq W_N$*. Then the IC constraints are satisfied if and only if*

$$P_{i+1} - P_i \quad \leq \quad c_{i+1}(W_i - W_{i+1}), \tag{4}$$

and

$$P_{i+1} - P_i \quad \geq \quad c_i(W_i - W_{i+1}), \tag{5}$$

for all $i = 1, 2, \ldots, N-1$*.*

Proof. The only if part follows from the definition of incentive compatibility, so we only need to prove the if part. Suppose that (4) and (5) are satisfied for all $i = 1, 2, \ldots, N-1$. Let $k > l$ be two customer types. Then we have

$$\begin{aligned}
P_k - P_l &= (P_k - P_{k-1}) + (P_{k-1} - P_{k-2}) + \ldots + (P_{l+1} - P_l) \\
&\geq c_{k-1}(W_{k-1} - W_k) + c_{k-2}(W_{k-2} - W_{k-1}) + \ldots + c_l(W_l - W_{l+1}) \quad [\text{ From (5) }] \\
&\geq c_l(W_{k-1} - W_k) + c_l(W_{k-2} - W_{k-1}) + \ldots + c_l(W_l - W_{l+1}) \\
&\quad [\text{ because } W_{k-1} - W_k \leq 0 \ \text{ and } \ c_{k-1} < c_l] \\
&= c_l[W_l - W_k]
\end{aligned}$$

Also, we have

$$\begin{aligned}
P_k - P_l &= (P_k - P_{k-1}) + (P_{k-1} - P_{k-2}) + \ldots + (P_{l+1} - P_l) \\
&\leq c_k(W_{k-1} - W_k) + c_{k-1}(W_{k-2} - W_{k-1}) + \ldots + c_{l+1}(W_l - W_{l+1}) \quad [\text{ From (4) }] \\
&\leq c_k(W_{k-1} - W_k) + c_k(W_{k-2} - W_{k-1}) + \ldots + c_k(W_l - W_{l+1}) \\
&\quad [\text{ because } W_l - W_{l+1} \leq 0 \ \text{ and } \ c_{l+1} > c_k] \\
&= c_k[W_l - W_k]
\end{aligned}$$

Therefore both the IC conditions are satisfied and the lemma is proved. ■

Lemma 2 shows how we can effectively handle incentive compatibility: It provides us with the additional conditions under which we may freely assume that the service provider knows the customer types. Specifically, as long as $W_1 \leq \ldots \leq W_N$, and (4) and (5) are satisfied in the ensuing equilibrium, we can assume that the service provider knows the customer types while designing the pricing-scheduling policy. Note that we still need to ensure that the individual rationality constraints are satisfied. We shall illustrate this more clearly when we formulate the profit maximization problem as a mathematical programming problem in the next subsection.

2.1 Solution structure

Our main goal in this section is to prove that if type k customers are served in the optimal solution then customer types $1, 2, \ldots, k-1$ are served in full, and to show how the problem of finding an optimal pricing-scheduling policy can be converted into one of finding an optimal scheduling policy. First we prove the following result for the arrival rates of the various types of customers.

Lemma 3 *In any IC policy, if $P_k \geq 0$ and $\lambda_k > 0$ then $\lambda_i = \Lambda_i$ for all $i < k$.*

Proof. Consider an IC policy with $P_k \geq 0$ and $\lambda_k > 0$. In equilibrium we will have $P_k \leq R_k - c_k W_k$ (from (1)). This, together with $P_k \geq 0$, implies $W_k \leq \frac{R_k}{c_k}$.

Fix any $i < k$. Now, if a type i customer reports his type as k, then his utility will be $u'_i = R_i - c_i W_k - P_k$. But

$$W_k \leq \frac{R_k}{c_k} < \frac{R_i - R_k}{c_i - c_k} \implies W_k < \frac{R_i - R_k}{c_i - c_k} \implies R_i - c_i W_k > R_k - c_k W_k \geq P_k.$$

Therefore we will have $u_i' > 0$. Since the policy is IC, it follows that we will have $u_i = R_i - c_i W_i - P_i \geq u_i' > 0$; hence a type i customer will enter in full (from (1)). ■

In view of Lemma 3, if we prove that there is an optimal IC policy in which the highest indexed customer type entering the system pays a non-negative price, then it follows that all the customer types of lower index will enter in full. We do this below.

Lemma 4 *Suppose there are N customer types, and let n be the highest indexed customer type that enters service in an optimal IC policy. Then, we will have $P_n \geq 0$. Consequently, we will have $P_i \geq 0$ for all customer types i that enter the system in the optimal solution (from Lemma 1 and equation (3)).*

Proof. Suppose that there is an optimal IC policy (t, r) such that $P_n < 0$ and $\lambda_n > 0$. Let the equilibrium of this policy be $[(P_1, P_2, \ldots, P_n); (\lambda_1, \lambda_2, \ldots, \lambda_n)]$ and let the corresponding equilibrium waiting times be $(W_1, \ldots, W_n)$. The profit of this system is given by $Z^N = \lambda_1 P_1 + \lambda_2 P_2 + \ldots + \lambda_n P_n$. Note that at least one of the customer types $\{1, 2, \ldots, n-1\}$ has to enter the system by paying a non-negative price; otherwise we will have a negative profit (contradicting the optimality of this policy). Let s be the highest indexed customer such that $P_s \geq 0$ and $\lambda_s > 0$. Then, as $P_s \leq R_s - c_s W_s$ (from equation(1)), we will have $W_s \leq \frac{R_s}{c_s}$. Due to Lemma 1, we will have $W_i \leq W_s \quad \forall\ i \leq s$. Also, due to Lemma 3, we will have $\lambda_i = \Lambda_i \quad \forall i < s$. We shall now construct an IC policy in which all the customer types entering the system pay a non-negative amount, and whose revenue is greater than the revenue generated by the policy (t, r).

First, we claim that there is a scheduling rule under which the equilibrium waiting times resulting from the arrival rate vector $(\Lambda_1, \ldots, \Lambda_{s-1}, \lambda_s, 0, \ldots, 0)$ is $(W_1, \ldots, W_s)$. This scheduling rule is the same as the scheduling rule r except for the following difference: the server itself generates imaginary customers of types $\{s+1, \ldots, n\}$ at the respective rates $\{\lambda_{s+1}, \ldots, \lambda_n\}$ and "schedules" them according to the rule r; thus "serving" a customer of type i ($i > s$) amounts to idling the server. Denote this scheduling policy by r'. Now consider the pricing scheme t' defined as follows:

$$P'_{s+1} = P'_{s+2} = \ldots = P'_N > R_1; \qquad P'_s = R_s - c_s W_s; \qquad P'_i = P_i + \delta, \quad \forall\ i < s,$$

where $\delta = P'_s - P_s$. As $0 \leq P_s \leq R_s - c_s W_s$, it follows that $\delta \geq 0$ and $P'_s \geq 0$. Note that the agents of types $\{s+1, \ldots, N\}$ will never enter the system by reporting their true types as this will always result in a negative utility. We prove that the policy (t', r') is incentive compatible and individually rational, with an arrival rate vector $(\Lambda_1, \ldots, \Lambda_{s-1}, \lambda_s)$ in the Nash equilibrium.

Consider the arrival rate vector $(\Lambda_1, \ldots, \Lambda_{s-1}, \lambda_s)$; as already noted the waiting times vector associated with this is $(W_1, \ldots, W_s)$. Note that we have $P'_i - P'_j = P_i - P_j \quad \forall\ \ i, j \leq s$. Hence the IC conditions (2) and (3) are satisfied for all such i, j pairs (as these conditions were satisfied in the policy (t, r)). Also since it never pays to report as a type $k > s$ customer, we can conclude that IC constraints are satisfied for customers of types $1, \ldots, s$. Next we prove that this arrival rate vector satisfies IR conditions for the types $j \leq s$. This is obviously true for type s (as $P'_s = R_s - c_s W_s$). For any $j < s$, we have

$$P'_j - P'_s \leq c_j(W_s - W_j) \quad [\text{ equation (2) }] \;\Rightarrow P'_j \leq R_s - c_s W_s + c_j W_s - c_j W_j.$$

Also, we have

$$W_s \leq \frac{R_s}{c_s} < \frac{R_j - R_s}{c_j - c_s} \implies W_s \leq \frac{R_j - R_s}{c_j - c_s} \implies R_s - c_s W_s + c_j W_s \leq R_j.$$

It follows that $P'_j \leq R_j - c_j W_j$ and hence the IR constraint is satisfied for type j customers.

To conclude that the equilibrium arrival rate vector is $(\Lambda_1, \ldots, \Lambda_{s-1}, \lambda_s)$, we only need to show that customers of type $k > s$ never enter the system. Note that

$$W_s \leq \frac{R_s}{c_s} < \frac{R_s - R_k}{c_s - c_k} \implies W_s < \frac{R_s - R_k}{c_s - c_k} \implies R_k < R_s - c_s W_s + c_k W_s = P'_s + c_k W_s.$$

Hence, a type k customer will not enter the system by reporting type s. Now consider any $j < s$. From the IC constraint for type s (so that type s does not pretend to be a type j customer) it follows that $R_s \leq P'_j + c_s W_j \implies R_s - c_s W_j \leq P'_j$. Therefore, we have

$$W_j \leq W_s < \frac{R_s - R_k}{c_s - c_k} \implies W_j < \frac{R_s - R_k}{c_s - c_k} \implies R_k < R_s - c_s W_j + c_k W_j \leq P'_j + c_k W_j$$

Hence, a type k customer will not enter the system by reporting his type as j. As it never pays to enter by revealing their true type, we can conclude that customers of type $k > s$ never enter the system.

Thus $(\Lambda_1, \ldots, \Lambda_{s-1}, \lambda_s)$ is the equilibrium arrival rate into the system under the policy (t', r'). The revenue generated by this policy is given by

$$\begin{aligned} Z' &= \Lambda_1 P'_1 + \ldots + \Lambda_{s-1} P'_{s-1} + \lambda_s P'_s \\ &\geq \Lambda_1 P_1 + \ldots + \Lambda_{s-1} P_{s-1} + \lambda_s P_s \quad [\text{ as } P'_j \geq P_j \quad \forall \;\; j \leq s \;] \\ &> \Lambda_1 P_1 + \ldots + \Lambda_{s-1} P_{s-1} + \lambda_s P_s + \lambda_{s+1} P_{s+1} + \ldots + \lambda_n P_n \end{aligned}$$

The last inequality follows because for all $s < k < n$ we have either $P_k < 0$ or $\lambda_k = 0$ and we also have $P_n < 0, \lambda_n > 0$. But this implies that (t', r') generates more revenue that (t, r), contradicting the optimality of (t, r). Thus we cannot have $P_n < 0$ and $\lambda_n > 0$; and the lemma is proved. ■

Price Structure. Suppose that, in an optimal IC solution, type n customers enter into the system at the rate λ_n; and that they are the highest indexed customer type entering the system. From Lemmas 3 and 4 we know that the customers of type $(1, \ldots, n-1)$ enter the system in full. Hence the optimal arrival rate vector is of the form $(\Lambda_1, \ldots, \Lambda_{n-1}, \lambda_n)$ (where $\lambda_n > 0$). From Lemma 4 we have $P_n \geq 0$ and from equation (1) we have $P_n \leq R_n - c_n W_n$. A necessary condition for both these equations to be valid is $W_n \leq R_n/c_n$. Thus adding this constraint does not affect the optimal solution. The problem of finding a profit maximizing IC policy can be expressed as :

$$\begin{aligned} \max \quad & \Lambda_1 P_1 + \ldots + \Lambda_{n-1} P_{n-1} + \lambda_n P_n \\ s.t. \quad & W_1 \leq W_2 \leq \ldots \leq W_n \\ & P_i - P_{i+1} \leq c_i(W_{i+1} - W_i) \quad \forall i = 1, 2, \ldots, n-1 \\ & P_i - P_{i+1} \geq c_{i+1}(W_{i+1} - W_i) \quad \forall i = 1, 2, \ldots, n-1 \\ & P_i \leq R_i - c_i W_i \quad \forall i = 1, 2, \ldots, n \\ & W_n \leq \frac{R_n}{c_n}, \quad (W_1, \ldots, W_n) \in A \end{aligned} \tag{6}$$

The first three constraints ensure that the policy is IC. The fourth constraint is the individual rationality constraint. The last constraint ensures that the scheduling policies considered are admissible. Now suppose that we fix the waiting time vector and that it satisfies the equation

$$W_1 \leq W_2 \leq \ldots \leq W_n \leq \frac{R_n}{c_n} \tag{7}$$

Consider the following prices:

$$P_n = R_n - c_n W_n; \quad P_{n-1} = P_n + c_{n-1} W_n - c_{n-1} W_{n-1}; \quad \ldots; \quad P_1 = P_2 + c_1 W_2 - c_1 W_1 \tag{8}$$

Note that here P_n is as high as individual rationality for type n will allow it to be (there is no other upper bound). Fixing this P_n, the prices $(P_{n-1}, \ldots, P_1)$ are as high as equation (5) allows them to be. Thus, any set of feasible prices P' will have the property $P'_i \leq P_i \ \ \forall i = 1, \ldots, n$. Therefore if the prices P are feasible, then they have to be the optimal prices. Below, we argue that these prices are indeed feasible by showing that they are incentive compatible and individually rational.

Lemma 5 *If $W_1 \leq \ldots \leq W_n \leq \frac{R_n}{c_n}$, then the prices in equation (8) are Incentive Compatible and Individually Rational (IR).*

Proof. As $P_{i+1} - P_i = c_i[W_i - W_{i+1}]$, equations (5) are satisfied. Also, $c_i[W_i - W_{i+1}] \leq c_{i+1}[W_i - W_{i+1}]$ (as $c_i > c_{i+1}$ and $W_i - W_{i+1} \leq 0$). Therefore we will have $P_{i+1} - P_i \leq c_{i+1}[W_i - W_{i+1}]$. Hence equations (4) are also satisfied and the prices in (8) are incentive compatible.

We will prove the IR part by induction. The prices are IR for type n as $P_n = R_n - c_n W_n$. Assume that the prices are IR for all types $\{n, n-1, \ldots, i+1\}$; thus we will have $P_{i+1} \leq R_{i+1} - c_{i+1} W_{i+1}$. Note that we will have

$$P_i = P_{i+1} + c_i W_{i+1} - c_i W_i \leq R_{i+1} - c_{i+1} W_{i+1} + c_i W_{i+1} - c_i W_i$$

Also

$$W_{i+1} \leq W_n \leq \frac{R_n}{c_n} < \frac{R_{i+1}}{c_{i+1}} < \frac{R_i - R_{i+1}}{c_i - c_{i+1}} \implies W_{i+1} < \frac{R_i - R_{i+1}}{c_i - c_{i+1}} \implies R_{i+1} - c_{i+1} W_{i+1} + c_i W_{i+1} \leq R_i$$

Therefore, it follows that $P_i \leq R_i - c_i W_i$. Hence the prices are IR for type i; and the lemma is proved by induction. ■

Therefore the prices of equation (8) are indeed the optimal prices. Notice that in this price structure, we will have that the net utility of a type n customer is 0. Thus, we have proved the following theorem.

Theorem 6 *If type k customers are served in an optimal IC solution then customer of the types $1, 2, \ldots, k-1$ are served in full; moreover the highest indexed customers entering the system have 0 net utility. The waiting times vector in this optimal solution satisfies equation (7), and given such a waiting time vector, the optimal prices are given by equation (8).*

Substituting for the prices from equation (8), the profit maximization problem (6) can be rewritten as :

$$\begin{array}{lll} \text{Max} & R_n[\Lambda_1 + \ldots + \Lambda_{n-1} + \lambda_n] - \sum_{i=1}^{n} W_i \Big[c_i(\lambda_1 + \ldots + \lambda_i) - c_{i-1}(\lambda_1 + \ldots + \lambda_{i-1}) \Big] & \\ \text{subject to} & W_1 \leq \ldots \leq W_n \leq \dfrac{R_n}{c_n}; \qquad (W_1, \ldots, W_n) \in A & (9) \\ & \lambda_i = \Lambda_i \ \ \forall \ i \in \{1, \ldots, n-1\}; \qquad \lambda_n \leq \Lambda_n & \end{array}$$

Note that the only variables in this formulation are $W_1, \ldots, W_n$. Thus, given the arrival rate vector in the optimal solution, we have reduced the profit maximization problem to an optimal scheduling problem. In the next section, we give an efficient algorithm to solve this optimal scheduling problem.

2.2 Optimal customer segmentation when arrival rate is fixed

Suppose that the arrival rate in the optimal IC solution is $(\Lambda_1, \ldots, \lambda_n)$. We assume that the arrival rate vector is such that the feasible region of the problem (9) is non-empty (otherwise it will never be possible to achieve this vector using an IC policy). The $c\mu$ rule need not always be optimal for the resulting scheduling problem, and it might be beneficial to pool some of the customer types together. Our objective here is to determine the segmentation of the customer types in the optimal solution. Specifically we want to arrange the customer types into various groups, with the property that all the customers within a group are treated equivalently for scheduling purposes in the optimal solution. We use the idea from the $c\mu$ rule in order to come up with a *customer segmentation.* For now, we will restrict ourselves to the scheduling rules in $\overline{A}$ (i.e. work conserving policies). As we show later (in Remark 3), there is no loss in optimality for the original pricing-scheduling problem due to this assumption.

Let M_i be the coefficient of W_i in the objective function of (9). Initially, we treat each customer type as a separate group. Suppose k is a group such that $\frac{M_k}{\Lambda_k} > \frac{M_{k-1}}{\Lambda_{k-1}}$. Then, in the absence of any additional constraints, we would have given preemptive priority to group k over group $k-1$

in the optimal solution. But, we have to contend with the constraints $W_1 \leq \ldots \leq W_{k-1} \leq W_k$. Therefore we can only go as far as having $W_{k-1} = W_k$. To ensure that this always happens we merge the groups $k-1$ and k into a single larger group with arrival rate $\Lambda_{k-1} + \Lambda_k$, endow the new group with the characteristics of group k and calculate its coefficient. Thus we have a new set of groups (with cardinality reduced by 1). We repeat the above procedure for the new set until we cannot find a group k (from the current existing groups) with the property that $\frac{M_k}{\Lambda_k} > \frac{M_{k-1}}{\Lambda_{k-1}}$. We formally describe this procedure in *Segmentation Algorithm*.

Segmentation Algorithm

1. Let S $= \{1, \ldots, N\}$. Let c_i and R_i denote the waiting cost and benefit of *type* $i \in S$; and let M_i denote the coefficient of W_i in objective function of (9). Let $S' = S$, $s'(i) = \{i\}$ and $M'_i = M_i$ for all $i \in S$. Also let $\Lambda'_k = \Lambda_k \;\; \forall k < n$ and $\Lambda'_n = \lambda_n$. Initialize $n' = n$ and $t = 2$.

2. If $\frac{M'_t}{\Lambda'_t} > \frac{M'_{t-1}}{\Lambda'_{t-1}}$ then Go to step 3. Else let $t = t+1$. If $t > n'$ then STOP else repeat step 2.

3. Modify $s'(t-1) = s'(t-1) \cup s'(t)$; $s'(i) = s'(i+1) \;\; \forall t \leq i \leq n'-1$; $S' = S'/\{n'\}$. Let $\Lambda'_{t-1} = \Lambda'_{t-1} + \Lambda'_t$; $\Lambda'_i = \Lambda'_{i+1} \quad \forall t \leq i \leq n'-1$ and $M'_{t-1} = M'_{t-1} + M'_t$; $M'_i = M'_{i+1} \quad \forall t \leq i \leq n'-1$. Let $n' = n'-1$, $t = t-1$ and go to step 2.

Figure 1: Segmentation Algorithm

Note that if we merge two groups, then the attributes of the groups with an index lower than these two groups do not change. Thus, since the algorithm already checks the condition $\frac{M'_i}{\Lambda'_i} > \frac{M'_{i-1}}{\Lambda'_{i-1}}$ for these lower indexed groups, we only need to check the condition for the groups starting from this merged group and higher. Thus we need to check for this condition at most $2n$ times, hence the algorithm runs in $O(n)$ time. Suppose that the output of this algorithm is the set $S' = \{1, \ldots, n'\}$. Thus all the customer types are segmented into n' groups; with the elements of group i given by $s'(i)$. Moreover the ratio $\frac{M'_i}{\Lambda'_i}$ is decreasing in the index i (from the termination condition of the algorithm). Hence in the absence of any other constraints, it follows that it is optimal to give preemptive priority to the group i over group j if $i < j$ (because of the optimality of the $c\mu$ rule). But we do have the additional constraint that all the waiting times be less than $\frac{R_n}{c_n}$. We will explain how to deal with with these constraints in the next sub-section. However note that if the preemptive priority policy leads to expected waiting times which are all less than $\frac{R_n}{c_n}$, then it has to be the optimal policy. We will denote the waiting time of a type i customer under this policy by $W_i^{c'\mu}$, here c' indicates that the costs are not the actual waiting costs but the modified ones from the segmentation algorithm.

Remark 1 Notice that if we are trying to segment the customer types $\{1, \ldots, i, i+1, \ldots, n\}$, the algorithm will first segment the customer types $\{1, \ldots, i\}$ as if the rest of the customers were not present. The algorithm will consider the types $\{i+1, \ldots, n\}$ only after it finishes segmenting $\{1, \ldots, i\}$ completely.

Correctness of Segmentation Algorithm: We need to prove that the grouping produced by the segmentation algorithm is always valid. As we restrict our attention to the scheduling policies in $\overline{A}$, the polymatroid theory applies; see [7, 13, 35] for more details. The following constraints are always satisfied for any scheduling policy in $\overline{A}$:

$$\sum_{i \in S} \lambda_i W_i \geq \frac{\mu}{\mu - \sum_{i \in S} \lambda_i} \qquad \forall S \subseteq \{1, \ldots, n\}$$

where λ_i is the arrival rate of type i customers into the system. Also, when the set $S = \{1, \ldots, n\}$, the inequality holds as an equality. From standard polymatroid theory, we known that any waiting times vector $(W_1, \ldots, W_n)$ achieved by a policy in $\overline{A}$ can be achieved by the following alternate procedure. The server has n different classes and assigns an incoming type i customer to class j with a probability $\Pi(i,j)$, where Π is an $(n \times n)$ bi-stochastic matrix; the lower indexed classes are given pre-emptive priority over the higher indexed classes.

Let $\pi \in \Pi$ be some bistochastic routing matrix, let T_k denote the average waiting time at class k associated with π. Also, let f_{ij} denote the amount of type i customers that are assigned to class j. Suppose that $i+1$ is a customer type such that $\frac{M_{i+1}}{\Lambda_{i+1}} > \frac{M_i}{\Lambda_i}$ where

$$\mathrm{M}_{i+1} = c_{i+1}(\Lambda_1 + \ldots + \Lambda_{i+1}) - c_i(\Lambda_1 + \ldots + \Lambda_i)$$

We want to argue that we will not have $W_{i+1} > W_i$ in the optimal solution. Suppose that this is the case. Then, there will be a class $k < l$ such that $\pi(i,k) > 0$ and $\pi(i+1,l) > 0$ (otherwise we cannot have $W_{i+1} > W_i$). Note that we will have $T_k < T_l$ (as $k < l$). Now suppose that we make the following changes

$$\begin{aligned} f(i,k) = f(i,k) - \epsilon \quad &, \quad f(i+1,k) = f(i+1,k) + \epsilon \\ f(i,l) = f(i,l) + \epsilon \quad &, \quad f(i+1,l) = f(i+1,l) - \epsilon \end{aligned}$$

As a result of these manipulations, the average waiting time of type i customers increases and that of type $i+1$ customers decreases. Note that by choosing ϵ small enough, we can ensure that we still have $W_i < W_{i+1}$ in the perturbed system. Also, note that the waiting times of the other customer types do not change, as the average waiting time of any class is the same before and after the perturbation. Let Y be the value of $\sum_{i=1}^{n} \frac{M_i}{\Lambda_i} \Lambda_i W_i$ before perturbation and let Y' be the corresponding value after perturbation. Then,

$$\begin{aligned} Y' - Y &= \frac{M_i}{\Lambda_i}\epsilon T_l - \frac{M_i}{\Lambda_i}\epsilon T_k + \frac{M_{i+1}}{\Lambda_{i+1}}\epsilon T_k - \frac{M_{i+1}}{\Lambda_{i+1}}\epsilon T_l \\ &= \epsilon(T_l - T_k)(\frac{M_i}{\Lambda_i} - \frac{M_{i+1}}{\Lambda_{i+1}}) < 0 \end{aligned}$$

Therefore, we will get a solution with a strictly higher revenue, contradicting the fact that the original solution was optimal. Therefore, in optimal solution, we will definitely have $W_i = W_{i+1}$.

Consider the expression which we get by substituting $W_i = W_{i+1}$ in objective function of (9). Observe that this is the same as the profit expression for the problem in which types i and $i+1$ are replaced by a single group with the attributes of type $i+1$ and with a net arrival rate of $\Lambda_i + \Lambda_{i+1}$. Therefore, we can reduce the problem into one consisting of one fewer customer groups. Continuing this argument it follows that the segmentation algorithm finds the correct segmentation for the scheduling problem.

2.3 Searching for the optimal arrival rate

Section 2.2 provides an optimal segmentation of the customers when the optimal arrival rates are known. To determine the optimal policy, one could search over all the potential arrival rates and pick the one that maximizes revenue. In this section, we provide an efficient way to do this by exploiting the underlying structure of the problem. First we prove the following result.

Lemma 7 *Consider the case where the arrival vector is* $(\Lambda_1, \ldots, \Lambda_{n-1})$. *Suppose that in this case, the segmentation algorithm divides the customers into groups* $(G_1, G_2, \ldots, G_k)$*; where group* G_i *gets preemptive priority over group* G_j *for any* $i < j$*. Then, for any arrival vector* $(\Lambda_1, \ldots, \Lambda_{n-1}, \lambda_n)$ *(where* $\lambda_n \leq \Lambda_n$*) , the segmentation algorithm will divide the customers into groups* $(G_1, G_2, \ldots, G_i, L)$ *; where* L *is the group got by pooling groups* $G_{i+1}, \ldots, G_k, n$ *into one single least priority group for some* $i \leq k$*.*

Proof. When there are only 2 customer types, each one of them will be in a separate group (as $\frac{c_1\Lambda_1}{\Lambda_1} > \frac{c_2(\Lambda_1+\lambda_2)-c_1\Lambda_1}{\lambda_2}$). Hence the claim is true for $n = 2$. Assume that the claim is true when $n = s-1$. Below, we will prove that the claim is true when $n = s$.

Consider any value of λ_s and apply the segmentation algorithm for this particular value of λ_s. Suppose that at the end of the Algorithm 1, type s customers constitute a separate group (which has to be the least priority group). Then, due to Remark 1, it follows the way in which the customer types $\{1, \ldots, s-1\}$ are segmented will be independent of the presence of the type s customers. Hence, at the end of the segmentation algorithm, the customers are segmented into the groups $(G_1, \ldots, G_k, s)$ and we are done.

Now suppose that this is not the case and that type s customers are grouped together with some other group at an intermediate stage of the algorithm. Due to Remark 1, it follows that first the segmentation $\{G_1, \ldots, G_k, s\}$ forms and then type s merges with the group G_k. Note that, at this stage, the group consisting of $G_k \cup s$ has the attributes of type s customers. We will prove the lemma by exhibiting an alternate problem which has the same segmentation as this problem; and which satisfies the statement of the lemma.

Consider the alternate problem in which we have all the customer types present in the groups $G_1, \ldots, G_{k-1}$ with the exact same statistics and type s customers with a potential arrival rate

$\sum_{j \in G_k} \Lambda_j + \lambda_s$. Now suppose that we apply the segmentation algorithm to this alternate problem. Again, due to Remark 1, it follows that at some intermediate stage of the algorithm, we will have the segmentation $\{G_1, \ldots, G_{k-1}, G_k \cup s\}$. Hence, the final customer segmentation in this alternate problem will be the same as in the original problem. Note that the number of types in this alternate problem is at most $s-1$. Also, when the last customer type is not present, the optimal segmentation for the remaining types is to form them into the groups $G_1, \ldots, G_{k-1}$. Therefore, by the induction assumption, it follows that this alternate problem will have the structure in the lemma statement; hence our original problem also has this structure and the lemma is proved. ■

We now use Lemma 7 to construct an efficient algorithm. Suppose that type n customer is the highest indexed customer type entering the system. The only variable in the arrival rate vector is the arrival rate of the type n customers, say λ_n (as all the lower indexed types will enter in full). Also, we can focus our attention only on the customer segmentations of the form identified in the statement of Lemma 7. Consider the particular segmentation $\{G_1, \ldots, G_{i-1}, G_i \cup \ldots \cup G_k \cup n\}$ where $i \leq k$. There is no loss of optimality in restricting attention to the region where $W_n^{c'\mu}(\lambda_n) \leq \frac{R_n}{c_n}$ for this particular customer segmentation (due to Remark 2 below). This region is given by

$$W_n^{c'\mu} = \frac{\mu}{(\mu - \Lambda_1 - \ldots - \Lambda_{i-1})(\mu - \Lambda_1 - \ldots - \Lambda_{n-1} - \lambda_n)} \leq \frac{R_n}{c_n}$$

$$\iff \quad \lambda_n \leq \mu - \Lambda_1 - \ldots - \Lambda_{n-1} - \frac{c_n \mu}{R_n(\mu - \Lambda_1 - \ldots - \Lambda_{i-1})} = \lambda_n^A \qquad (10)$$

Thus we will never allow the arrival rate to be more than λ_n^A. Hence, if $\lambda_n^A \leq 0$ then the arrival rate of type n in the optimal solution is 0. Suppose that this is not the case. Now assume that we restrict ourselves to the policies in $\overline{A}$ (due to Remark 3 below, there is no loss of optimality in doing this). As noted in the discussion immediately following the segmentation algorithm, among all the policies in $\overline{A}$, the $c'\mu$ priority rule is optimal over the region $W_n^{c'\mu}(\lambda_n) \leq \frac{R_n}{c_n}$. The profit from this priority rule, as a function of λ_n, is given by

$$\text{Profit}(\lambda_n) = K + R_n \lambda_n - \frac{\mu[c_n(\Lambda_1 + \ldots + \Lambda_{n-1} + \lambda_n) - c_{i-1}(\Lambda_1 + \ldots + \Lambda_{i-1})]}{(\mu - \Lambda_1 - \ldots - \Lambda_{i-1})(\mu - \Lambda_1 - \ldots + -\Lambda_{n-1} - \lambda_n)}$$

where K is a term independent of λ_n. Differentiating, we have

$$\frac{\partial \text{Profit}(\lambda_n)}{\partial \lambda_n} = R_n - \frac{\mu[c_n \mu - c_{i-1}(\Lambda_1 + \ldots + \Lambda_{i-1})]}{(\mu - \Lambda_1 - \ldots - \Lambda_{i-1})(\mu - \Lambda_1 - \ldots - \Lambda_{n-1} - \lambda_n)^2} \geq 0$$

$$\iff (\mu - \Lambda_1 - \ldots + -\Lambda_{n-1} - \lambda_n)^2 \geq \frac{\mu[c_n \mu - c_{i-1}(\Lambda_1 + \ldots + \Lambda_{i-1})]}{R_n(\mu - \Lambda_1 - \ldots - \Lambda_{i-1})} = F^* \qquad (11)$$

If $F^* \leq 0$, then it follows that the first derivative of profit function is always nonnegative and we will try to send in as many type n customers as possible. Thus the optimal arrival rate will be $\min\{\lambda_n^A, \Lambda_n\}$. Now suppose that $F^* > 0$. Then equation (11) can be rewritten as

$$\lambda_n \leq \mu - \Lambda_1 - \ldots - \Lambda_{n-1} - \sqrt{F^*} = \lambda_n^B$$

Therefore if $0 \leq \lambda_n^B \leq \Lambda_n$ then the optimal arrival rate will be $min\{\lambda_n^A, \Lambda_n^B\}$; if $\lambda_n^B > \Lambda_n$ the the optimal arrival rate is $min\{\lambda_n^A, \Lambda_n\}$; and if $\lambda_n^B < 0$ then the optimal arrival rate will be 0. Thus, for a given segmentation, we can calculate the optimal arrival rate and hence the optimal pricing-scheduling policy easily. Note that we can calculate all possible customer segmentations by applying the segmentation algorithm to the arrival rate vector $\{\Lambda_1, \ldots, \Lambda_{n-1}\}$ (due to Lemma 7). By computing the optimal arrival rate for each possible customer segmentation and picking the one with the highest profit, we can get the optimal solution when type n customers are the highest indexed customers entering the system. Finally, we can get the global optimal solution by computing the optimal solutions for each possible $n = 1, \ldots, N$ and choosing the best among them . Note that this procedure for finding the global optimal has $O(N^2)$ complexity (as the segmentation algorithm is the bottleneck procedure and we need to apply it N times).

Remark 2 Note that in the above description, for each customer segmentation, we restrict our attention to the region in which $W_n^{c'\mu}(\lambda_n) \leq \frac{R_n}{c_n}$. We lose nothing by doing this, the reason being as follows. Suppose that the last group is n' and that the arrival vector is such that we have $W_n^{c'\mu}(\lambda_n) > \frac{R_n}{c_n}$. Then, in the optimal solution for this arrival rate vector, we will have

$$W'_{n'} = \frac{R_n}{c_n}$$

where W'_i denotes the average waiting time of group i customers. This can be proved easily by means of a cycle argument similar to the one used in proving the correctness of the segmentation algorithm. The main idea here is that if $W'_{n'} < \frac{R_n}{c_n}$, then we can increase the profit by increasing the waiting time of group n' and reducing the waiting time of some other higher indexed group; thus proving the desired result. Note that in this case the price of the customers in group n' is given by $P_{n'} = R_n - c_n W_n = R_n - c_n W'_{n'} = 0$. Hence, the group n' contributes nothing to the profit. If we now consider the optimal solution in the situations where group n' does not enter the system; we will get a solution which is at least as good as the current solution. Hence, we lose nothing by restricting our attention, for each *customer segmentation*, to the region with $W_n^{c'\mu}(\lambda_n) \leq \frac{R_n}{c_n}$.

Remark 3 Suppose that we do not restrict ourselves to the policies in $\overline{A}$. Then, the optimal solution for the scheduling problem would change only if some of the groups have negative M'_i s. All such groups will have $W'_i = \frac{R_n}{c_n}$ in the optimal solution to the scheduling problem. But note that if some group has a negative coefficient then group n' will definitely have a negative coefficient (as $\frac{M'_i}{\Lambda'_i}$ is decreasing in i). Thus we will have $W'_{n'} = \frac{R_n}{c_n}$ in the optimal solution. Hence by the argument in Remark 2 it follows that there will be another solution, with fewer customer types entering the system, that is at least as good as the current optimal solution. Therefore this new solution, with fewer customer types entering, will also be at least as good as the solution we get from the $c'\mu$ rule; and therefore we lose nothing by restricting our attention to the policies in $\overline{A}$.

3 Restricted service differentiation

In this section, we consider the scenario where we have the additional restriction that at most m classes of service can be offered. Note that all the results proved in section 2.1 hold in this case as well, in particular Theorem 6 holds (there is no change in the analysis). But the optimal policy in the unrestricted service classes case might involve offering more than m classes of service; hence we cannot implement it in this scenario. Our goal is to come up with an optimal solution for this problem, we will do this by first showing how to solve the optimal scheduling problem (9) for a fixed arrival rate vector.

Consider a fixed arrival rate vector, say $(\Lambda_1, \ldots, \lambda_n)$. By applying the same cyclic argument as in the proof of the segmentation algorithm, it can be shown that if $\frac{M_{i+1}}{\Lambda_{i+1}} > \frac{M_i}{\Lambda_i}$ then we will have $W_i = W_{i+1}$ in the optimal solution for (9). The only difference in the proof will be that we will have m service classes instead of n service classes (As it can be readily verified, this makes no difference to the proof technique). Thus we can apply the segmentation algorithm; suppose that this algorithm aggregates the customers into n' groups. Let the sum of arrival rates of the customer types in group i be Λ'_i. Denote $c'_i = \frac{M'_i}{\Lambda'_i}$. These groups will have the property that

$$\frac{M'_1}{\Lambda'_1} > \ldots > \frac{M'_{n'}}{\Lambda'_{n'}} \quad \equiv \quad c'_1 > \ldots > c'_{n'}$$

Recall that for a fixed arrival rate vector, the problem (9) can be reformulated as minimizing the cost function $\sum_{i=1}^{n'} M'_i W'_i = \sum_{i=1}^{n'} c'_i \Lambda'_i W'_i$ (where W'_i is the average waiting time of the customers in group i). For now we ignore the constraint $W_n \leq \frac{R_n}{c_n}$ and consider only the scheduling policies in $\overline{A}$ (later we will show how to relax these). Thus our *modified problem* is to look for work conserving scheduling policies that minimize $\sum_{i=1}^{n'} c'_i \Lambda'_i W'_i$, using at most m classes of service. We claim that in the optimal scheduling policy for this *modified problem*, all the customers within a group will be assigned to exactly one service class. Petersen and Rao [31] proved this for the case where preemptions are not allowed. The proof for our case, where preemptions are allowed, is essentially similar to their argument; we provide the proof for the sake of completeness.

Lemma 8 *In the optimal solution to the modified problem, all the customers within a group are assigned to a single service class. Moreover, customers belonging to group i are assigned to a class no greater than the one to which customers in group $i+1$ are assigned.*

Proof. The ideal solution to the modified problem, in the absence of restriction to m service classes, would be to give preemptive priority to lower indexed groups. Thus, if $n' \leq m$, there is nothing to prove. We will prove the lemma for the case $n' > m$ by using induction on the number of service classes m.

First consider the case when there are only 2 service classes. As before, any scheduling policy can be implemented by assigning the customers to these two classes based on some routing matrix, and then giving preemptive priority to class 1 over class 2. Using a cycle argument similar to the one in the proof of the segmentation algorithm, along with the fact that c'_i is decreasing in i, it can be shown that the arrival rates to the two service classes in the optimal solution can only be of the kind $[(\Lambda'_1, \ldots, \lambda_k); (\Lambda'_k - \lambda_k, \ldots, \Lambda'_{n'})]$. For this particular arrival rate, and the policy that gives preemptive priority to class 1 over class 2, the cost function $\sum_{i=1}^{n'} c'_i \Lambda'_i W'_i$ becomes

$$\text{Cost} \;=\; \frac{c'_1\Lambda'_1 + \ldots + c'_k\lambda_k}{\mu - \Lambda'_1 - \ldots - \Lambda'_{k-1} - \lambda_k} + \frac{c'_k(\Lambda'_k - \lambda_k) + c'_{k+1}\Lambda'_{k+1} + \ldots + c'_n\Lambda'_n}{(\mu - \Lambda'_1 - \ldots - \Lambda'_{k-1} - \lambda_k)(\mu - \sum_{i=1}^{n}\Lambda'_i)}\mu$$

Differentiating the above function with respect to λ_k, we get

$$\begin{aligned}
\frac{\partial \text{ cost}}{\partial \lambda_k} &= \frac{(\mu - \Lambda'_1 - \ldots - \Lambda'_{k-1} - \lambda_k)c'_k + (c'_1\Lambda'_1 + \ldots + c'_k\lambda_k)}{(\mu - \Lambda'_1 - \ldots - \Lambda'_{k-1} - \lambda_k)^2} \\
&\quad + \mu\frac{[\mu - \Lambda'_1 - \ldots - \Lambda'_{k-1} - \lambda_k](-c'_k) + [c'_k(\Lambda'_k - \lambda_k) + \ldots + c'_n\Lambda'_n]}{(\mu - \sum_{i=1}^{n}\Lambda'_i)(\mu - \Lambda'_1 - \ldots - \Lambda'_{k-1} - \lambda_k)^2} \\
&= \frac{1}{(\mu - \Lambda'_1 - \ldots - \Lambda'_{k-1} - \lambda_k)^2} \times \text{ Term independent of } \lambda_k
\end{aligned}$$

If this term is positive then we will have $\lambda_k = \Lambda'_k$ in the optimal solution; if this term is negative then we will have $\lambda_k = 0$ in the optimal solution. In both these cases, group k is assigned to only one service class and hence the lemma holds. Now assume that the lemma is true when there are $m-1$ service classes.

Consider the case where there are m service classes. As before, any scheduling policy can be implemented by assigning the customers to these m classes based on some routing matrix, and then giving preemptive priority to lower indexed service classes. Again using a cycle argument similar to the one in the proof of the segmentation algorithm, along with the fact that c'_i is decreasing in i, it can be shown that arrival rate to the class m can only be of the kind $(\Lambda'_k - \lambda_k, \ldots, \Lambda'_{n'})]$. Note that once the arrival rate into class m is determined, the arrival rate into the rest of the classes will be determined as if there were only $m-1$ classes in the system. Hence from induction assumption, it follows that the entry to class $m-1$ has to be of the form $(\Lambda'_l, \ldots, \Lambda'_{k-1}, \lambda_k)$. Also note that the waiting cost of the customers served in the first $m-2$ classes will be independent of the arrival rates into the classes $m-1$ and m. Therefore for any fixed pair (l,k), and the policy that gives preemptive priority to the lower indexed service classes, the cost function $\sum_{i=1}^{n'} c'_i \Lambda'_i W'_i$ becomes

$$\begin{aligned}
\text{Cost} &= \text{Waiting cost of groups in first } m-2 \text{ classes} + \frac{c'_l\Lambda'_l + \ldots + c'_k\lambda_k}{(\mu - \Lambda'_1 - \ldots - \Lambda'_{l-1})(\mu - \Lambda'_1 - \ldots - \Lambda'_{k-1} - \lambda_k)}\mu \\
&\quad + \frac{c'_k(\Lambda'_k - \lambda_k) + c'_{k+1}\Lambda'_{k+1} + \ldots + c'_n\Lambda'_n}{(\mu - \Lambda'_1 - \ldots - \Lambda'_{k-1} - \lambda_k)(\mu - \sum_{i=1}^{n}\Lambda'_i)}\mu \\
&= \text{Term independent of } \lambda_k + \frac{c'_l\Lambda'_l + \ldots + c'_k\lambda_k}{(\mu - \Lambda'_1 - \ldots - \Lambda'_{l-1})(\mu - \Lambda'_1 - \ldots - \Lambda'_{k-1} - \lambda_k)}\mu \\
&\quad + \frac{c'_k(\Lambda'_k - \lambda_k) + c'_{k+1}\Lambda'_{k+1} + \ldots + c'_n\Lambda'_n}{(\mu - \Lambda'_1 - \ldots - \Lambda'_{k-1} - \lambda_k)(\mu - \sum_{i=1}^{n}\Lambda'_i)}\mu
\end{aligned}$$

Mimicking the proof in the 2 service classes case, it can be shown that the minimum of the above function will be attained either at $\lambda_k = \Lambda'_k$ or at $\lambda_k = 0$. Since this is true for any pair (l, k), it follows that every group is sent to exactly one customer class and the lemma is proved ■

Next we give a dynamic programming formulation to solve for an optimal scheduling policy, for a given value of $(\Lambda'_1, \ldots, \Lambda'_{n'})$. As already noted, for fixed arrival rates the revenue maximization problem is equivalent to an optimal scheduling problem with cost function $\sum_{i=1}^{n'} c'_i \Lambda'_i W'_i$. Here, the cost of group i is $c'_i \Lambda'_i W'_i$. Let $T(i,j)$ be the optimum cost of the customer groups $(1, 2, \ldots, i)$ when they use classes $(1, 2, \ldots, j)$. Suppose that the customer groups $\{i - k + 1, \ldots, i\}$ are assigned to class j in this optimal solution. Then notice that the waiting cost of the customer groups $\{1, \ldots, i - k\}$ in this optimal solution will be exactly $T(i - k, j - 1)$ (a shortest path argument applies here). Since there is always an optimal solution in which some customer groups are assigned to only the last class (from Lemma 8), we can solve for $T(i, j)$ by simply searching over all possible values of k. Therefore, the dynamic programming recursion is :

$$T(i,j) = \min_{k \leq i-(j-1)} [T(i-k, j-1) + \text{ cost of serving last k types in class j }] \tag{12}$$

This dynamic program can be solved in $O(n^2)$ time. The optimal scheduling cost is given by $\min_{1 \leq j \leq m} T(n, j)$. Thus, given a particular arrival rate vector $(\Lambda_1, \ldots, \lambda_n)$, we can solve for the optimal scheduling policy efficiently. We still need to search for the optimal arrival rate vector; below we give a scheme to do this.

Suppose that type n customers are the highest indexed ones to enter the system. For any value of λ_n the groups formed will be of the form $(G_1, G_2, \ldots, G_i, G_{i+1} \cup \ldots \cup G_k \cup \{n\})$; where G_i are as defined in Lemma 7. If $i \leq m - 1$ then we can calculate the optimal arrival rate for this segmentation using the procedure outlined immediately after Lemma 7. If $i > m - 1$, then we know that the groups have to be grouped together further. Due to Lemma 8, we know that the customers assigned to the class m will be of the form $G_{s+1} \cup \ldots \cup G_k \cup \{n\}$ (where $s \geq m - 1$). Note that once we fix the groups which are going to be in the service class m, there is only one way to allocate the remaining groups to the classes $\{1, \ldots, m - 1\}$. The cost of doing this can be easily calculated using the DP recursion in equation (12). Hence only the cost of the groups in class m will depend on λ_n. Hence, for this particular constitution of class m, we can calculate the optimal arrival rate using the procedure described after Lemma 7. By computing the optimal values for each possible constitution of class m, and choosing the best among them, we get the optimal solution when type n is the highest indexed customer type entering the system. Finally, by solving the problem for all $n = 1, \ldots, N$ and comparing them, we can find the globally optimal solution. Note that for a given n, the bottleneck operations are the segmentation algorithm and the DP recursion of equation (12), the former runs in $O(n)$ time and the latter in $O(n^2)$ time. Therefore, we can solve the restricted service differentiation problem in $O(N^3)$ time.

Recall that we had ignored the constraint $W_n \leq \frac{R_n}{c_n}$ and restricted ourselves to the policies in $\overline{A}$. But in the procedure following Lemma 7, we would restrict ourselves to the region in which giving preemptive priority to lower indexed classes would lead to an average waiting time of at most $\frac{R_n}{c_n}$ for the customers in group n' (due to Remark 2 there is no loss of optimality in doing this). Hence, the constraint $W_n \leq \frac{R_n}{c_n}$ is being considered implicitly. The restriction to $\overline{A}$ was to make sure that we do not idle any jobs even if some of the $M_i's$ are negative. Due to Remark 3, there is no loss of optimality in doing this.

4 Continuum of types

In this section, we consider the scenario in which there are a continuum of customer types, and each customer type can be represented by his value for service. Customers arrive to the system following a Poisson process with Λ; the service time of each customer is exponential with rate μ. The customers differ only in their value of service i.e. their willingness to pay for service in the absence of delay. The service values are i.i.d. draws from a continuous distribution Φ (independent of arrival and service times) with pdf ϕ. We assume that ϕ is strictly positive and continuous on $[\underline{v}, \overline{v}]$, where $0 \leq \underline{v} < \overline{v} \leq \infty$. Let $\overline{\Phi} = 1 - \Phi$. If all the jobs with values $\geq s$ join the system, the arrival rate into the system will be $\lambda = \Lambda\overline{\Phi}(s)$. Conversely, when the arrival rate is λ, the value of the marginal customer is equal to $R(\lambda) = \overline{\Phi}^{-1}(\frac{\lambda}{\Lambda})$, where $\overline{\Phi}^{-1}$ is the inverse of $\overline{\Phi}$. Observe that we will have $R(\lambda) > 0$ and $R'(\lambda) < 0$ for $\lambda < \Lambda$.

We assume that the customers have the generalized delay cost structure proposed by Afeche and Mendelson [2]. Specifically, the utility of a customer with service value s who pays P and experiences a delay t is given by $u(s,t,P) = sD(t) - C(t) - P$. Here $C(t)$ is an increasing *Delay cost function* with $C(0) = 0$ and $D(t)$ is a non-increasing *Delay discount function* with $D(0) = 1$. We restrict our attention to the case where the functions $D(t)$ and $C(t)$ are linear : $D(t) = 1 - d.t$ and $C(t) = c.t$; with delay sensitive parameters $d > 0$ and $c \geq 0$ respectively. The utility function reduces to $u(s,t,P) = s - (sd + c)t - P$. Therefore, we can view our system as one in which the customers have heterogeneous value of service and where the waiting cost of a customer with value s is linear with rate $c + sd$. Let $C(\lambda) = c + dR(\lambda)$. Here $C(\lambda)$ is the waiting cost rate of a customer with value $R(\lambda)$. Note that we will have

$$\lambda_1 < \lambda_2 \implies R(\lambda_1) > R(\lambda_2) \implies R(\lambda_1)(c + dR(\lambda_2)) > R(\lambda_2)(c + dR(\lambda_1)) \implies \frac{R(\lambda_1)}{C(\lambda_1)} > \frac{R(\lambda_2)}{C(\lambda_2)}$$

Therefore, the ratio $\frac{R(\lambda)}{C(\lambda)}$ is decreasing in λ. Thus, the generalized delay cost structure can be viewed as a continuous version of our 'discrete model' assumption that the ratio $\frac{R_i}{c_i}$ be decreasing in c_i. We want to solve for the profit maximizing mechanism in this setting; the priority auction mechanism, that schedules the customers according to the $c\mu$ rule, is a natural candidate for this. In this section, we show that the optimal mechanism might not schedule the customers according to the strict $c\mu$ rule; thus the priority auction mechanism need not always be optimal. We identify the conditions under which this priority auction mechanism is optimal.

Priority auction Analysis. First, we re-derive expressions for the bids in the priority auction mechanism, originally obtained by Afeche and Mendelson in [2]. We find this alternate way of deriving them useful for our purposes. We provide an informal and intuitive argument and show that the expressions we get are the same as the ones in [2]. Recall that in the discrete case the price charged for type i customers can be expressed as (see equation (8))

$$P_i = P_n + \sum_{k=i}^{n-1} \Delta P_k = P_n + \sum_{k=i}^{n-1} c_k(W_{k+1} - W_k) = P_n + \sum_{k=i}^{n-1} c_k \Delta W_k \tag{13}$$

Now consider the continuous version. Let λ be the entry rate into the system when we use the auction. Then, analogous to the discrete case, all the customers with values in $[\overline{\Phi}^{-1}(\frac{\lambda}{\Lambda}), \overline{v}]$ will enter the system and get served. Moreover, the customers whose value of service is less than $\overline{\Phi}^{-1}(\frac{\lambda}{\Lambda})$ choose not to enter the system. Let P_s denote the amount paid by a customer with value $R(s)$ and let $C(s)$ denote his waiting cost. Thus, P_λ is the amount which the marginal customer pays for entering the system. The continuous version of equation (13) will be

$$P_r \;=\; P_\lambda + \int_r^\lambda C(x)d(W(x)) = P_\lambda + \int_r^\lambda C(x)W'(x)dx \tag{14}$$

where $W(x)$ is the waiting time of a customer with value $\overline{\Phi}^{-1}(\frac{x}{\Lambda})$. Since the customers with higher value are given pre-emptive priority over the customers with lower value in the auction, the expression for $W(x)$ is given by $W(x) = \frac{\mu}{(\mu - x)^2}$. Note that the expressions for the bids in (14) is different from the expression for equilibrium bids derived in [2]. However, we know that $P_\lambda = R(\lambda) - C(\lambda)W(\lambda)$ (because, analogous to the discrete case, the net utility of the marginal customer will be 0). Substituting for P_λ in (14), and integrating by parts, it can be seen easily that this reduces to the expression derived in [2]. Therefore, the bids in (14) are the equilibrium bids. The overall revenue earned is given by

$$\begin{aligned}
\text{Revenue} &= \int_0^\lambda P_y dy = \lambda P_\lambda + \int_0^\lambda \Big(\int_y^\lambda C(x)W'(x)dx\Big)dy \\
&= \lambda P_\lambda + \int_0^\lambda C(x)W'(x)\Big(\int_0^x dy\Big)dx \qquad \text{[by interchanging the integrals]} \\
&= \lambda P_\lambda + \int_0^\lambda xC(x)W'(x)dx \\
&= \lambda P_\lambda + \lambda C(\lambda)W(\lambda) - \int_0^\lambda W(x)[xC'(x) + C(x)]dx \qquad \text{[integrating by parts]} \\
&= \lambda[R(\lambda) - C(\lambda)W(\lambda)] + \lambda C(\lambda)W(\lambda) - \int_0^\lambda W(x)[xC'(x) + C(x)]dx \\
&= \lambda R(\lambda) - \int_0^\lambda W(x)[xC'(x) + C(x)]dx \qquad (15)
\end{aligned}$$

Priority auction Optimality. We have seen that the revenue generated in the priority auction that serves all the customers with value greater than $R(\lambda)$ is given by the expression in equation (15). But is this optimal? Recall that in the priority auction, the customers are served according to the $c\mu$ rule. Hence, the optimality of the priority auction in the continuous types case can be thought of being analogous to the optimality of the $c\mu$ rule in the discrete types case. We will analyze the continuous case by approximating it as a discrete types case problem (with a large number of types).

Given any instance of the problem with continuous types, and a $\delta > 0$, we can associate a natural discrete-types problem with $K = \frac{\Lambda}{\delta}$ types of customers. Let the value of service of type k customers be given by $\overline{R}^k = R[(k-1)\delta]$. Also let the cost of waiting of type k customers be given by $\overline{C}^k = C[(k-1)\delta]$ and let $\Lambda_k = \delta$ for all customer types. Note that in this new problem we will have $\overline{R}^k$ and $\overline{C}^k$ decreasing in k, and $\frac{\overline{R}^k}{\overline{C}^k}$ will also be decreasing in k (as $\frac{R(\lambda)}{C(\lambda)}$ is decreasing in λ). Thus, this discrete-types problem satisfies all the model assumptions of section 2 and all the results of section 2 apply. We claim that the optimal revenue of this discrete types problem is higher than that of the continuous types problem. The intuitive reasoning is as follows. In the discrete-types problem, all the customers with values of service in $[R(i\delta), R((i+1)\delta)\ )$ are replaced by customers of value $R(i\delta)$. As it is more beneficial to have customers of higher value as opposed to lower valued customers (due to our delay cost structure), we can expect the discrete-types problem to have a higher revenue than the continuum problem. We will prove this formally below.

Consider an optimal solution for the continuous case. Suppose that the lowest valued customer entering the system has a value $R(x)$. Let P_s and W_s denote the price and waiting time of the customers with value $R(s)$. For convenience we also denote $R_s = R(s)$ and $C_s = C(s)$. Without loss of generality, we can restrict attention to the pricing-scheduling policies where we have $P_x \geq 0$ and $W_s \leq W_t$ if $R(s) \geq R(t)$ (for the same reasons as in the discrete types problem). Now consider the discretised version of this problem; suppose that $m = \lceil \frac{x}{\delta} \rceil$. In the discrete problem we want to have the same amount of customer entry as in the continuous problem. Thus we want a solution in which all of customer types $\{1, \ldots, m-1\}$ enter the system in full and $\delta_1 = x - (m-1)\delta$ amount of type m customers enter the system. Our aim is to come up with a pricing - scheduling policy for the discrete problem with the same arrival amount and a revenue higher than in the continuous case. We will denote the price and waiting time of the customer type i in the discrete problem by $\overline{P}^i$ and $\overline{W}^i$ respectively. Suppose that we schedule the customers such that

$$\overline{W}^i = \frac{1}{\delta}\int_{(i-1)\delta}^{i\delta} W_y dy \quad \forall i < m; \quad \overline{W}^m = \frac{1}{\delta_1}\int_{(m-1)\delta}^{x} W_y dy$$

These waiting times are just the average waiting times that these *customer types* would have had to wait in the continuous problem, and can be easily achieved by a scheduling policy $\overline{r} \in A$. Consider the prices

$$\overline{P}^m = \overline{R}^m - \overline{C}^m \overline{W}^m; \qquad \overline{P}^i = \overline{P}^{i+1} + \overline{C}^i(\overline{W}^{i+1} - \overline{W}^i) \quad \forall i = 1, \ldots, m-1$$

These prices have the same structure as in equation (8). Note that we will have $\overline{W}^i \leq \overline{W}^j \quad \forall i < j$. Also note that $0 \leq P_x \leq R_x - C_x W_x \implies W_x \leq \frac{R_x}{c_x} < \frac{R_{(m-1)\delta}}{C_{(m-1)\delta}}$; therefore we have $\overline{W}^m \leq W_x <$

$\frac{R_{(m-1)\delta}}{C_{(m-1)\delta}}$. Therefore, by Lemma 5, it follows that the policy $(\overline{P}, \overline{r})$ is incentive compatible and individually rational; hence it is a valid policy. Below we will prove that the revenue of this policy is at least as high as the continuous types revenue.

Lemma 9 *The revenue generated by the policy $(\overline{P}, \overline{r})$ in the 'discrete problem' is at least as high as the corresponding continuous types revenue.*

Proof. In the continuous case, for any $(m-1)\delta \leq k \leq x$, we have

$$\begin{aligned} P_k &\leq P_x + C_k(W_x - W_k) \qquad \text{[From IC constraints in equation (2)]} \\ &\leq P_x + C_{(m-1)\delta}(W_x - W_k) \qquad [\text{ as } C_{(m-1)\delta} \geq C_k \text{ and } W_x - W_k \geq 0] \\ &\leq R(x) - C_x W_x + C_{(m-1)\delta} W_x - C_{(m-1)\delta} W_k \qquad [\text{ IR for } x] \end{aligned}$$

Note that

$$R_x - C_x W_x + C_{(m-1)\delta} W_x \leq R_{(m-1)\delta} \iff W_x \leq \frac{R_{(m-1)\delta} - R_x}{C_{(m-1)\delta} - C_x}$$

But, we have $W_x \leq \frac{R_x}{C_x}$ (as $P_x \geq 0$) and $\frac{R_x}{C_x} \leq \frac{R_{(m-1)\delta} - R_x}{C_{(m-1)\delta} - C_x}$. Therefore the above equation is true and we have

$$P_k \leq R_{(m-1)\delta} - C_{(m-1)\delta} W_k \tag{16}$$

Integrating we have

$$\int_{(m-1)\delta}^{(m-1)\delta+\delta_1} P_k dk \leq \delta_1 R_{(m-1)\delta} - C_{(m-1)\delta} \overline{W}^m \delta_1 = \delta_1 \overline{P}^m$$

Thus, the revenue earned in the continuous types solution from the customers in $[(m-1)\delta, x]$ is less than the revenue from the *type m* customers in the discrete case.

In the continuum types solution, for any $(m-2)\delta \leq k \leq (m-1)\delta$, we have

$$\begin{aligned} P_k &\leq P_l + C_k(W_l - W_k) \qquad \forall (m-1)\delta \leq l \leq x \qquad \text{[due to eqn (2)]} \\ \Rightarrow P_k &\leq R_{(m-1)\delta} - C_{(m-1)\delta} W_l + C_k(W_l - W_k) \qquad \forall (m-1)\delta \leq l \leq x \qquad \text{[from eqn (16)]} \\ \Rightarrow \int_{(m-1)\delta}^{(m-1)\delta+\delta_1} P_k dl &\leq \int_{(m-1)\delta}^{(m-1)\delta+\delta_1} \Big[R_{(m-1)\delta} - C_{(m-1)\delta} W_l + C_k(W_l - W_k) \Big] dl \qquad \text{[Integrating]} \\ \Rightarrow \delta_1 P_k &\leq \delta_1 R_{(m-1)\delta} - C_{(m-1)\delta} \delta_1 \overline{W}^m + \delta_1 C_k(\overline{W}^m - W_k) \\ \Rightarrow P_k &\leq \overline{P}^m + C_k(\overline{W}^m - W_k) \\ \Rightarrow P_k &\leq \overline{P}^m + C_{(m-2)\delta}(\overline{W}^m - W_k) \qquad [\text{ as } C_k \leq C_{(m-2)\delta} \text{ and } \overline{W}^m - W_k \geq 0] \end{aligned} \tag{17}$$

Integrating, we have

$$\int_{(m-2)\delta}^{(m-1)\delta} P_k dy \leq \delta \overline{P}^m + \delta C_{(m-2)\delta}(\overline{W}^m - \overline{W}^{m-1}) = \delta \overline{P}^{m-1}$$

Thus, the revenue earned in the continuum types solution from the customers in $[(m-2)\delta,(m-1)\delta)$ is less than the revenue from the type $m-1$ customers in the discrete case.

We will finish the proof by induction. For some i suppose that the revenue earned in the continuous types solution from the customers in $[(i-1)\delta, i\delta)$ is less than the revenue from the type i customers in the discrete case. Also suppose that

$$P_k \leq \overline{P}^{i+1} + C_{(i-1)\delta}(\overline{W}^{i+1} - W_k) \quad \forall \quad (i-1)\delta \leq k < i\delta \tag{18}$$

In the continuous types solution, for any $(i-2)\delta \leq k \leq (i-1)\delta$, we have

$$\begin{aligned}
P_k &\leq P_l + C_k(W_l - W_k) \quad \forall (i-1)\delta \leq l \leq i\delta \qquad [\text{ due to eqn (2) }] \\
\Rightarrow P_k &\leq \overline{P}^{i+1} + C_{(i-1)\delta}(\overline{W}^{i+1} - W_l) + C_k(W_l - W_k) \quad \forall (i-1)\delta \leq l \leq i\delta \qquad [\text{ from eqn (18) }] \\
\Rightarrow \int_{(i-1)\delta}^{i\delta} P_k dl &\leq \int_{(i-1)\delta}^{i\delta} \left[\overline{P}^{i+1} + C_{(i-1)\delta}(\overline{W}^{i+1} - W_l) + C_k(W_l - W_k)\right] dl \qquad [\text{Integrating}] \\
\Rightarrow \delta P_k &\leq \delta\overline{P}^{i+1} + \delta C_{(i-1)\delta}(\overline{W}^{i+1} - \overline{W}^{i}) + \delta C_k(\overline{W}^{i} - W_k) \\
\Rightarrow P_k &\leq \overline{P}^{i} + C_k(\overline{W}^{i} - W_k) \\
\Rightarrow P_k &\leq \overline{P}^{i} + C_{(i-2)\delta}(\overline{W}^{i} - W_k) \qquad [\text{ as } C_k \leq C_{(i-2)\delta} \text{ and } \overline{W}^{i} - W_k \geq 0] \qquad (19)
\end{aligned}$$

Integrating, we have

$$\int_{(i-2)\delta}^{(i-1)\delta} P_k dy \leq \delta\overline{P}^{i} + \delta C_{(i-2)\delta}(\overline{W}^{i} - \overline{W}^{i-1}) = \delta\overline{P}^{i-1}$$

Thus, the revenue earned in the continuous types solution from the customers in $[(i-2)\delta,(i-1)\delta)$ is less than the revenue from the type $i-1$ customers in the discrete case. Thus, by induction, it follows that this is true for all the intervals. By adding the revenue over all these intervals, we can conclude that the revenue in the 'discrete types' problem is at least as high as the revenue in the continuous types problem. ■

Due to Lemma 9, in order to prove the optimality of the priority auction mechanism, it is enough to prove that the optimal revenue of the discrete types problem converges to the priority auction revenue as $\delta \to 0$. As already mentioned, the optimality of the priority auction can be intuitively viewed as being equivalent to the optimality of the $c\mu$ rule in the discrete types approximation. Recall from section 2 that in the discrete case, for the optimality of the $c\mu$ rule, we would need

$$\frac{c_i(\sum_{k=1}^{i-1}\Lambda_k + \Lambda_i) - c_{i-1}(\sum_{k=1}^{i-1}\Lambda_k)}{\Lambda_i} > \frac{c_{i+1}(\sum_{k=1}^{i-1}\Lambda_k + \Lambda_i + \Lambda_{i+1}) - c_i(\sum_{k=1}^{i-1}\Lambda_k + \Lambda_i)}{\Lambda_{i+1}} \tag{20}$$

Note that, in the 'discrete approximation problem', we can always express any type i (where $1 \leq i \leq K$) as $i = \alpha K$ (for some $0 < \alpha < 1$). Then $\sum_{k=1}^{i-1}\Lambda_k = (i-1)\delta = (\alpha K - 1)\delta = \alpha\Lambda - \delta$.

Denote $x = \alpha\Lambda$. Equation (20) can be rewritten as

$$\begin{aligned}
& \frac{xC(x) - (x-\delta)C(x-\delta)}{\delta} > \frac{(x+\delta)C(x+\delta) - xC(x)}{\delta} && (21)\\
\iff\quad & x\frac{C(x) - C(x-\delta)}{\delta} - x\frac{C(x+\delta) - C(x)}{\delta} > C(x+\delta) - C(x-\delta) &&\\
\iff\quad & x\frac{\frac{C(x)-C(x-\delta)}{\delta} - \frac{C(x+\delta)-C(x)}{\delta}}{\delta} > 2\frac{C(x+\delta) - C(x-\delta)}{2\delta} &&\\
\iff\quad & -xC''(x) > 2C'(x) \qquad [\text{ as } \delta \to 0] &&\\
\iff\quad & (xC(x))'' < 0 && (22)
\end{aligned}$$

Below we prove that if equation (22) is satisfied then the $c\mu$ rule is optimal for the discrete types problem.

Lemma 10 *If the function $xC(x)$ is concave then $c\mu$ rule is optimal for the 'discrete' approximation problem.*

Proof. Recall that there are m customer types entering the system for service, and that $\delta_1 \leq \delta$ amount of type m enters the system. If $i+1 \neq m$ then equation (20) is exactly the same as equation (21), and hence is satisfied (follows from the equivalence of (21) and (22) above). Now suppose that $i+1 = m$; equation (20) can be rewritten as

$$\begin{aligned}
& \frac{xC(x) - (x-\delta)C(x-\delta)}{\delta} > \frac{(x+\delta_1)C(x+\delta_1) - xC(x)}{\delta_1}\\
\iff\quad & xC(x) > \frac{\delta}{\delta+\delta_1}(x+\delta_1)C(x+\delta_1) + \frac{\delta_1}{\delta+\delta_1}(x+\delta)C(x-\delta_1)
\end{aligned}$$

which is satisfied due to concavity of $xC(x)$. Hence $c\mu$ rule is optimal for the discrete types problem. ■

The only thing left to show is that, if we use the $c\mu$ scheduling rule, the discrete problem revenue converges to the priority auction revenue as $\delta \to 0$. Suppose that exactly x amount of customers enter the system when we use the $c\mu$ scheduling rule, let $m = \lceil \frac{x}{\delta} \rceil$ and $\delta_1 = x - \delta\lfloor \frac{x}{\delta} \rceil$. The revenue is given by (rewriting the objective function of (9))

$$\text{Revenue } = xR_m - \sum_{i=1}^{m} c_i W_i \delta - \sum_{i=1}^{m-1} (c_{i+1} - c_i) W_i i\delta + (\delta - \delta_1) c_m W_m \tag{23}$$

As $\delta \to 0$, equation (23) converges to

$$\text{Revenue } = xR(x) - \int_0^x C(\lambda)W(\lambda)d\lambda - \int_0^x C'(\lambda)W(\lambda)\lambda d\lambda$$

which is the same as the priority auction revenue with arrival rate x (see eqn (15)). Thus we have proved that the priority auction mechanism is optimal when $xC(x)$ is concave. Note that, in a sense, the concavity of $xC(x)$ is also a necessary condition necessary condition for priority auction optimality. Specifically, the following theorem is true.

Theorem 11 *Suppose that the arrival rate into the system in the optimal priority auction is λ^{opt}. Then, this optimal priority auction mechanism is the profit-maximizing mechanism* if and only if *$xC(x)$ is concave for $0 \leq x \leq \lambda^{opt}$.*

The necessity of the concavity of $xC(x)$ for the priority auction optimality should not come as a surprise considering the equivalence of equations (21) and (22). This part can again be proved by using a limiting discrete-types case argument to exhibit an alternate policy which achieves a higher revenue than the priority auction. Instead of proving it, we will give an example in which $xC(x)$ is not convex for $0 \leq x \leq \lambda^{opt}$ and exhibit a pricing-scheduling policy that achieves a higher revenue than the priority auction.

Counterexample for priority auction Optimality

Let $R(x) = 6 + x^2 - 3x$, $\mu = 1.7$, $d = \frac{0.09}{1.7}$ and $c = 0$. Let the potential customer base be $\Lambda = 1.5$. Recall that $C(x) = dR(x) + c = dR(x)$. Note that in this case $xc(x)$ will be concave *if and only if* $L(x) = xR(x) = 6x + x^3 - 3x^2$ is concave. $L''(x) = 6x - 6 \quad \geq 0 \iff x \leq 1$ and hence $L(x)$ is not concave for $x > 1$. As we show below, the revenue in the optimal priority auction will be 4.4373 and the arrival rate in this solution will be $\lambda^{opt} = 1.4$. Since $\lambda^{opt} > 1$, the priority auction mechanism will not be optimal in this setting. We will exhibit an alternate mechanism with a revenue of 4.4537, which is more than the priority auction revenue.

The condition for the optimal priority auction arrival rate is $L'(\lambda)[1 - d\frac{\mu}{\mu-\lambda)^2}] = 0$ (from equation (29) in [2]). But, note that $L'(\lambda) = 6 + 3x^2 - 6x = 3(x-1)^2 + 3 \geq 3$. Therefore, the optimality condition reduces to $1 - d\frac{\mu}{\mu-\lambda)^2} = 0 \implies \lambda^{opt} = \mu - \sqrt{d\mu} = 1.4$. Therefore, the optimal priority auction profit is given by

$$\begin{aligned}
\text{Profit} &= \lambda^{opt}[6 - 3\lambda^{opt} + (\lambda^{opt})^2] - \int_0^{\lambda^{opt}} d[6 + x^2 - 3x + x(2x-3)]\frac{\mu}{(\mu-x)^2}dx \\
&= \lambda^{opt}[6 - 3\lambda^{opt} + (\lambda^{opt})^2] - d\mu\int_0^{\lambda^{opt}} [3 - \frac{6(\mu-1)}{\mu-x} + \frac{3(\mu^2-2\mu+2)}{(\mu-x)^2}]dx \\
&= \lambda^{opt}[6 - 3\lambda^{opt} + (\lambda^{opt})^2] - d\mu[3\lambda^{opt} - 6(\mu-1)log(\frac{\mu}{\mu-\lambda^{opt}}) + 3(\mu^2-2\mu+2)(\frac{1}{\mu-\lambda^{opt}} - \frac{1}{\mu})] \\
&= 4.4373
\end{aligned}$$

Now consider the following alternate mechanism in which the arrival rate is $\lambda = 1.4$. The customers in $\lambda \in [0, 0.8]$ will be given preemptive priority over the rest and within themselves they will be served according to the $c\mu$ rule. The customers in $(0.8, 1.4]$ will receive the least priority and within themselves they will be served according to the FCFS rule. Note that this is an admissible scheduling mechanism. First, we will identify the IC pricing that will achieve this scheduling rule. Recall, from the structure of the optimal solution in section 2, that if we promise the same waiting time to two customers then both of them will be charged the same. Hence, all the customers in (0.8.1.4] will be charged the same price and all of them will be treated like a customer with value $R(1.4)$. The expected waiting time of a $R(1.4)$ customer is $W_2 = \frac{\mu}{(\mu-0.8)(\mu-1.4)} = 6.2968$. Therefore, the price charged to these customers will be $P_2 = R(1.4) - c(1.4) * W_2 = 2.5067$ (this is because as before the marginal customer in the system will have 0 benefit from entering the system).

Now consider the customers who are served by the $c\mu$ rule; the marginal customer among them is the customer with value $R(0.8)$. The waiting time for this customer is $W_{0.8} = \frac{\mu}{(\mu-0.8)^2} = 2.0988$. Also, his waiting cost is $C(0.8) = 0.2245$. Therefore, the price charged to this customer will be $P_{0.8} = P_2 + C(0.8)[W_2 - W_{0.8}] = 3.4487$ [from the pricing structure in section 2]. Also, the price charged for a customer with revenue $R(x)$ (where $x \leq 0.8$) is given by $P_x = P_{0.8} + \int_x^{0.8} C(x)W'(x)dx$ [analogous to equation (14)]. It is easy to check that these prices are IC (these are in the same spirit as in section 2). Therefore, the revenue in this mechanism is given by

$$\begin{aligned}
\text{Profit} &= 0.6 * P_2 + \int_0^{0.8} P_y dy \\
&= 0.6P_2 + \int_0^{0.8} \big[P_{0.8} + \int_y^{0.8} C(x)W'(x)dx\big] dy \\
&= 0.6P_2 + 0.8P_{0.8} + \int_0^{0.8} xC(x)W'(x)dx \qquad [\text{ By interchanging the integrals }] \\
&= 0.6P_2 + 0.8P_{0.8} + 0.8C(0.8)W(0.8) - \int_0^{0.8} W(x)[xC'(x) + C(x)]dx \qquad [\text{ Integrating by parts }] \\
&= 0.6P_2 + 0.8P_{0.8} + 0.8C(0.8)W(0.8) - \int_0^{0.8} d[6 + x^2 - 3x + x(2x-3)]\frac{\mu}{(\mu-x)^2}dx \\
&= 4.6399 - d\mu[3 * 0.8 - 6(\mu-1)log(\frac{\mu}{\mu-0.8}) + 3(\mu^2 - 2\mu + 2)(\frac{1}{\mu-0.8} - \frac{1}{\mu})] \\
&= 4.4538
\end{aligned}$$

Therefore the profit from this alternate mechanism is greater than that in the optimal priority auction. Hence priority auction mechanisms are no longer the optimal mechanisms.

5 Simulations

In this section we study the scenario where the customer base is continuous and the service provider is restricted to m service classes. In many real life situations it is costlier to have more classes of service. Hence service providers offer a limited number of differentiated services and let the customers choose among these services; the customers opting for a higher priced service class get preemptive priority over the customers opting for a lower priced service class. The service provider has to decide on the number of classes of service to offer; the prices of these services and the way the customers are segmented among these service classes (which in turn depends on the prices). It is of interest to understand the dependence between the characteristics of the customer base and these decisions. The results of section 4 suggest that there might be a

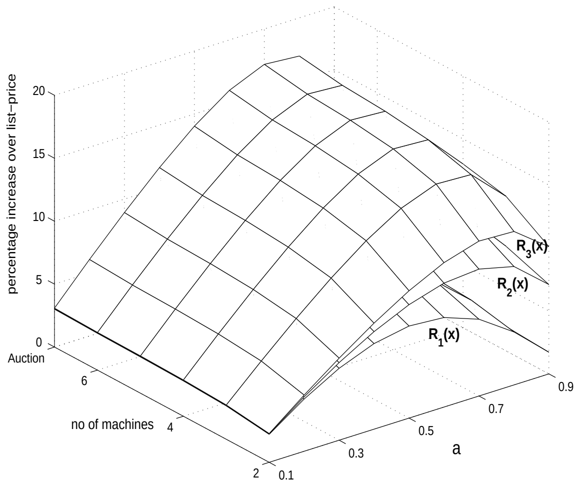


Figure 2: For the parameters $d = 0.6; c = 0.1; \mu = 1.35; \Lambda = 1$

dependence between 'degree' of of concavity of $xR(x)$ and these decisions; we try to understand this dependence using simulations. We used the following functions in our simulations:

(i) $R_1(x) = (1 - a)x^{-a}$

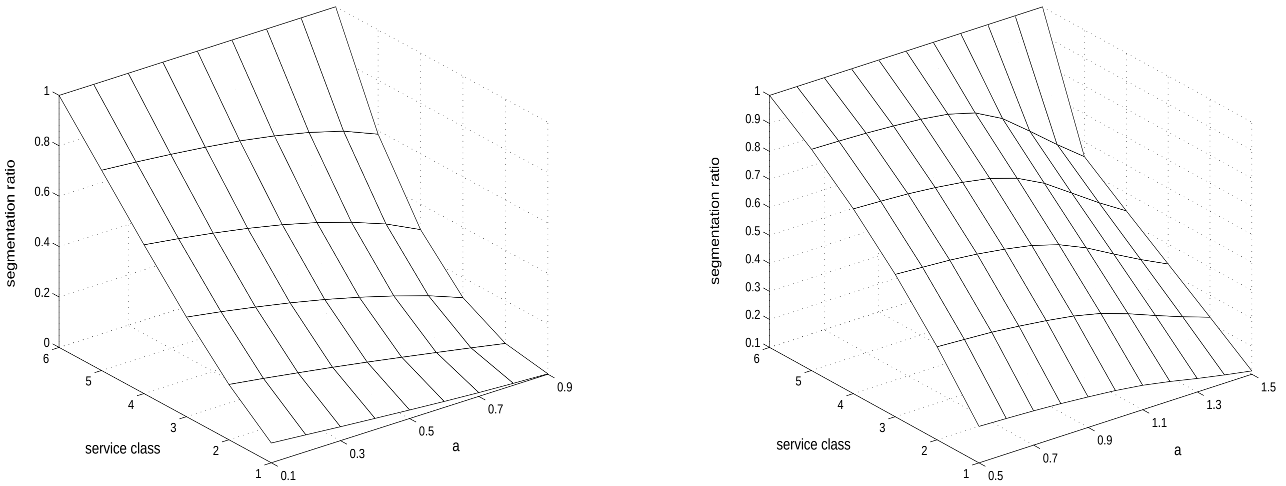
(ii) $R_2(x) = x^{-a}$

(iii) $R_3(x) = \frac{1}{1-a}x^{-a}$

(iv) $R_4(x) = 6 - 3x + ax^2$

The reason for choosing these functions is as follows. Note that $[xR_i(x)]'', i = 1, 2, 3$ will be negative for all values of $x \in [0, 1]; a \in (0, 1)$; hence these are concave. Moreover, for a fixed value of a, the absolute value of $(xR_i(x))''$ increases with i (for $i = 1, 2, 3$); therefore in a sense (iii) is 'more' concave than (ii) and (i). In this same sense, the concavity of the function $xR_3(x)$ increases with a. We chose the function $R_4(x)$ because concavity of the function $xR_4(x)$ decreases with increase in a; in fact beyond $a = 1$ the second derivative of this function is non-negative for certain values of x. Also $R_4(x)$ is polynomial, whereas the other three functions are exponential. Note that in all these functions, the value of $R_i(x)$ increases with a.

Number of service classes. Here we focus on the issue of number of service classes offered. The naivest strategy for the service provider is to have a single list price for all the customers; the service provider can improve on this by providing multiple service classes. Let $Z(m)$ denote the optimal revenue when using m service classes, we focus on the ratio $\frac{z(m)}{z(1)}$ in our simulations. This ratio indicates the percentage improvement in revenue, over the List price revenue,by offering m service classes. For each of the functions $R_i(x)$, and for various values of c and d, we computed this



(a) For $R_3(x); d = 0.6; c = 0.1; \mu = 1.35; \Lambda = 1$

(b) For $R_4(x); d = 0.1; c = 0.1; \mu = 1.35, \Lambda = 1$

Figure 3: Segmentation Ratios

ratio for $m = 2, \ldots, 6$. We also computed this ratio for the priority auction, this would provide us with the best achievable ratio. In all these simulations we took $\Lambda = 1$ and restricted ourselves to the region where priority auctions are optimal. The general observations can be summarized in Figure 2.

First observe that the increase in profit per unit increase in the number of service classes goes down as the number of service classes increases. This observation is intuitive and says that the additional increase in profit decreases as the number of service classes increases. As there is a cost associated with each additional service class, a service provider would offer a finite number of classes. In most cases simulated (we had also considered other forms for the function $R(x)$), a large percentage of the achievable profits were recovered by offering a relatively smaller number of service classes (say 8 - 10 classes).

The second observation is that the ratio of interest decreases with i for $i = 1, 2, 3$. Due to section 4, as the concavity of of $xC(x)$ 'decreases', we would expect the restriction to m-service classes to hurt us to a lesser extent. This is because, as shown in the counter-example of section 4, when $xR(x)$ is not concave the optimal strategy might involve 'pooling' of different customer types. Hence we expect the ratio of interest, for a fixed value of m, to be lower as the concavity of $xC(x)$ decreases (as priority is less important here). Thus our second observation is as expected from the explanation above. We might be tempted to conclude that the 'intuition' described above holds in general; but this is not true. As already pointed out the concavity function $(xR_3(x))''$ increases with a; however the results are not as expected beyond $a = 0.8$. We had done simulations with other forms for the function $R(x)$ and this observation holds true for most instances of the problem parameters; interestingly whenever the results violate this intuition, they go in exactly the opposite direction (i.e. decreasing instead of increasing).

No of classes offered	class 1	class 2	class 3	class 4	class 5	class 6
1	2.9686					
2	3.4384	2.0106				
3	3.5662	2.9170	1.4950			
4	3.6236	3.2183	2.5108	1.1807		
5	3.6552	3.3630	2.9159	2.1945	0.9691	
6	3.6760	3.4489	3.1280	2.6577	1.9445	0.8254

Table 1: Prices Table

No of classes offered	class 1	class 2	class 3	class 4	class 5	class 6
1	0.8928					
2	0.5900	0.9580				
3	0.4300	0.7470	0.9710			
4	0.3360	0.6050	0.8160	0.9760		
5	0.2760	0.5070	0.6980	0.8540	0.9790	
6	0.2330	0.4350	0.6080	0.7550	0.8780	0.9800

Table 2: Cumulative arrival rates Table

Customer Segmentation. Here we focus on the issue of customer segmentation among the various service classes. We focus on the fraction of customers who opt for service class i or lower, when m service classes are offered. For each of the functions $R_i(x)$, and for various values of c and d, we computed this ratio for $m = 6$ and $\Lambda = 1$. Two typical scenario are shown in Figure 3; Figure 3(a) uses the function $R_3(x)$ whereas Figure 3(b) uses the function $R_4(x)$. Recall that the concavity of $xR_3(x)$ increases with a whereas the concavity of $xR_4(x)$ decreases with a; thus we do not see a definite trend with increase or decrease in concavity. However note that in both cases the change is monotone, this was observed in all the instances simulated. But another interesting phenomenon can be observed from Figure 3(b). Note that the change in the segmentation ratio is gradual until the value $a = 1$ beyond which point it changes more steeply. This is exactly the point beyond which the second derivative of $xR_4(x)$ becomes positive for high enough values of x. Note that this phenomenon does not occur in Figure 3(a). This observation suggests that as $xR(x)$ becomes non-concave (concave-convex in case of $R_4(x)$), the changes in the parameter a affect the segmentation decisions in a bigger way.

Another observation from the simulations, which is highly intuitive, is the following. Suppose that P_i^n is the price of class i service when classes of service are offered, and let z_i^k be the cumulative amount of customers opting for service class i or lower. Then in all the simulations, we have $P_i^{n+1} > P_i^n > P_{i+1}^{n+1}$ and $z_i^{n+1} < z_i^n < z_i^{n+1}$. Thus as the number of service classes increases, the prices of the service classes and the cumulative amount of customers opting for these service classes exhibit a nested property. Table 1 shows prices and Table 2 shows cumulative arrival rates for the case $R_4(x), d = 0.1, c = 0; , \mu = 1.35, \Lambda = 1$.

6 Conclusions

In this paper we studied the problem of designing a profit maximizing pricing-scheduling policy for a capacity-constrained firm with a heterogeneous customer base. The main conclusion we draw form our analysis is that under certain conditions it might be beneficial to pool customers of different characteristics together and treat them equally; this happens because customers self-select their service class. The results of Theorem 6, Segmentation Algorithm, Lemma 7 and Theorem 11 extend to the cases of M/M/1 and M/G/1 queues without preemption, and to the M/M/s system (with a single queue) with and without preemption; the analysis in these cases remains essentially the same. These results can also be extended to the preemptive M/G/1 case by observing that it is enough to consider policies in which the preemption decisions are based only on the identity of the customer types (and not on the information learned/revealed while serving). Thus the results of our model are valid in more general settings. Also the result of Lemma 8 can be extended to M/M/1 system without preemptions, thus providing an efficient procedure for solving the restricted service differentiation case.

There are various avenues for extending this work. One is to extend the analysis to the setting in which the value for obtaining service for the customers within a customer type is a random variable and none of the customer types can be served in full. This is the same setting as in Mendelson and Whang [28], and a similar analysis, without a priori fixing the scheduling policy to be the the $c\mu$ rule, might work. If, in addition, it is possible to serve some of the customer types in full, then the problem will have to be analyzed differently (as the marginal customer of some customer type i can now have a positive surplus). The same applies for the case of heterogeneous service requirements. These questions are the topic of ongoing research. Also given that increasing the capacity and increasing the number of service classes are both costly, it would be interesting to get an understanding of the situations under which one would be preferred to the other (this might depend on the existing capacity). Finally, it would be interesting to analyze the strategies of two service providers who face the same customer base and are free to offer as many service levels as needed.

Acknowledgements

We are grateful to Ward Whitt for a suggestion regarding the simulations, and to Philipp Afeche for several useful comments on an earlier draft of the paper.

References

[1] P.Afeche (2004),"Incentive-Compatible Reveue Management in Queuing Systems: Optimal Strategic Idleness and other Delaying Tactics", *Working paper*, Kellog school, Northwesetrn University.

[2] P.Afeche, H.Mendelson (2004),"Pricing and Priority Auctions in Queuing Systems with a generalized Delay Cost structure", *Management Science*, 50(7), 869 - 882.

[3] K.R.Balachandran (1972),"Purchasing Priorites in Queues", *Management Science*, 18(5), 319-326.

[4] A.Beja, E.Sid (1975),"Optimal Priority Assignment with Heterogeneous Waiting Costs", *Operations Research*, 23(1), 107 - 117.

[5] R.M.Bradford (1996),"Pricing, routing and incentive compatibility in multiserver queues", *European Journal of Operational Reserach*, 89, 226-236.

[6] R.M.Caldentey and L.M.Wein (2003),"Revenue Management of a Make-to-Stock Queue", *Working paper*, New York University, New York.

[7] E.G.Coffman Jr. and I.Mitrani (1980),"A Characterization of Waiting Time Performance Realizable by Single-Server Queues", *Operations Research*, 28(3), 810 - 821.

[8] R.W.Conway, W.L.Maxwell and L.Miller (1967), *Theory of Scheduling*, Add.-Wesley, Mass.

[9] D.Cox and W.Smith (1961), *Queues*, Methuen, London.

[10] S.Dewan and H.Mendelson (1990),"User Delay Costs and Internet Pricing for a Service Facility", *Management Science* , 36(12), 1502-1517.

[11] R.J.Dolan (1978),"Mechanisms for Priority Queuing Problems", *Bell Journal of Economics*, 9(2), 421-436.

[12] N.M.Edelson and K.Hilderbrand (1975),"Congestion tolls for Poisson queueing processes", *Econometrica*, 43, 81-92.

[13] A.Federgruen and H.Groenvelt (1988), "Characterization and Optimization of achievable performance in general queuing systems", *Operations Research*, 36(5), 733 - 741.

[14] S.B.Ghanem (1975),"Computing central optimization by a pricing priority policy", *IBM Sys. Journal*, 14, 272-292.

[15] A.Glazer and R.Hassin (1986),"Stable Priority Purchasing in Queues", *OR letters*, 4, 285-288.

[16] D.Gupta and L.Wang (2005),"Manufacturing Capacity Revenue Management", *Working paper*, University of Minnesota, Minneapolis.

[17] A.Y.Ha (2000),“Optimal Pricing that coordinates queues with customer chosen service requirements”, *Management Science*, 47(7), 915-930.

[18] M.Harris and R.M.Townsend (1981),“Resource allocation under assymetric information”, *Econometrica*, 49, 33-64.

[19] R.Hassin (1995),“Decentralized regulation of a queue”, *Management Science*, 41(1), 163- 173.

[20] R.Hassin, M.Haviv (2003),“To Queue or not to Queue”, Kluwer, Boston.

[21] L.Kleinrock(1967),“Optimum Bribing for Queue Position”,*Operations Research*, 15, 304-318.

[22] P.J.Lederer and L.Li (1997),“Pricing, Production, Scheduling and Delivery-Time Competition”, *Operations Research*, 45(3), 407-420.

[23] F.T.Lui (1985),“An Equilibrium Queuing Model of Bribery”, *J. Oof. Pol. Econ.*, 93, 760-781.

[24] C.Maglaras (2005),“Revenue management for a multi-class single-server queue”, *working paper*, Graduate School of Business, Columbia University.

[25] C.Maglaras, A.Zeevi (2005),“Pricing and Design of Differentiated Services: Approximate Analysis and Structural Insights”, *Operations Research*, 53(2), 242 - 262.

[26] M.Marchand (1974),“Priority Pricing”, *Management Science*, 20(7), 1131 - 1140.

[27] H.Mendelson(1985),“Pricing Computer Services: Queuing Effects”, ACM, 28, 312-321.

[28] H.Mendelson and S.Whang (1990),“Optimal Incentive-Compatible Priority pricing for the M/M/1 Queue”, *Operations Research*, 38(5), 870-883.

[29] R.B.Myerson (1981),“Optimal Auction Design”, *Math. of OR*, 6(1), 58 - 73.

[30] P.Naor (1969), “On the Regulation of queue size by Levying tolls”, *Econometrica*, 37(1), 15-24.

[31] E.R.Petersen and S.Rao (1993),“A Dynamic Programming Model for Assigning Customers to Priority Service classes”, School of Business Working Paper No. 93-24, Queen’s University, Kingston, Canada.

[32] E.Plambeck (2004),“Optimal Leadtime Differentiation via Diffusion Approximations”, *Operations Research*, 52(2), 213 - 228.

[33] A.Printezis, A.Burnetas (2004),“Pricing in a Single Queue with Two Customer Classes With and Without Price Discrimination”, *Working paper*, Weatherhead School of Management, Case Western reserve University.

[34] S.Rao and E.R.Petersen (1998),"Optimal Pricing of Priority Services", *Operations Research*, 46(1), 46-56.

[35] J.Shanthikumar and D.Yao (1992), "Multiclass Queuing Systems: Polymatroidal Structure and Optimal Scheduling Control", *Operations Research*, 50(1), 197 - 216.

[36] S. Stidham Jr (2002),"Analysis, design and control of Queuing systems", *Operations Research*, 50 (1), 197-216.

[37] J.A.Van Mieghem (2000),"Price and Service Discrimination in Queuing Systems: Incentive Compatibility of $Gc\mu$ Scheduling", *Management Science*, 46(9), 1249-1267.