RUTCOR
RESEARCH
REPORT

Classes, Priorities and Fairness in Queueing Systems

David Raz [a] Benjamin Avi-Itzhak [b]
Hanoch Levy [c]

RRR 21-2004, June, 2004

RUTCOR
Rutgers Center for
Operations Research
Rutgers University
640 Bartholomew Road
Piscataway, New Jersey
08854-8003
Telephone: 732-445-3804
Telefax: 732-445-5472
Email: rrr@rutcor.rutgers.edu
http://rutcor.rutgers.edu/∼rrr

[a] School of Computer Science, Tel-Aviv University, Tel-Aviv, Israel, davidraz@post.tau.ac.il

[b] RUTCOR, Rutgers, the State University of New Jersey, 640 Bartholomew Road, Piscataway, NJ 08854-8003, USA, aviitzha@rutcor.rutgers.edu

[c] School of Computer Science, Tel-Aviv University, Tel-Aviv, Israel, hanoch@cs.tau.ac.il

Rutcor Research Report
RRR 21-2004, June, 2004

Classes, Priorities and Fairness in Queueing Systems

David Raz Benjamin Avi-Itzhak Hanoch Levy

Abstract. Customer classification and prioritization are commonly used in many applications to provide queue preferential service. Their influence on queuing systems has been thoroughly studied from the delay distribution perspective. However, the fairness aspects, which are inherent to any preferential system and highly important to customers, have hardly been studied and not been quantified to date. In this work we use the Resource Allocation Queueing Fairness Measure (RAQFM) to analyze such systems and derive their relative fairness values. Results from analyzing and comparing systems with class priority to systems with no prioritization, show that assigning higher priority to short jobs often increases the system fairness, but not always. We also analyze the effect multiple servers have on fairness, showing that multiple servers increase the fairness of the system. Practitioners can use the derived results to weigh efficiency aspects versus fairness aspects in controlling their queueing systems.

1 Introduction

Queueing models have been used in a large variety of applications, including human service systems (supermarkets, airports, government offices, etc.), computer systems, and telecommunication systems, to control the service given by the system. The classification of customers/jobs based on their service requirements and the prioritization of such classes is a very common mechanism used for providing preferential service and for controlling the service in a queueing system. One common example is the priority mechanism used in computer systems where short jobs receive higher priority over long jobs. Another very common application is the queueing structure in supermarkets and other stores where short jobs (customers with a few items in hand) receive preferential service through special servers (cashiers) dedicated to them.

A major reason for using priorities and preferential service is that of fairness, that is, the wish to make the system operation "fair". Fairness among customers/jobs is a crucial and fundamental issue for queueing systems. A recent Experimental Psychology study by Rafaeli et al. [23, 24], where attitude of people in queues was studied, shows that fairness in the queue is very important to people, perhaps not less than the wait itself. In fact, our own observation is that perhaps the major reason for using a queue at all is the wish to provide fair service and fair waiting to the customers.

Despite its fundamental role in queueing systems, fairness in queues has hardly been studied (a review of the literature status is given below – Section 1.1). In particular, the issue of how priorities and preferential service affect fairness in queues has not been evaluated in a quantitative manner and is not understood. Questions such as 1) Does the provision of high priority to short jobs improve fairness, 2) To what degree, and 3) Under what conditions, have not been addressed.

The objective of this work is to use a quantitative model for studying priority and classification systems, focusing on evaluating the relative fairness of these mechanisms. Such analysis will provide measures of fairness for these systems, that can be used to quantitatively account for fairness when considering alternative designs. The quantitative approach can enhance the existing design approaches in which efficiency (e.g., utilization and delays) is accounted for quantitatively, while fairness is accounted for only in a qualitative way.

In order to focus here on the pure fairness and pure queueing properties, we will limit our discussion to systems where job classification is based only on service characteristics. Systems where classes have other attributes (such as different economic values) can be treated by weighting mechanisms, which are out of the scope of this study, and are to be dealt with in future work. To carry out our analysis we use the *Resource Allocation Queueing Fairness Measure (RAQFM)* introduced in Raz et al. [26]. The measure is based on the basic principle that all customers present in the system at epoch t deserve equal service at that epoch, and deviations from that principle result in discrimination (positive or negative); a summary statistics of these discriminations yields a measure of unfairness. A more detailed description is given in Section 2 of this paper. Using RAQFM we then study three topics.

First, in Section 3 we study general properties of prioritization mechanisms as measured by RAQFM, which apply to the systems studied in this paper as well as to a wider class of systems, under general arrival and service patterns. We show that under RAQFM, for any service policy that selects customers for service *independently* of their *service times* (that is, does not "discriminate" based on service time), the discrimination experienced by a customer is monotone non-decreasing in its service time. This means that in such systems, which "do not discriminate", an implicit discrimination is applied in favor of the long jobs and against the short jobs. This general result suggests that, from *fairness perspective*, providing preferential service to shorter jobs may be justified in many cases. We first show that the discrimination is monotone in the service time (if the service time is deterministically known), and then show that it is monotone in the service time distribution, in a stochastic dominance sense. We then deal with systems that use the preemptive priority scheduling (applying to both resume and non-resume variants), for which we show, under *general arrival and service conditions*, that the expected discrimination of the highest priority class is always positive, while the expected discrimination of the lowest priority class is always negative.

Second, we turn into more in-depth analysis of priority scheduling where we aim at addressing the issue of how fair it is to grant "full" priority to short jobs, that is to prioritize all short jobs residing in the system over *all* long jobs residing in the system. To this end we analyze single-server multiple class systems with priorities, which are common in computer systems. These systems consist of a single server, and the jobs (customers) are classified into multiple classes, where the classes may differ from each other in their service requirements and arrival rates. The service is given in these systems is that of class-priority, namely, customers of a high-priority class are always served ahead of those of a low priority class. There is a large body of literature on these systems (e.g. Avi-Itzhak and Naor [5], Avi-Itzhak [2, 3], Jaiswal [14], Kleinrock [18], Takagi [29]) where the focus in evaluating system performance is on the system expected waiting time, or in a more general framework, the mean waiting cost, under linear cost parameters varying across the classes. Optimization of the system with non-preemptive priorities, based on this performance objective, shows (e.g. Kleinrock [18, sec. 3.6], Cox and Smith [6, pp. 84-85]) that the optimal scheduling policy is to provide a higher priority to jobs with smaller mean service times (or when costs are involved, apply the μC rule). Such priority may, however, result with long jobs waiting for the completion of many short jobs who arrive behind them, and thus, possibly, to unfair treatment by the system. Thus, system operation that accounts both for efficiency and fairness, might have to resort to a different scheduling.

The unfairness of such a system is studied in Section 4, where we analyze a 2-class system, with Poisson arrivals and exponential service times. We derive the expected discrimination experienced by each class as well as the system unfairness, presented by the second moment of discrimination (since, as described in Section 2, the system expected discrimination is inherently zero, and thus the second moment serves as an unfairness measure). For comparison, we also provide the fairness analysis for an equivalent system where jobs are served in the order of arrival (FCFS). The results show that in many cases prioritization of the short jobs over the long jobs leads to higher fairness (than that of FCFS); nonetheless, in some cases FCFS is more fair. We extend the analysis to systems with more than two customer classes, at the expense of higher computational complexity or the use of some approximation.

Third, in Section 5, we turn to the analysis of multiple-server systems which are very common in human-service facilities. Again, there is large body of literature focusing on the performance of these systems (e.g. Davis [7], Kella and Yechiali [15] and many more). Our objective is to understand how the use of multiple servers, in the presence of a single queue, compares with a single server and affects the system fairness. Our analysis concentrates on a system with two classes of customers and two servers. The analysis method is extendable to a system consisting of many servers and many classes, with a common queue. We numerically compare the fairness of this system to that of a single server system. We observe that the use of multiple servers (as opposed to a single server), while maintaining the processing rate fixed, improves the system fairness (under the same conditions, the single server system is more efficient, see Kleinrock [17, Theorem 4.2]). This improvement, depending on service variability, may be quite meaningful.

Concluding remarks are given in Section 6.

1.1 Additional Related Work

The aspect of fairness associated with waiting in a queue was recognized and discussed by quite a number of authors: Palm [20] deals with judging the annoyance caused by congestion, Larson [19] in his discussion of dis-utility recognizes the central role of 'Social Justice', and Whitt [31] addresses overtaking in queues, to mention just three. However, most authors do not supply a quantitative measure for the fairness of the system. One research exception is Wang and Morris [30], where the Q-factor is proposed, that measures the performance, relative to global FCFS (namely FCFS among the customers of all sources), as observed by the customer stream treated worst, under the worst possible combination of stream loads. Fairness is thus related to classes of customers defined via a measure of extreme treatment given by the service provider with respect to customer streams. A large volume of literature also exists on weighted fair queueing (e.g. Demers et al. [9], Greenberg and Madras [11], Parekh and Gallager [21, 22], Golestani [10], Rexford et al. [27]). However, this body of work deals with fairness to streams, which fits communications, rather than fairness to jobs, which fits the models we are interested in.

Some recent studies proposing a fairness measure that fits jobs are Avi-Itzhak and Levy [4], and Wierman and Harchol-Balter [32]. In Avi-Itzhak and Levy [4] measures based on order of service, and on pair-wise comparison of waiting times, have been devised, starting with an axiomatic approach. In Wierman and Harchol-Balter [32] the slowdown, $E[T(x)/x]$, derived for all service times x, is used as a criterion for evaluating whether a system is fair or unfair. However, the slowdown is suggested as a criterion, rather than a measure of fairness. RAQFM, which inherently accounts both for seniority differences and service time differences among customers, seems to be advantageous over Avi-Itzhak and Levy [4] in handling service time differences, and on Wierman and Harchol-Balter [32] in handling seniority differences.

2 System Model and Review of RAQFM

Consider a non idling queueing system, (i.e. a queueing system where if there are n customers in the system, and the system is composed of m servers, then $\min\{m, n\}$ of these servers are operational), with m servers, indexed $1, 2, \ldots, m$. The system is subject to the arrival of a stream of customers, $C_1, C_2, \ldots$, who arrive at the system at this order. Each customer belongs to one of u classes, indexed $1, 2, \ldots, u$. An order of priorities is assigned to the classes, where lower class index means higher priority.

Let a_l and d_l denote the arrival and departure epochs of C_l respectively. Let s_l denote the service requirement (measured in time units) of C_l.

RAQFM evaluates the unfairness in the system as follows: The basic fundamental assumption is that at each epoch, all customers present in the system, deserve an *equal share* of the *total service granted* by the system at that epoch. If we let $0 \leq \omega(t) \leq m$ denote the total service rate granted at epoch t (which usually is an integer equaling the number of working servers at that epoch), and $N(t)$ denote the number of customers in the system at epoch t, then the fair share, called the momentary *warranted service* rate, is $\omega(t)/N(t)$.

Let $\sigma_l(t)$ be the momentary rate at which service is given to C_l at epoch t. This is called the momentary *granted service* rate of C_l.

The momentary discrimination rate of C_l at epoch t, when C_l is in the system, denoted $c_l(t)$, is therefore the difference between its granted service and warranted service,

$$c_l(t) = \sigma_l(t) - \frac{\omega(t)}{N(t)}. \tag{1}$$

This can be viewed as the rate at which customer discrimination accumulates for C_l at epoch t. Also let $c_l(t) \overset{def}{=} 0$ if C_l is not in the system at epoch t. C_l is not in the system at epoch t. However, as we are only interested in $c_l(t)$ when C_l is in the system, and for the omitted. Also note that in principle, the above equation might also apply in systems that that are not with these systems, and therefore they will be discussed in a future paper.

The total discrimination of C_l, denoted D_l, is

$$D_l = \int_{a_l}^{d_l} c_l(t)dt. \tag{2}$$

Remark 2.1 *(An alternative definition of the momentary warranted service and discrimination).* The definition of the momentary warranted service (and thus discrimination) given above is based on the concept that a customer deserves an equal share of the *resources granted* by the system at that epoch ($\omega(t)$) and any deviation from it creates discrimination among the customers residing in the system. If some of the resources are not granted at epoch t, e.g., due to system idling, or due to the use of only part of the servers, it may be considered as being *inefficient* but not as a discrimination and unfairness.

One could consider an alternative concept by which at epoch t a customer deserves an equal share of *all the available system resources.* Under the notation given above this means that the warranted service will be defined as $m/N(t)$ (instead of $\omega(t)/N(t)$), and that the momentary discrimination given in (1) will be replaced by $c_l(t) = \sigma(t) - m/N(t)$.

The difference between the two alternatives is conceptual and relates to situations where the system does not grant all of its resources. One such case is a multi-server system at epochs where the number of customers is smaller than the number of servers, $N(t) < m$. Another case is a system which allows server idling (when there are customers in the system).

This issue and the tradeoff between the alternatives is more pronounced in multi-server multi-queue systems, and thus is discussed in depth in a study that focuses on these systems ([25]). For this work we choose to focus on the concept of fair division of the *granted* resources (Equation (1)); this might be appealing since the cases in this paper where the system does not grant all resources are limited to situations resulting from system operations constraints (system cannot serve a single customer by many servers), and thus may possibly be interpreted by customers as non discriminatory.

For work conserving systems (defined as systems in which the total service given to a customer over time equals its service requirement, i.e. $\int_0^\infty \sigma_l(t) = s_l$), we have from (1) and (2)

$$D_l = s_l - \int_{a_l}^{d_l} \frac{\omega(t)}{N(t)} dt. \tag{3}$$

As shown in Raz et al. [26] for a single server, work conserving and non idling (WCNI) system, the expected value of discrimination always obeys $E[D] = 0$. Thus, according to RAQFM, the unfairness of the system is defined as the second moment of the discrimination, namely $E[D^2]$. The same property will be shown to hold in multiple server non idling systems.

In the analysis (in Section 4 and onward) we will consider a WCNI $M/M/m$ type system, with u classes of customers, where class j arrivals follow a Poisson process with rate λ_j, and their required service times are i.i.d. exponentially with mean $1/\mu_j$, $j = 1, 2, \ldots, u$. The total arrival rate is denoted by $\lambda \overset{def}{=} \sum_{j=1}^{u} \lambda_j$ and, for stability, it is assumed that $\rho \overset{def}{=} \sum_{j=1}^{u} \lambda_j/\mu_j < m$.

To facilitate the mathematical analysis, arrival and departure epochs are labeled event epochs, and time is viewed as being slotted by these event epochs. The i-th time slot, of duration $T_i, i = 1, 2, \ldots,$ is bounded by the $(i-1)$-th and the i-th event epochs.

We limit the analysis to systems where a service decision is made only on arrival and departure epochs. Thus, the number of working servers, and the rate of service given to each customer, are constant during each slot. We define $0 \le \omega_i \le u$ as the number of working servers in the i-th slot, $\sigma_{i,l}$ as the rate at which service is given to C_l at the i-th slot, and N_i as the number of customers in the system during the i-th slot. Using these, $c_{i,l}$, the momentary discrimination of C_l at the i-th slot, which is the rate at which customer discrimination accumulates for C_l at this slot, is

$$c_{i,l} = \sigma_{i,l} - \frac{\omega_i}{N_i}. \tag{4}$$

In this formulation we modify a_l, d_l to denote the indexes of the arrival and departure slots of C_l respectively (C_l arrives at the beginning of the a_l-th slot and departs at the end of the d_l-th slot).

The total discrimination accumulated for C_l during the i-th slot is $c_{i,l}T_i$. Thus, the slotted version of (2) is

$$D_l = \sum_{i=a_l}^{d_l} c_{i,l}T_i,$$

and for work conserving systems

$$D_l = s_l - \sum_{i=a_l}^{d_l} \frac{T_i\omega_i}{N_i}.$$

We define the *Preemptive Priority* class of scheduling policies. In this class of scheduling policies the server always serves the customer with the highest priority present in the system. If a higher priority customer arrives, and finds a lower priority customer in service, the served customer is displaced by the arriving customer. The order of service within each class of customers is usually FCFS. In the *Preemptive Resume* variant, a specific policy analyzed in Section 4.2 and Section 4.3, the preempted customer returns to the head of the queue of its class, and resumes its service from the point it was interrupted, upon reentering service. For discussion of this, and other variants, see Takagi [29, sec. 3.4].

3 General Properties of RAQFM

This section presents several properties of RAQFM, mostly dealing with discrimination and unfairness in multiple class systems. Here we deal with a general and arbitrary arrival pattern and service times.

Note that the measure used throughout this paper is the one described in the Section 2 and not the alternative measure mentioned in Remark 2.1. For the alternative measure, other properties can be proven, and we leave this for future work.

Theorem 3.1. *In a stationary non idling system, the expected value of discrimination always obeys $E[D] = 0$.*

Proof. Observe that the total momentary discrimination rate at any epoch t is

$$\sum_{\{l|a_l<t<d_l\}} c_l(t) = \sum_{\{l|a_l<t<d_l\}} \left(\sigma_l(t) - \frac{\omega(t)}{N(t)}\right) = \omega(t) - N(t)\frac{\omega(t)}{N(t)} = 0, \qquad (5)$$

where the first equality is from the definition in (1) and the second is due to the fact that the sum of the service rates given to all the customers, equals the total rate of service given, and the fact that there are exactly $N(t)$ customers in the system at epoch t (i.e. exactly

$N(t)$ customers satisfy $a_l < t < d_l$). We have

$$E[D] = \lim_{l\to\infty} \frac{1}{l} \sum_{k=1}^{l} D_l = \lim_{l\to\infty} \frac{1}{l} \sum_{k=1}^{l} \int_{a_l}^{d_l} c_l(t)dt = \lim_{l\to\infty} \frac{1}{l} \int_{0}^{\infty} \sum_{\{l|a_l<t<d_l\}} c_l(t)dt = 0,$$

where the first equality is due to ergodicity assumption, the third equality is due to changing the order of summation, and the forth equality is due to (5). □

Note that this theorem applies to work conserving system, as well as systems which are not.

Definition 3.1 (Stochastic Dominance Between Random Variables). Consider non negative random variables X_1, X_2 whose distributions are $F_{X_1}(t) = Pr[X_1 \le t], F_{X_2}(t) = Pr[X_2 \le t]$. We say that X_1 *stochastically dominates* X_2, denoted $X_1 \succ X_2$, if $F_{X_1}(t) \le F_{X_2}(t) \quad \forall t \ge 0$.

Theorem 3.2. *Let C_l be a customer with service requirement s_l. Consider a $G/G/m$ system under non-preemptive service policy, where the service decision is independent of the service times. Let $D_l^{(s_l)}$ be a random variable denoting the discrimination of C_l, when it arrives at the system in steady state. Then $D_l^{(s_l)}$ is monotone non-decreasing in s_l, namely if $s_l' > s_l$ then $D_l^{(s_l')} \succ D_l^{(s_l)}$.*

Proof. Consider service times $s_l, s_l', \quad s_l' > s_l$. Observe a customer C_l. Under any non-preemptive service policy, C_l waits until epoch q_l, when it enters service, and stays in service until its departure. (2) can thus be written as

$$D_l = \int_{a_l}^{q_l} c_l(t)dt + \int_{q_l}^{d_l} c_l(t)dt. \tag{6}$$

The first term in this sum is independent of the service requirement. The second term is an integral of $c_l(t)$, over the interval (q_l, d_l). Observe that the length of this interval is the service time of C_l.

To prove the monotonicity we consider a specific sample path π and compare the values of $D_l^{(s_l)}$ and $D_l^{(s_l')}$ for this path, denoted by $D_{l,\pi}^{(s_l)}$ and $D_{l,\pi}^{(s_l')}$. From (6) we have

$$D_{l,\pi}^{(s_l')} - D_{l,\pi}^{(s_l)} = \int_{q_l}^{q_l+s'} c_l(t)dt - \int_{q_l}^{q_l+s} c_l(t)dt = \int_{q_l+s}^{q_l+s'} c_l(t)dt \ge 0, \tag{7}$$

where the last inequality is due to $c_l(t) \ge 0$, which is obvious from (1). Since (7) holds for every sample path π, the proof follows. □

Theorem 3.3. *Let S_l and S_l' be random variables representing two alternate service times of C_l and let $F_{S_l}(t)$ and $F_{S_l'}(t)$ be their distribution functions. Consider a $G/G/m$ system under non-preemptive service policy, where the service decision is independent of the service times. If $S_l' \succ S_l$ then $D^{(S_l)} \succ D^{(S_l')}$.*

Proof. The proof follows by applying Theorem 3.2 for the whole range of service times, and using the fact that $S'_l \succ S_l$. □

Corollary 3.1. *Consider a $G/G/m$ system under non-preemptive service policy, where the service decision is independent of the service times. Let $D^{(S_l)}$ be a random variable denoting the discrimination of C_l, when it arrives at the system in steady state, given its service time S_l (random variable). Then $E[D^{(S_l)}]$ is monotonically non-decreasing in S_l. That is, if $S'_l \succ S_l$ then $E[D^{(S'_l)}] \geq E[D^{(S_l)}]$.*

Remark 3.1. Using the same arguments it can be shown that Theorem 3.2 also holds in the case of a preemptive system, providing that the preemption of a customer with service $s' > s$, during the period at which it receives the first s units of service, is unchanged, i.e. preemptions are not determined by the length of the service required by the customer. In a similar manner Theorem 3.3 will hold as well.

Theorem 3.4. *Let δ_p denote the expected discrimination of class p customers. In a $G/G/m$ system, with u classes, if the scheduling policy belongs to the class of preemptive priority scheduling policies, then $\delta_1 \geq 0$ and $\delta_u \leq 0$.*

Proof. Let $r_{p,k}$ be the index of the k-th class p customer to arrive at the system.

$$\delta_p = \lim_{l\to\infty} \frac{1}{l} \sum_{k=1}^{l} D_{r_{p,k}} = \lim_{l\to\infty} \frac{1}{l} \sum_{k=1}^{l} \int_{a_{r_{p,k}}}^{d_{r_{p,k}}} c_{r_{p,k}}(t)dt = \lim_{l\to\infty} \frac{1}{l} \int_{a_1}^{\max_k d_{r_{p,k}}} \sum_{k=1}^{l} c_{r_{p,k}}(t)dt,$$

where the first equality is due to ergodicity assumption, the second equality is due to (2) and the third one is due to changing the order of summation.

Let $c^p(t)$ be the sum of momentary discriminations of class p customers in the system at epoch t, which equals

$$c^p(t) = \sum_{\{l | a_l < t < d_l, C_l \text{ is of class p}\}} c_l(t).$$

Then

$$\delta_p = \lim_{\tau\to\infty} \frac{1}{l_p(\tau)} \int_{a_1}^{\tau} c^p(t)dt, \tag{8}$$

where $l_p(\tau)$ is the number of class p customers to leave the system before the epoch τ.

Let $N_p(t)$ be the number of class p customers in the system at epoch t. As the scheduling policy belongs to the class of preemptive priority scheduling policies, if $N_1(t) \leq m$, then all $N_1(t)$ customers are served at epoch t. Otherwise, m out of them are served. Thus

$$c^1(t) = \begin{cases} N_1(t) - \frac{\omega(t)N_1(t)}{N(t)} & N_1(t) \leq m \\ m - \frac{mN_1(t)}{N(t)} & N_1(t) > m \end{cases} = \begin{cases} N_1(t)\left(1 - \frac{\omega(t)}{N(t)}\right) & N_1(t) \leq m \\ m\left(1 - \frac{N_1(t)}{N(t)}\right) & N_1(t) > m \end{cases}, \tag{9}$$

which is greater or equal to zero since $\omega(t) \leq N(t)$ and $N_1(t) \leq N(t)$. Thus, $c^1(t) \geq 0, \forall t$, and from (8), $\delta_1 \geq 0$.

Note that (9) also provides the only epochs in which $c^1(t) = 0$, namely when either $N_1(t) = N(t)$ (all the customers in the system are of class 1), $N(t) < m$ (there are less than m customers in the system), or $N_1(t) = 0$. In fact, for every class p, $c^p(t) = 0$ when either $N_p(t) = N(t)$, $N(t) < m$, or $N_p(t) = 0$.

As for $c^u(t)$, it equals zero when either $N_u(t) = N(t)$, $N(t) < m$, or $N_u(t) = 0$. Otherwise there are two cases, either $N(t) - N_u(t) \geq m$ or $N(t) - N_u(t) < m$. In the first case there are more than m customers of higher priority in the system, and thus no class u customer is being served. Therefore, $c^u(t) = -N_u(t)m/N(t)$ which is negative. In the second case there are some class u customers being served. In this case let $\omega_u(t)$ be the number of class u customers served at epoch t. Using this notation

$$c^u(t) = \omega_u(t) - \frac{N_u(t)m}{N(t)} = \frac{\omega_u(t)N(t) - N_u(t)m}{N(t)}. \tag{10}$$

To prove that this value is negative, let $N'(t) = N(t) - m$ denote the number of customers waiting at epoch t, all of whom must be of class u. We can write $N(t) = m + N'(t)$, $N_u(t) = \omega_u(t) + N'(t)$. Substituting into (10) yields

$$c^u(t) = \frac{\omega_u(t)(m + N'(t)) - (\omega_u(t) + N'(t))m}{N(t)} = \frac{(\omega_u(t) - m)N'(t)}{N(t)} < 0,$$

since $\omega_u(t) < m$. Thus, $c^u(t) \leq 0$, and from (8), $\delta_u \geq 0$. □

Theorem 3.5. *In a $G/G/m$ system, with 2 classes, where the scheduling policy belongs to the class of preemptive priority scheduling policies, if the mean service time $1/\mu_1 \to 0$ then $\delta_1 \to 0, \delta_2 \to 0$.*

Proof. According to Theorem 3.4, $\delta_1 \geq 0$. Observe that in (3) the only positive part is s_l and thus $D_l < s_l$. For class 1 customers this yields $D_l \leq 1/\mu_1$. As the expected value cannot be larger than all the individual values, $\delta_1 \leq 1/\mu_1$, leading to $\delta_1 \to 0$. As Theorem 3.1 implies $\delta_1 + \delta_2 = 0$, this also means $\delta_2 \to 0$. □

Remark 3.2. $\delta_1 \to 0$ since $1/\mu_1 \to 0$ and thus class-1 jobs are of infinitesimal size, and the discrimination of these customers approaches zero. Class-2 customers, in contrast, are of finite size, and thus their discrimination approaches that in the single class system. This means $\delta_2 \to 0$ (as is in all single class systems, see Theorem 3.1). However, the second moment $\delta_2^{(2)}$ may be positive due to discrimination between class-2 customers and themselves (see Figure 5)

4 Single Server Systems With Multiple Customer Classes

As mentioned in the introduction, a common way to give preferential service to prioritized customers is to serve them ahead of unprioritized ones. This can be done in various ways e.g. assigning them to a special queue, served ahead of the other queues. Our interest is in evaluating how fair such a schedule is under a wide variety of conditions, and how it compares to the alternative of a simple First Come First Served (FCFS) schedule. Thus, we analyze the unfairness of the FCFS service policy, and that of the Preemptive Resume service policy, described in Section 2. We dub the later "Priority Scheduling" for the purpose of this section.

One practical way to implement the preemptive resume service policy (which is also easy to analyze) is to assign each class of customers to its own service queue. The server always serves the nonempty queue with the highest priority first, using the FCFS service policy within that queue, until the queue is empty. Preempted customers are returned to the head of their respective queues.

As we wish to focus on the pure fairness issues inherent in the system, we limit our discussion to systems where job classification is based only on service characteristics. The analysis starts with the two class case, $u = 2$.

4.1 FCFS Scheduling for Two Customer Classes in a WCNI M/M/1 System

As mentioned in Section 2, the time between the arrival of a customer and its departure, is slotted by arrivals and departures of other customers. Assume a class j customer, $j = 1, 2$, is served in a given slot. The first two moments of that slot's duration, $t_j^{(1)}$ and $t_j^{(2)}$, are

$$t_j^{(1)} = \frac{1}{\lambda + \mu_j}, \qquad t_j^{(2)} = \frac{2}{(\lambda + \mu_j)^2} = 2(t_j^{(1)})^2. \tag{11}$$

The probability that a slot in which a class j customer is served, ends with an arrival of a class k customer, is denoted $\tilde{\lambda}_{j,k}$. The probability that a slot in which a class j customer is being served, ends with an arrival of any customer, is denoted $\tilde{\lambda}_j$. The probability that a slot where a class j customer is being served, ends with the departure of the same customer, is denoted $\tilde{\mu}_j$. We have:

$$\tilde{\lambda}_{j,k} = \frac{\lambda_k}{\lambda + \mu_j}, \qquad \tilde{\lambda}_j = \frac{\lambda}{\lambda + \mu_j}, \qquad \tilde{\mu}_j = \frac{\mu_j}{\lambda + \mu_j}. \tag{12}$$

Consider an arbitrary tagged customer of class j, denoted C. Let a be the number of customers ahead of C in the queue, and let b be the number of customers behind C. Let s be the class of the customer now being served. Due to the memoryless properties of the system, the state (a, b, s), denoted $\mathcal{S}_{a,b,s}$, captures all that is needed to predict the future discrimination of C. This is the state *observed* by C, and for brevity, we say in this case that *C is in state $S_{a,b,s}$*.

From (4), the momentary discrimination during a slot where C is in state $\mathcal{S}_{a,b,s}$, denoted $c(a, b)$, since it does not depend on s, is

$$c(a,b) = \begin{cases} -\frac{1}{a+b+1} & a > 0 \\ 1 - \frac{1}{b+1} & a = 0 \end{cases}.$$

Let $E[D_j|k,s]$ denote the expected value of discrimination of a class j customer, given that the customer sees k customers on arrival (including the one being served), and the one being served is of class s. In the event that $k=0$, let $E[D_j|k=0,s=0]$ denote the expected value of discrimination when the customer finds an empty system. We will use this convention throughout this section, i.e. when the system is empty, we will consider the customer class being served to be 0. For completeness, $E[D_j|k=0,s\neq 0] \overset{def}{=} 0, E[D_j|k>0,s=0] \overset{def}{=} 0$.

Let $P_{k,s}$ be the steady state probability that there are k customers in the system, and the customer in service is of class s, (or $s=0$ if no customer is in service). According to the Poisson Arrivals Sees Time Averages (PASTA) property (see Wolff [33]), this is also the probability that an arbitrary arrival encounters in this state. Thus, the first two moments of D_j follow

$$\begin{aligned} E[D_j] &= \sum_{k=0}^{\infty}\sum_{s=0}^{2} E[D_j|k,s]P_{k,s}, \\ E[D_j^2] &= \sum_{k=0}^{\infty}\sum_{s=0}^{2} E[D_j^2|k,s]P_{k,s}. \end{aligned}$$

The first two moments of D follow

$$E[D] = \frac{1}{\lambda}\left(\lambda_1 E[D_1] + \lambda_2 E[D_2]\right) = 0 \tag{13}$$

$$E[D^2] = \frac{1}{\lambda}\left(\lambda_1 E[D_1^2] + \lambda_2 E[D_2^2]\right), \tag{14}$$

where the equality to zero in (13) results from Theorem 3.1.

To calculate $P_{n,s}$, recall the alternate representation of the system, in which when a customer leaves the system, the next customer to be served is chosen, and it is of class s with probability $p_s = \lambda_s/\lambda, s=1,2$. Therefore (see Figure 1) the balance equations are:

$$\begin{aligned} (\lambda+\mu_s)P_{n,s} &= \lambda P_{n-1,s} + \mu_1 p_s P_{n+1,1} + \mu_2 p_s P_{n+1,2} \quad n>1 \\ (\lambda+\mu_s)P_{n,s} &= \lambda_s P_{0,0} + \mu_1 p_s P_{n+1,1} + \mu_2 p_s P_{n+1,2} \quad n=1 \\ \lambda P_{0,0} &= \mu_1 P_{1,1} + \mu_2 P_{1,2} \\ \sum_{i=1}^{\infty}\sum_{s=1}^{2} P_{i,s} &+ P_{0,0} = 1, \end{aligned} \tag{15}$$

and can be calculated numerically to any required accuracy.

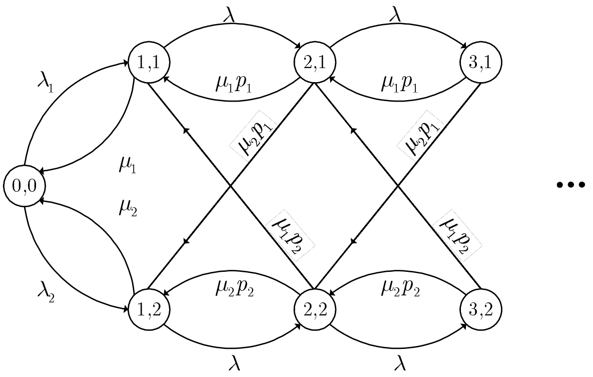

Figure 1: Single Server FCFS with Two Customer Classes: State Diagram

Remark 4.1. Steady state probabilities of multiple class systems were recently studied in Sleptchenko [28], Harten and Sleptchenko [12], Harten et al. [13] and more. It is not the purpose of this paper to introduce additional results, nor to demonstrate alternative methods to achieve the same results. The reader may use the results in Sleptchenko [28], Harten and Sleptchenko [12], Harten et al. [13] as appropriate. Alternately, one may possibly want to use the Laplace-Stieltjes transforms derived for $M/M/c$ and $M/G/1$ Vacations, with priorities, see Kella and Yechiali [15, 16].

Let $D_j(a,b,s)$ be a random variable, denoting the discrimination experienced by a class j customer, through a walk starting at $\mathcal{S}_{a,b,s}$, and ending at its departure. Then

$$E[D_j|k,s] = \begin{cases} E[D_j(k,0,s)] & k>0 \\ E[D_j(0,0,j)] & k=0 \end{cases}$$

$$E[D_j^2|k,s] = \begin{cases} E[D_j^2(k,0,s)] & k>0 \\ E[D_j^2(0,0,j)] & k=0 \end{cases}.$$

Let $d_j(a,b,s)$ and $d_j^{(2)}(a,b,s)$ be the first two moments of $D_j(a,b,s)$.

Assume customer C, of class j, is in $\mathcal{S}_{a,b,s}$ at slot i. The slot length is exponentially distributed with first two moments $t_s^{(1)}$ and $t_s^{(2)}$. At the slot end, the system will encounter one of the several possible events and C's state will change accordingly. For $a>1$ the possible events and state transitions are:

1. A customer arrives at the system. The probability of this event is $\tilde{\lambda}_s$. C's state changes to $\mathcal{S}_{a,b+1,s}$.

2. A customer leaves the system, and the next customer to be served is of class 1. The probability of this event is $\tilde{\mu}_s p_1$. C's state changes to $\mathcal{S}_{a-1,b,1}$.

3. A customer leaves the system, and the next customer to be served is of class 2. The probability of this event is $\tilde{\mu}_s p_2$. C's state changes to $\mathcal{S}_{a-1,b,2}$.

For $a \leq 1$ the possible events are:

4. A customer arrives at the system. The probability of this event is $\tilde{\lambda}_s$. C's state changes to $\mathcal{S}_{a,b+1,s}$.

5. A customer leaves the system, The probability of this event is $\tilde{\mu}_s$. If C is being served ($a = 0$), C leaves the system. If C is at the head of the queue ($a = 1$), C's state changes to $\mathcal{S}_{0,b,j}$.

This leads to the recursive expressions

$$d_j(a,b,s) = \begin{cases} t_s^{(1)}c(a,b) + \tilde{\lambda}_s d_j(a,b+1,s) + \tilde{\mu}_s p_1 d_j(a-1,b,1) + \tilde{\mu}_s p_2 d_j(a-1,b,2) & a > 1 \\ t_s^{(1)}c(a,b) + \tilde{\lambda}_s d_j(a,b+1,s) + \tilde{\mu}_s d_j(a-1,b,j) & a = 1 \\ t_s^{(1)}c(a,b) + \tilde{\lambda}_s d_j(a,b+1,s) & a = 0 \end{cases}, \tag{16}$$

$$d_j^{(2)}(a,b,s) = \begin{cases} t_s^{(2)}(c(a,b)^2 + \tilde{\lambda}_s d_j^{(2)}(a,b+1,s) + \tilde{\mu}_s p_1 d_j^{(2)}(a-1,b,1) + \tilde{\mu}_s p_2 d_j^{(2)}(a-1,b,2) \\ \quad +2t_s^{(1)}c(a,b)\left(\tilde{\lambda}_s d_j(a,b+1,s) + \tilde{\mu}_s p_1 d_j(a-1,b,1) + \tilde{\mu}_s p_2 d_j(a-1,b,2)\right) & a > 1 \\ t_s^{(2)}(c(a,b))^2 + \tilde{\lambda}_s d_j^{(2)}(a,b+1,s) + \tilde{\mu}_s d_j^{(2)}(a-1,b,j) \\ \quad +2t_s^{(1)}c(a,b)\left(\tilde{\lambda}_s d_j(a,b+1,s) + \tilde{\mu}_s d_j(a-1,b,j)\right) & a = 1 \\ t_s^{(2)}(c(a,b))^2 + \tilde{\lambda}_s d_j^{(2)}(a,b+1,s) \\ \quad +2t_s^{(1)}c(a,b)\tilde{\lambda}_s d_j(a,b+1,s) & a = 0 \end{cases} \cdot \tag{17}$$

4.1.1 Computational Aspects

(15) provides a set of linear equations for calculating $P_{n,s}$. To solve it, one first needs to decide on a maximum relevant number of customers that can be seen on arrival, say K. This leads to a sparse set of $2K+1$ linear equations. See, for example, Anderson et al. [1], Davis [8], for efficient methods of solving such a set.

(16) and (17) provide a recursive method for calculating $d_j(a,b,s)$ and $d_j^2(a,b,s)$. Specifically, if one sets $d_j(a,b,s) = d_j^{(2)}(a,b,s) = 0$ for $a > K$ or $b > K$, one can start with $b = K, a = 0$, calculating $d_j(a,b,s)$ for $a = 1, 2, \ldots, K$, then moving on to $b = K-1, a = 0$ and so on. After the calculation of $d_j(a,b,s)$ is completed, one can do the same for $d_j^2(a,b,s)$. This leads to a time complexity of $O(K^2)$. Since the computation can be done in a column by column fashion, each column depending only on the column preceding it, the space complexity is linear in K.

Our experience shows that choosing $K = 30$ for $\rho = 0.8$, yields values sufficiently close to those achieved through simulation.

Remark 4.2 *(The size of K).* Since p_n, the steady state probability of having n customers in the system, is $p_n = (1-\rho)\rho^n$, and P_K, the steady state probability of having K or less customers in the system is $P_K = \sum_{n=0}^{K} p_n = 1 - \rho^{K+1}$, then the probability of having more than K customers in the system is $1 - P_K = \rho^{K+1}$. The selection of K may be determined by the value one wants to allow for ρ^{K+1}. Suppose $\rho^{K+1} = 10^{-3} \Rightarrow K = -3/\log\rho - 1$, which for $\rho = 0.8$ yields $K = 30$.

4.1.2 System Evaluation: Numerical Results

In this section we present unfairness properties of the system for a variety of parameter sets. Of particular interest is to examine these properties as functions of the service difference between the customer classes, expressed by the mean service time ratio μ_1/μ_2. As demonstrated in Raz et al. [26], the unfairness of the system is sensitive to the utilization ρ. Therefore, we maintain a constant $\rho = 0.8$, independently of μ_1/μ_2. For simplicity, the evaluation is done for equal arrival rates of $\lambda_1 = \lambda_2 = 0.1$.

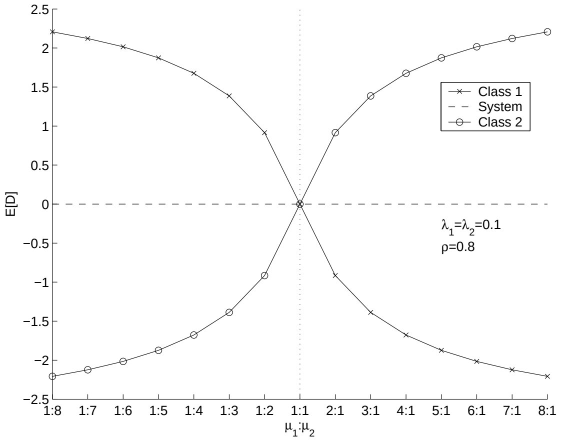

Figure 2: Expected Discrimination in a Single Server FCFS system with Two Customer Classes

Figure 2 depicts $E[D]$, for the two classes of customers, as well as for the entire system. We note the following properties:

1. The image is symmetric around $\mu_1 = \mu_2$ (a dotted line), since the classes are interchangeable.

2. $E[D] = 0$, as expected from Theorem 3.1.

3. For $\mu_1 \neq \mu_2$, the class with larger mean service requirement (smaller μ), is positively discriminated, at the expense of the other class. This is due to Corollary 3.2. Note that this discrimination is an implicit discrimination inherent in the FCFS schedule of this system.

4. The discrimination is monotone-increasing with the service requirement, as expected from Theorem 3.2.

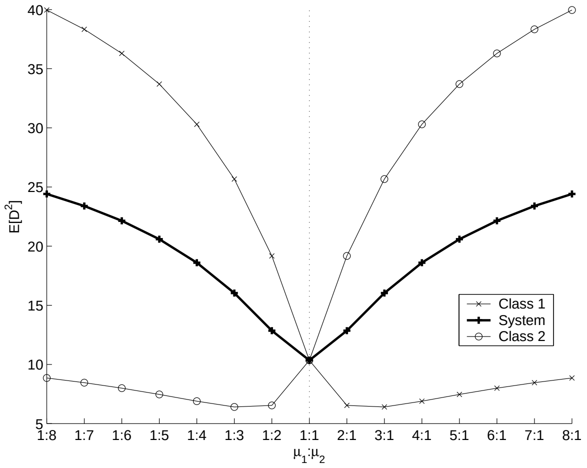

Figure 3: Unfairness in a Single Server FCFS system with Two Customer Classes

Figure 3 depicts the unfairness, measured by the second moment of the discrimination, for the two classes, and for the system. We observe the following properties:

1. The unfairness observed by a specific class of customers, is largest when that class has a large service requirement. A similar property was observed in Raz et al. [26] for a single class $M/M/1$, and agrees with the the intuition that larger mean service times result in larger unfairness.

2. The system unfairness increases as the ratio between the service requirements increases. There are two reasons for this phenomenon. The first was observed in Figure 2, namely that the difference between the expected discriminations of the classes increases (an increase in “inter-class” unfairness). The second was observed in the previous item, namely that there are increasing differences between the discriminations of customers within the same class (an increase in “intra-class” unfairness).

One can conclude from this the following conjecture:

Conjecture 4.1. *The system unfairness of a FCFS queueing system, increases with the variability of the service requirements.*

4.2 Priority Scheduling for Two Customer Classes

The discipline analyzed here is the preemptive resume one, in which class 1 gets preemptive priority over class 2. Within each queue, the service order follows FCFS.

Consider an arbitrary tagged customer of class j, denoted C. Let a_i be the number of class i customers ahead of C in the queue. Note that for $j = 1$, $a_2 = 0$, due to the priority class 1 gets over class 2. Let b be the number of customers behind C. Note that for $j = 1$ this includes both class 1 customers behind C in its queue and all class 2 customers in the system, while for $j = 2$ it only includes class 2 customers behind C. Due to the memoryless properties of the system, The state (a_1, a_2, b), denoted $\mathcal{S}_{a_1,a_2,b}$, captures all that is needed to predict the future discrimination of C.

From (4), the momentary discrimination during a slot where C is in state $S_{a_1,a_2,b}$ denoted $c_j(a_1, a_2, b)$, is

$$c_1(a_1, a_2, b) = \begin{cases} -\frac{1}{a_1+b+1} & a_1 > 0 \\ 1 - \frac{1}{b+1} & a_1 = 0 \end{cases}$$
$$c_2(a_1, a_2, b) = \begin{cases} -\frac{1}{a_1+a_2+b+1} & a_1 > 0 \text{ or } a_2 > 0 \\ 1 - \frac{1}{b+1} & a_1, a_2 = 0 \end{cases}.$$

Let $E[D_j|k_1, k_2]$, $j = 1, 2$, denote the expected value of discrimination of a class j customer, given that the customer sees k_1 customers of class 1, and k_2 customers of class 2 on arrival.

Let P_{k_1,k_2} be the steady state probability that there are k_1 customers of class 1 and k_2 customers of class 2 in the system. The first two moments of D_j are given as

$$E[D_j] = \sum_{k_1=0}^{\infty} \sum_{k_2=0}^{\infty} E[D_j|k_1, k_2] P_{k_1,k_2}$$
$$E[D_j^2] = \sum_{k_1=0}^{\infty} \sum_{k_2=0}^{\infty} E[D_j^2|k_1, k_2] P_{k_1,k_2}.$$

The first two moments of $E[D]$ are the same as in (13) and (14).

The balance equations for $P_{i,j}$ are

$$\begin{cases} (\lambda_1 + \lambda_2 + \mu_1) P_{i,j} = \lambda_1 P_{i-1,j} + \lambda_2 P_{i,j-1} + \mu_1 P_{i+1,j} & i, j > 0 \\ (\lambda_1 + \lambda_2 + \mu_1) P_{i,j} = \lambda_1 P_{i-1,j} + \mu_1 P_{i+1,j} & i > 0, j = 0 \\ (\lambda_1 + \lambda_2 + \mu_2) P_{i,j} = \lambda_2 P_{i,j-1} + \mu_1 P_{i+1,j} + \mu_2 P_{i,j+1} & i = 0, j > 0 \\ (\lambda_1 + \lambda_2) P_{i,j} = \mu_1 P_{i+1,j} + \mu_2 P_{i,j+1} & i, j = 0, \end{cases} \tag{18}$$

$$\sum_{i=0}^{\infty}\sum_{j=0}^{\infty} P_{i,j} = 1.$$

$P_{i,j}$ can be calculated numerically to any required accuracy. See Remark 4.1 for additional work on this subject.

Let $D_j(a_1, a_2, b)$ be a random variable, denoting the discrimination experienced by a class j customer, through a walk starting at $\mathcal{S}_{a_1,a_2,b}$, and ending at its departure. Then

$$\begin{aligned} E[D_1|k_1, k_2] &= E[D_1(k_1, 0, k_2)] \\ E[D_2|k_1, k_2] &= E[D_2(k_1, k_2, 0)]. \end{aligned}$$

Let $d_j(a_1, a_2, b)$ and $d_j^{(2)}(a_1, a_2, b)$ be the first two moments of $D_j(a_1, a_2, b)$.

We first analyze $d_1(a_1, a_2, b)$. Note that a class 1 customer sees the system as a single class FCFS queue. As the customer arrives, all class 1 customers in the system are in front of him in the queue, and class 2 customers are behind him (as opposed to a single class system where all customers are in front of an arriving customer). However, from that moment on, the types of customers behind the customer do not matter to the calculation, and thus the system behaves like a queue with an arrival rate of λ. All departures are of class 1 customers, and therefore the departure rate is μ_1.

The single class FCFS system was analyzed in Raz et al. [26]. Using the notation in that paper, $\mathcal{S}_{a,b}$ represents the state where there are a customers ahead of C and b customers behind him. Thus, as analyzed there, a customer arriving into a single class FCFS system, seeing k customers in the system on arrival, starts his walk at $\mathcal{S}_{k,0}$. Similarity, a class 1 customer in a system with two classes, seeing k_1 customers of class 1, and k_2 customers of class 2 on arrival, starts his walk at $\mathcal{S}_{k_1,k_2}$. Thus, we resort to the analysis of the single class FCFS system.

Using the same notation as in Raz et al. [26], let $D(a, b)$ be a random variable, denoting the discrimination experienced by a customer, through a walk starting at $\mathcal{S}_{a,b}$, and ending at its departure, with first two moments $d(a, b)$ and $d^{(2)}(a, b)$. This leads to

$$\begin{aligned} E[D_1|k_1, k_2] &= E[D(k_1, k_2)] = d(k_1, k_2) \\ E[D_1^2|k_1, k_2] &= E[D(k_1, k_2)^2] = d^{(2)}(k_1, k_2). \end{aligned}$$

Expressions for deriving $d(a, b)$ and $d^{(2)}(a, b)$ were given in Raz et al. [26], and are provided for completeness in Appendix A.

For a customer of class 2, assume C is in state $\mathcal{S}_{a_1,a_2,b}$ at slot i. Let s be the type of customer being served in this slot. Then

$$s = \begin{cases} 1 & a_1 > 0 \\ 2 & a_1 = 0 \end{cases},$$

i.e. s is implied directly from a_1 and thus does not have to be accounted for in the state.

The slot length is exponentially distributed with first two moments $t_s^{(1)}$ and $t_s^{(2)}$. At the slot end, the system will encounter one of the following events and C's state will change accordingly:

1. A class 1 customer arrives at the system. The probability of this event is $\tilde{\lambda}_{s,1}$. C's state changes to $\mathcal{S}_{a_1+1,a_2,b}$.

2. A class 2 customer arrives at the system. The probability of this event is $\tilde{\lambda}_{s,2}$. C's state changes to $\mathcal{S}_{a_1,a_2,b+1}$.

3. A customer leaves the system. The probability of this event is $\tilde{\mu}_s$. If C is being served $(a_1 = 0, a_2 = 0)$, C leaves the system. Otherwise, if $a_1 > 0$, C's state changes to $\mathcal{S}_{a_1-1,a_2,b}$, and if $a_1 = 0$, C's state changes to $\mathcal{S}_{0,a_2-1,b}$.

This leads to the following recursive expressions

$$
\begin{aligned}
&d_2(a_1,a_2,b)\\
&= \begin{cases}
t_s^{(1)}c_2(a_1,a_2,b) + \tilde{\lambda}_{s,1}d_2(a_1+1,a_2,b) + \tilde{\lambda}_{s,2}d_2(a_1,a_2,b+1) + \tilde{\mu}_s d_2(a_1-1,a_2,b) & a_1 > 0\\
t_s^{(1)}c_2(a_1,a_2,b) + \tilde{\lambda}_{s,1}d_2(a_1+1,a_2,b) + \tilde{\lambda}_{s,2}d_2(a_1,a_2,b+1) + \tilde{\mu}_s d_2(a_1,a_2-1,b) & a_1 = 0, a_2 > 0\\
t_s^{(1)}c_2(a_1,a_2,b) + \tilde{\lambda}_{s,1}d_2(a_1+1,a_2,b) + \tilde{\lambda}_{s,2}d_2(a_1,a_2,b+1) & a_1 = 0, a_2 = 0
\end{cases} .
\end{aligned}
\tag{19}
$$

$$
\begin{aligned}
&d_2^{(2)}(a_1,a_2,b)\\
&= \begin{cases}
\begin{aligned} &t_s^{(2)}(c_2(a_1,a_2,b))^2 + \tilde{\lambda}_{s,1}d_2^{(2)}(a_1+1,a_2,b) + \tilde{\lambda}_{s,2}d_2^{(2)}(a_1,a_2,b+1) + \tilde{\mu}_s d_2^{(2)}(a_1-1,a_2,b)\\ &\quad +2t_s^{(1)}c_2(a_1,a_2,b)\left(\tilde{\lambda}_{s,1}d_2(a_1+1,a_2,b) + \tilde{\lambda}_{s,2}d_2(a_1,a_2,b+1) + \tilde{\mu}_s d_2(a_1-1,a_2,b)\right)\end{aligned} & a_1 > 0\\
\begin{aligned} &t_s^{(2)}(c_2(a_1,a_2,b))^2 + \tilde{\lambda}_{s,1}d_2^{(2)}(a_1+1,a_2,b) + \tilde{\lambda}_{s,2}d_2^{(2)}(a_1,a_2,b+1) + \tilde{\mu}_s d_2^{(2)}(a_1,a_2-1,b)\\ &\quad +2t_s^{(1)}c_2(a_1,a_2,b)\left(\tilde{\lambda}_{s,1}d_2(a_1+1,a_2,b) + \tilde{\lambda}_{s,2}d_2(a_1,a_2,b+1) + \tilde{\mu}_s d_2(a_1,a_2-1,b)\right)\end{aligned} & a_1 = 0, a_2 >\\
\begin{aligned} &t_s^{(2)}(c_2(a_1,a_2,b))^2 + \tilde{\lambda}_{s,1}d_2^{(2)}(a_1+1,a_2,b) + \tilde{\lambda}_{s,2}d_2^{(2)}(a_1,a_2,b+1)\\ &\quad +2t_s^{(1)}c_2(a_1,a_2,b)\left(\tilde{\lambda}_{s,1}d_2(a_1+1,a_2,b) + \tilde{\lambda}_{s,2}d_2(a_1,a_2,b+1)\right)\end{aligned} & a_1 = 0, a_2 =
\end{cases}
\end{aligned}
$$

4.2.1 Computational Aspects

We discuss only the computational aspects of evaluating $E[D_j]$ and $E[D_j^2]$ for $j = 2$ since the evaluation for $j = 1$ is much simpler, and thus the overall computation is dominated by that of $j = 2$.

Similarly to Section 4.1.1, (18) is replaced by a set of K^2 linear equations.

Observe that the column by column method described in Section 4.1.1, for evaluating $E[D_j]$ and $E[D_j^2]$, does not work in this case, as each value of $d_2(a_1,a_2,b)$ can depend on both $d_2(a_1-1,a_2,b)$, and $d_2(a_1+1,a_2,b)$. However, one may iterate the equality from (19), starting with an initial guess, say zero, until the required relative accuracy is reached.

For example, one can keep iterating until the relative change in the sum of absolute values is smaller than some small constant α, say 10^{-10}. If I iterations are needed to reach the required relative accuracy, the time complexity is $O(IK^3)$. As the computation requires keeping a copy of the values of $d_2(a_1, a_2, b)$ for one iteration, the space complexity is $O(K^3)$.

Our experience shows that for $rho = 0.8$, when choosing $K = 30, \alpha = 10^{-10}$ led to $I = 200$ at the worst case. The results achieved were sufficiently close to those reached by simulation.

4.2.2 System Evaluation: Numerical Results

Here again, of particular interest is the unfairness as function of the mean service times ratio μ_1/μ_2. In this section we also demonstrate how the analysis applies to a very common real life situation.

Suppose that there are two distinct classes of customers, with different service time distributions, arriving at a single server system. For example, suppose that a clerk issues two types of documents, one requiring only his signature, and one requiring his full attention for several minutes. It is common to suggest, due to fairness reasons, that customers with shorter service requirement (those requiring only a signature) should be served ahead of other customers. For simplicity assume that the rates of arrival of both customer classes are equal.

It might be true that this suggestion is indeed fair. This, however, may depend on the parameters, and it is reasonable to predict that the shorter the service times of the priority class are, the greater are the fairness benefits (relative to FCFS). One can therefore predict that there is some minimum ratio of the mean service requirement of the "preferred" class, to that of the rest of the population, below which the priority schedule is more fair than FCFS.

To demonstrate this, we will compare the FCFS system, analyzed in Section 4.1, to the priority system analyzed here. As before, we maintain a constant $\rho = 0.8$, independently of μ_1/μ_2, and set $\lambda_1 = \lambda_2 = 0.1$.

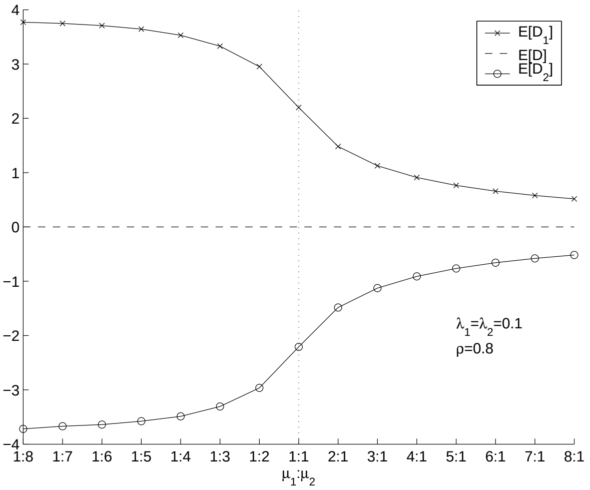

Figure 4: Expected Discrimination in a Single Server Priority system with Two Customer Classes

Figure 4 depicts $E[D]$ for the two classes as well as for the entire system. We observe the following properties:

1. Class 1 is always positively discriminated, and class 2 is always negatively discriminated. Indeed, the discrimination of class 1 customers is at the expense of class 2 customers, since $E[D] = 0$. This result is in fact correct for a much wider set of arrival and service distributions, and for any number of servers, due to Theorem 3.4.

2. The positive (negative) discrimination is monotone-increasing (decreasing) in the expected service requirement of class 1 customers, as expected from Theorem 3.5.

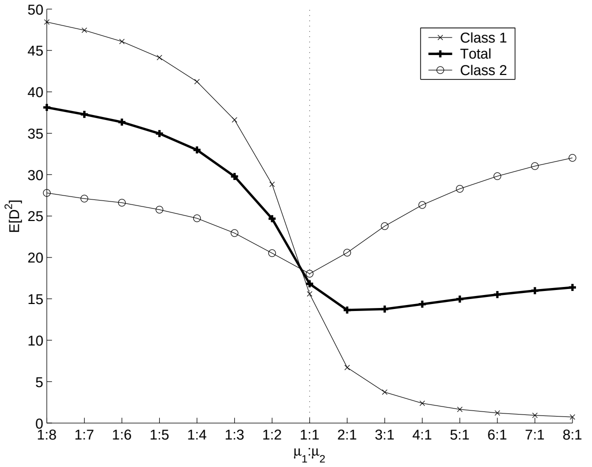

Figure 5: Unfairness in a Single Server Priority system with Two Customer Classes

Figure 5 depicts the unfairness, as measured by the second moment of the discrimination, for the two classes, and for the system. We observe the following properties:

1. The highest system unfairness is observed at the left part of the figure. This is the case where very long jobs (class 1) receive priority over the short jobs. This behavior is naturally expected.

2. One may observe that in the right side of the figure, where the shorter jobs receive priority, the system unfairness slightly increases with the service requirement ratio. This might be counter-intuitive at first sight, since priority is given to the short jobs. This increase is however explainable, and results from unfairness between class 2 customers and themselves, which increases at this region due to the increased variability in service time. Note that the unfairness observed by class 1 customers approaches zero at this range, as expected (see Theorem 3.5, and observe that the same proof applies for the second moment).

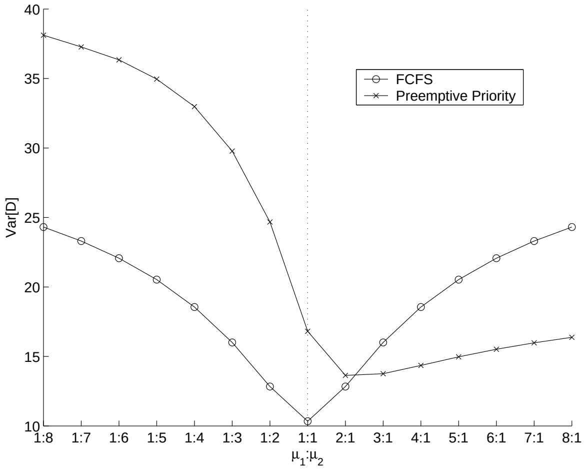

Figure 6: Unfairness in Single Server systems with Two Customer Classes

Finally, Figure 6 compares the unfairness in the FCFS schedule that in the priority one. We observe the following properties:

1. When class 1 customers have longer expected service requirement it is less fair, system-wise, to give them priority.

2. When class 1 customers have shorter expected service requirement and the ratio is over 2 : 1 it is more fair, system-wise, to use two queues and give priority to the shorter jobs.

To conclude this section, we observe that in the specific case analyzed, there is indeed a threshold value for the ratio of the mean non-priority job size, to the mean prioritized job size. If the ratio is below this threshold, it is more fair to serve the customers in FCFS manner. If the ratio is above this threshold, the priority manner is more fair. We conjecture, and leave it open in the framework of this paper, that this property will also apply to non exponential distributions of service and inter-arrival times.

Recall the clerk case, presented in the beginning of this section. The results seem to agree with common intuition - it is less fair to prioritize a specific class of customers (over another class), unless the service requirement of the prioritized customers, is small enough compared to the others.

4.3 Priority Scheduling for Multiple (> 2) Customer Classes

Below, we analyze the priority schedule for a system with u classes of customers. The methodology presented can be used in the analysis of other multi-class systems.

Consider an arbitrary tagged customer of class j, denoted C. Let b be the number of customers behind C. This includes customers of lower priorities (of classes $i > j$) and class j customers behind C in the queue. Let $a_i, \quad i = 1, 2, \dots, j$ be the number of class i customers ahead of C and let $\bar{\bar{a}}$ denote the vector $(a_1, a_2, \dots, a_j)$. Due to the memoryless properties of the system, the state $(\bar{\bar{a}}, b)$, denoted $\mathcal{S}_{\bar{\bar{a}},b}$, as observed by C, captures all that is needed to predict the future discrimination of C. Note that $j+1$ variables are required to describe the state of a class j customer.

From (4), the momentary discrimination during a slot where C is in state $\mathcal{S}_{\bar{\bar{a}},b}$, denoted $c(\bar{\bar{a}}, b)$, is

$$c(\bar{\bar{a}}, b) = \begin{cases} -\frac{1}{b+1+\sum_{l=1}^{j} a_l} & \sum_{l=1}^{j} a_l > 0 \\ 1 - \frac{1}{b+1} & \sum_{l=1}^{j} a_l = 0 \end{cases}.$$

Let $\bar{\bar{k}}$ denote the vector $(k_1, k_2, \dots, k_u)$. Let $E[D_j|\bar{\bar{k}}]$ denote the expected value of discrimination of a class j customer, given that on arrival the customer sees k_i class i customers, $i = 1, 2, \dots, u$.

Let $P_{\bar{\bar{k}}}$ be the steady state probability that there are k_i class i customers, $i = 1, 2, \dots, u$, in the system. Then

$$E[D_j] = \sum_{k_1=0}^{\infty} \sum_{k_2=0}^{\infty} \cdots \sum_{k_u=0}^{\infty} E[D_j|\bar{\bar{k}}] P_{\bar{\bar{k}}}$$

$$E[D_j^2] = \sum_{k_1=0}^{\infty} \sum_{k_2=0}^{\infty} \cdots \sum_{k_u=0}^{\infty} E[D_j^2|\bar{\bar{k}}] P_{\bar{\bar{k}}}.$$

The first two moments of $E[D]$ are the same as in (13) and (14).

$P_{\bar{\bar{k}}}$ can be calculated from the balance equations in a similar way to (18).

Let $D_j(\bar{\bar{a}}, b)$ be a random variable, denoting the discrimination experienced by a class j

customer, through a walk starting at $\mathcal{S}_{\bar{\bar{a}},b}$, and ending at its departure. Then

$$E[D_j|\bar{\bar{k}}] = E\left[D_j((k_1,k_2,\ldots,k_j),\sum_{i=j+1}^{u} k_i\right].$$

Let $d_j(\bar{\bar{a}},b)$ and $d_j^{(2)}(\bar{\bar{a}},b)$ be the first two moments of $D_j(\bar{\bar{a}},b)$.

For a class j customer, assume C is in $\mathcal{S}_{\bar{\bar{a}},b}$ at slot i. Let s be the type of customer being served in the i-th slot. Then

$$s = \min_{i:\, 1\le i\le j, a_i>0}\{i\}.$$

Again s is implied directly from $\bar{\bar{a}}$ and thus does not have to be explicitly accounted for in the state.

The slot length is exponentially distributed with first two moments $t_s^{(1)}$ and $t_s^{(2)}$, from (11). Let $\bar{\bar{0}}$ be the zero vector of length j. Let $\bar{\bar{I}}_k$ be the zero vector of length j, where the k-th element equals one.

At the slot end, the system will encounter one of the following events and C's state will change accordingly:

1. A customer of class $k \ge j$ arrives at the system. The probability of this event is $\tilde{\lambda}_{s,k}$ (for each $k \ge j$). C's state changes to $\mathcal{S}_{\bar{\bar{a}},b+1}$.
2. A customer of class $k < j$ arrives at the system. The probability of this event is $\tilde{\lambda}_{s,k}$ (for each $k < j$). C's state changes to $\mathcal{S}_{\bar{\bar{a}}+\bar{\bar{I}}_k,b}$.
3. A customer leaves the system. The probability of this event is $\tilde{\mu}_s$. If C is being served ($\bar{\bar{a}} = \bar{\bar{0}}$), C leaves the system. Otherwise, C's state changes to $\mathcal{S}_{\bar{\bar{a}}-\bar{\bar{I}}_s,b}$.

Note that $\tilde{\lambda}_{s,k}, \tilde{\mu}_s$ are given in (12), which applies to systems with multiple classes. This leads to the following recursive expressions

$$\begin{aligned} d_j(\bar{\bar{a}},b) = t_s^{(1)}c(\bar{\bar{a}},b) + \sum_{k=j}^{u}\tilde{\lambda}_{s,k}d_j(\bar{\bar{a}},b+1) + \sum_{k=1}^{j-1}\tilde{\lambda}_{s,k}d_j(\bar{\bar{a}}+\bar{\bar{I}}_k,b)& \\ + \begin{cases}\tilde{\mu}_s d_j(\bar{\bar{a}}-\bar{\bar{I}}_s,b) & \bar{\bar{a}} \ne \bar{\bar{0}} \\ 0 & \bar{\bar{a}} = \bar{\bar{0}}\end{cases}&. \end{aligned}$$

$$\begin{aligned} d_j^{(2)}(\bar{\bar{a}},b) = t_s^{(2)}c(\bar{\bar{a}},b)^2 + \sum_{k=j}^{u}\tilde{\lambda}_{s,k}d_j^{(2)}(\bar{\bar{a}},b+1) + \sum_{k=1}^{j-1}\tilde{\lambda}_{s,k}d_j^{(2)}(\bar{\bar{a}}+\bar{\bar{I}}_k,b)& \\ + 2t_s^{(1)}c(\bar{\bar{a}},b)\left(\sum_{k=j}^{u}\tilde{\lambda}_{s,k}d_j(\bar{\bar{a}},b+1) + \sum_{k=1}^{j-1}\tilde{\lambda}_{s,k}d_j(\bar{\bar{a}}+\bar{\bar{I}}_k,b)\right)& \\ + \begin{cases}\tilde{\mu}_s d_j^{(2)}(\bar{\bar{a}}-\bar{\bar{I}}_s,b) + 2t_s^{(1)}c(\bar{\bar{a}},b)\tilde{\mu}_s d_j(\bar{\bar{a}}-\bar{\bar{I}}_s,b) & \bar{\bar{a}} \ne \bar{\bar{0}} \\ 0 & \bar{\bar{a}} = \bar{\bar{0}}\end{cases}&. \end{aligned}$$

4.3.1 Computational Aspects

While the method described in Section 4.2.1 applies here as well, in some cases, when the number of classes is large, the amount of computation required makes it impractical. Assume the number of classes is u. Evaluating $P_{\bar{k}}$ requires the solution of a set of K^u linear equations, where we recall that K is the maximum relevant number of customers that can be seen on arrival. The rest of the algorithm described in Section 4.2.1 has time complexity of $O(IK^{u+1})$ where I is the number of iterations required, and space complexity of $O(K^{u+1})$.

In order to make the algorithm practical the following approximation methods can be utilized:

1. If several classes with higher priority than j have similar expected service times, they can be treated as one class, with an arrival rate which is the sum of the arrival rates of all the relevant classes.

2. When calculating $E[D_j^2]$ for a specific class, a lower (upper) bound can be calculated by treating all classes of higher priority than j as one class, with the expected service time of the class with the lowest (highest) expected service time between those classes, and an arrival rate which is the sum of the arrival rates. Note that all classes with lower priority are treated as one class anyway. This leads to $u = 3$ which is usually practical to analyze.

3. An approximation of $E[D_j]$ or $E[D_j^2]$ can be calculated by treating all classes of higher priority than j as one class, with an expected service time which is the weighted (by arrival rate) average of their expected service times, and an arrival rate which is the sum of the arrival rates. As before, all classes with lower priority are treated as one class anyway. This also leads to $u = 3$. How far this result is from the exact one, depends on the variability across the service times of these classes

5 Multiple Server Systems - FCFS

Multiple server systems are common in a many applications such as banks, airports and others. There is a large variety of strategies to operate these systems, and they raise important fairness issues. In this paper we focus on one strategy, namely the single queue one, and leave other strategies, with the wealth of important fairness issues they raise, to future work.

In the single queue strategy all the customers arrive at a single queue. The servers use a service policy (usually FCFS) to serve the entire population of customers. This strategy raises an important fairness question. Suppose one has the capability to combine several servers into one server with a higher service rate, without losing service power (and while increasing efficiency, see Kleinrock [17, Theorem 4.2]). Would the new setup be more or less fair? For example, observe the check-in line for flights, where most airline companies use a single queue multiple server setup. Assume that the rate at which customers can be checked in, can be doubled by assigning to each post one person for baggage check-in, and another for flight seat assignment, while halving the number of serving posts. Would this increase or decrease the system fairness?

To answer this question, we analyze the dual-class dual-server case, $u = m = 2$. The methodology presented in Section 4.3 can be used to expand these results to the multiple (> 2) class case. We then show how this analysis can be applied to answer the question raised above.

5.1 Analysis

Consider an arbitrary tagged customer of class j, denoted C. Let a be the number of customers ahead of C in the queue, including those in service. If C is being served, a includes customers served by servers with index lower than the server serving C. Let b be the number of customers behind C. If C is served, b includes customers served by servers with index higher than the server serving C. Let s_k be the type of customer served by the k-th server, $k = 1, 2$. If no customer is being served by the k-th server, let $s_k = 0$. Assume that if a customer joins a system with several unoccupied servers, she is served by the unoccupied server with the lowest index. Due to the memoryless properties of the system, the state (a, b, s_1, s_2), denoted $\mathcal{S}_{a,b,s_1,s_2}$, captures all that is needed to predict the future discrimination of C.

From (4), the momentary discrimination during a slot where C is in state $\mathcal{S}_{a,b,s_1,s_2}$, denoted $c(a, b)$, is

$$c(a,b) = \begin{cases} -\frac{2}{a+b+1} & a > 1 \\ 1 - \frac{1}{a+b+1} & a = 0, b = 0 \\ 1 - \frac{2}{a+b+1} & a = 0, b > 0 \text{ or } a = 1 \end{cases}, \tag{20}$$

and is not dependent on s_1 and s_2.

Let $E[D_j|k, s_1, s_2]$, $k = 0, 1, \ldots,$ $s_1, s_2 = 0, 1, 2$, denote the expected value of discrimination of a class j customer, given that the customer sees k customers on arrival (including the ones being served), and the customers being served by the respective servers are of classes s_1 and s_2. For completeness, define $E[D_j|k, s_1, s_2] \overset{def}{=} 0$ for $k = 0, s_1 > 0; k = 0, s_2 > 0; k = 1, s_1, s_2 > 0; k = 1, s_1, s_2 = 0; k > 1, s_1 = 0; k > 1, s_2 = 0$.

Let P_{k,s_1,s_2} be the steady state probability that there are k customers in the system, and the customers being served are of classes s_1 and s_2, by the respective servers (again, $s_i = 0$ if no customer is being served by the i-th server). The first two moments of D_j are given by

$$E[D_j] = \sum_{k=0}^{\infty} \sum_{s_1=0}^{2} \sum_{s_2=0}^{2} E[D_j|k, s_1, s_2] P_{k,s_1,s_2}$$

$$E[D_j^2] = \sum_{k=0}^{\infty} \sum_{s_1=0}^{2} \sum_{s_2=0}^{2} E[D_j^2|k, s_1, s_2] P_{k,s_1,s_2}.$$

The first two moments of $E[D]$ are the same as in (13) and (14).

For the next steps of the analysis, we follow the methodology presented in the previous section. We first find the steady state probabilities P_{k,s_1,s_2}. Then we let $D_j(a,b,s_1,s_2)$ be a random variable, denoting the discrimination experienced by a class j customer, through a walk starting at $\mathcal{S}_{a,b,s_1,s_2}$, and ending at its departure. We can thus express $E[D_j|k,s_1,s_2]$ and $E[D_j^2|k,s_1,s_2]$ in terms of the first two moments of $D_j(a,b,s_1,s_2)$. Using an enumeration of the possible future states a customer can see, and their probabilities, we reach a recursive expression for these moments. The analysis in full is brought in Appendix B

5.2 Comparative Results

Our goal is to compare a single server system to a multiple server system, with the same total service rate. For the single server system we use the system analyzed in Section 4.1. As the server in the single server system has a service rate of one unit, each of the servers in the dual queue system will have a service rate of half a unit. Thus, $w(t) = 1/2$ when only one customer is served, and $w(t) = 2 \times 1/2 = 1$ when both servers are served. Similarity $\sigma_l(t) = 1/2$ when C_l is served. This modifies the momentary discrimination function given in (20):

$$c(a,b) = \begin{cases} -\frac{1}{a+b+1} & a > 1 \\ 1/2 - \frac{1/2}{a+b+1} = 0 & a = 0, b = 0 \\ 1/2 - \frac{1}{a+b+1} & a = 0, b > 0 \text{ or } a = 1 \end{cases} .$$

Accounting for the service rate should also be done in determining the slot size in (27) and the state traversal probabilities in (28). This is done by noting that the service time of a customer equals its service requirement divided by the service rate, thus in (27) and (28) one needs to replace μ_1 and μ_2 with $\mu_1/2$ and $\mu_2/2$ respectively. This also applies to the dteady state probability equations (21)-(26).

One may claim that the comparison of these two systems is still inadequate, since the multi-server system is less efficient (see Kleinrock [17, Theorem 4.2]). Thus, and for identifying whether the fairness measure differences between the two systems are mainly due to fairness basic assumptions, we consider in addition to the two systems a third "hypothetical" system. In the third system when only one customer is present in the system, he is given the full service rate of the system. The system is denoted "hypothetical" since in some applications it might be impractical to implement. For conciseness, the full analysis of this system is not brought here. However, the differences in the analysis are minor - the steady state analysis is a little different, as are the slot size and state transition probabilities when there is only one customer in the system.

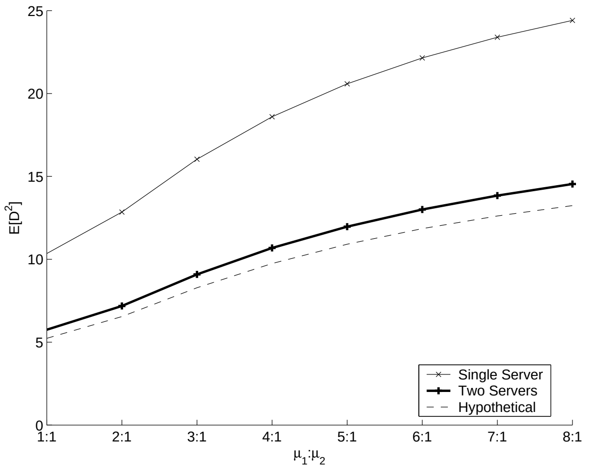

Figure 7: Comparison Between the Unfairness in Single Server and Dual Server Systems

Figure 7 depicts the unfairness in the three systems (single server, dual server and "hypothetical"), as a function of the mean service times ratio. The comparison shows that the dual server configuration is consistently more fair than the single server system. The figure also demonstrates that the unfairness measure of the hypothetical system is even lower (though only slightly) than that of the dual server system. This suggests that the lower measure observed for the dual-server system can be attributed to its more fair operation (and not due to its inefficiency).

The higher fairness of the multiple server configuration can be explained from the resource allocation point of view (as RAQFM does). In the single server system, the resources are always allocated to one customer at a time. In the multiple server system, the resources are allocated among more customers at any epoch, and therefore the resource allocation is more fair. In fact, as the number of servers grows, the system approaches the processor sharing model, which is the most fair system (see Raz et al. [26]); off-course under such conditions the system efficiency decreases and customer delays increase leading to potential customer dissatisfaction due to poor performance. The question whether the use of k servers is *always* more fair than the use of a single server remains open for future research.

6 Concluding Remarks

Using the RAQFM measure we have shown that under rather general assumptions the expected positive discrimination of a customer is increasing in his required service length (Theorem 3.2). Hence, prioritization of shorter jobs over longer jobs maybe justified, since otherwise shorter jobs are negatively discriminated. We then developed a method for calculating expected discrimination and fairness for a multi-class system with exponential inter-arrival and service times. The calculated results show that assigning preemptive priority to a class of shorter jobs is justified if the ratio of the expected service times of the longer jobs to the sorter jobs exceeds a certain number.

We also analyzed the multi-server system (with 2 servers) and showed that the use of multiple servers can increase fairness.

Several important fairness issues remain open: The fairness of multi-queue multi-server architectures, and accounting for class weights. These are currently under study.

References

[1] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK Users' Guide*. SIAM, Philadelphia, third edition, 1999.

[2] B. Avi-Itzhak. Preemptive repeat priority queues as a special case of the multipurpose server problem – I. *Oper. Res.*, 11(4):597–609, 1963.

[3] B. Avi-Itzhak. Preemptive repeat priority queues as a special case of the multipurpose server problem – II. *Oper. Res.*, 11(4):610–619, 1963.

[4] B. Avi-Itzhak and H. Levy. On measuring fairness in queues. *Advances of Applied Probability*, to appear, 2004.

[5] B. Avi-Itzhak and P. Naor. On a problem of preemptive priority queuing. *Oper. Res.*, 9(5):664–672, 1961.

[6] D. R. Cox and W. L. Smith. *Queues*. Methuen/Wiley, London, 1961.

[7] R. H. Davis. Waiting-time distribution of a multi-server priority queueing system. *Operations Research*, 14:133–136, 1966.

[8] T. A. Davis. *UMFPACK Version 4.0 User Guide*. Dept. of Computer and Information Science and Engineering, Univ. of Florida, Gainesville, Florida, 2002.

[9] A. Demers, S. Keshav, and S. Shenker. Analysis and simulation of a fair queueing algorithm. *Internetworking Research and Experience*, 1:3–26, 1990.

[10] S. J. Golestani. A self-clocked fair queueing scheme for broadband applications. In *Proceedings of IEEE INFOCOM '94*, pages 636–646, Toronto, Ontario, June 1994.

[11] A. G. Greenberg and N. Madras. How fair is fair queueing? *Journal of the ACM*, 3 (39):568–598, 1992.

[12] A. van Harten and A. Sleptchenko. On multi-class multi-server queueing and spare parts management. Technical Report WP-49, BETA publication, University of Twente, Enschede, The Netherlands, 2000.

[13] A. van Harten, A. Sleptchenko, and M. C. van der Heijden. On multi-class multi-server queue with preemptive priorities. Technical Report WP-77, BETA publication, University of Twente, Enschede, The Netherlands, 2003.

[14] N. K. Jaiswal. *Priority Queues.* Academic Press, New York, 1968.

[15] O. Kella and U. Yechiali. Waiting times in the non-preemptive priority M/M/c queue. *Commun. Statist.-Stochastic Models*, 1(2):257–262, 1985.

[16] O. Kella and U. Yechiali. Priorities in M/G/1 queue with server vacations. *Naval Research Logistics*, 35:23–24, 1988.

[17] L. Kleinrock. *Communication Nets: Stochastic Message Flow and Delay.* McGraw-Hill, New-York, 1964. Out of Print. Reprinted by Dover Publications, 1972.

[18] L. Kleinrock. *Queueing Systems, Volume 2: Computer Applications.* Wiley, 1976.

[19] R. C. Larson. Perspective on queues: Social justice and the psychology of queueing. *Operations Research*, 35:895–905, Nov-Dec 1987.

[20] C. Palm. Methods of judging the annoyance caused by congestion. *Tele. (English Ed.)*, 2:1–20, 1953.

[21] A. Parekh and R. G. Gallager. A generalized processor sharing approach to flow control in integrated services networks: The single node case. *IEEE/ACM Trans. Networking*, 1:344–357, June 1993.

[22] A. Parekh and R. G. Gallager. A generalized processor sharing approach to flow control in integrated services networks: The multiple node case. *IEEE/ACM Trans. Networking*, 2:137–150, 1994.

[23] A. Rafaeli, G. Barron, and K. Haber. The effects of queue structure on attitudes. *Journal of Service Research*, 5(2):125–139, 2002.

[24] A. Rafaeli, E. Kedmi, D. Vashdi, and G. Barron. Queues and fairness: A multiple study investigation. Faculty of Industrial Engineering and Management, Technion. Haifa, Israel. Under review, 2003.

[25] D. Raz, H. Levy, and B. Avi-Itzhak. Fairness in multiple server queueing systems. Forthcoming, 2004.

[26] D. Raz, H. Levy, and B. Avi-Itzhak. A resource-allocation queueing fairness measure. To appear in Proceedings of SIGMETRICS/Performance'04, June 2004.

[27] J. Rexford, A. Greenberg, and F. Bonomi. Hardware-efficient fair queueing architectures for high-speed networks. In *Proceedings of IEEE INFOCOM '96*, pages 638–646, March 1996.

[28] A. Sleptchenko. *Integral Inventory Control in Spare Parts Networks with Capacity Restrictions.* PhD thesis, University of Twente, 2002.

[29] H. Takagi. *Queueing Analysis, A Foundation of Performance Evaluation*, volume 1: Vacation and Priority Systems (Part 1). North-Holland, Amsterdam, The Netherlands, 1991.

[30] Y. T. Wang and R. J. T. Morris. Load sharing in distributed systems. *IEEE Trans. on computers*, C-34(3):204–217, 1985.

[31] W. Whitt. The amount of overtaking in a network of queues. *Networks*, 14(3):411–426, 1984.

[32] A. Wierman and M. Harchol-Balter. Classifying scheduling policies with respect to unfairness in an M/GI/1. In *Proceedings of ACM Sigmetrics 2003 Conference on Measurement and Modeling of Computer Systems*, San Diego, CA, June 2003.

[33] R. Wolff. Poisson arrivals see time averages. *Oper. Res.*, 30(2):223–231, 1982.

Appendix A Equations for the Single Class FCFS Queue

$$\tilde{\lambda} = \frac{\lambda}{\lambda+\mu} = \frac{\rho}{1+\rho} \qquad \tilde{\mu} = \frac{\mu}{\lambda+\mu} = \frac{1}{1+\rho}$$
$$t^{(1)} = \frac{1}{\lambda+\mu} \qquad t^{(2)} = \frac{2}{(\lambda+\mu)^2} = 2(t^{(1)})^2$$

$$c(a,b) = \begin{cases} -\frac{1}{a+b+1} & a>0 \\ 1-\frac{1}{b+1} & a=0 \end{cases}$$
$$d(a,b) = \begin{cases} t^{(1)}c(a,b) + \tilde{\lambda}d(a,b+1) + \tilde{\mu}d(a-1,b) & a>0 \\ t^{(1)}c(a,b) + \tilde{\lambda}d(a,b+1) & a=0 \end{cases}$$
$$d^{(2)}(a,b) = \begin{cases} \begin{array}{c} t^{(2)}(c(a,b))^2 + \tilde{\lambda}d^{(2)}(a,b+1) + \tilde{\mu}d^{(2)}(a-1,b) \\ +2t^{(1)}c(a,b)(\tilde{\lambda}d(a,b+1) + \tilde{\mu}d(a-1,b)) \end{array} & a>0 \\ t^{(2)}(c(a,b))^2 + \tilde{\lambda}d^{(2)}(a,b+1) + 2t^{(1)}c(a,b)\tilde{\lambda}d(a,b+1) & a=0 \end{cases}$$

Appendix B Analysis of the Multiple Server Case

P_{k,s_1,s_2} follows

$$(\lambda + \mu_{s_1} + \mu_{s_2})P_{n,s_1,s_2} = \lambda P_{n-1,s_1,s_2}$$

$$+ \mu_1 p_{s_2} P_{n+1,s_1,1} + \mu_2 p_{s_2} P_{n+1,s_1,2} + \mu_1 p_{s_1} P_{n+1,1,s_2} + \mu_2 p_{s_1} P_{n+1,2,s_2} \quad n > 2 \qquad (21)$$

$$\begin{aligned}(\lambda + \mu_{s_1} + \mu_{s_2}) P_{2,s_1,s_2} = {} & \lambda_{s_2} P_{1,s_1,0} + \lambda_{s_1} P_{1,0,s_2} \\ & + \mu_1 p_{s_2} P_{3,s_1,1} + \mu_2 p_{s_2} P_{3,s_1,2} + \mu_1 p_{s_1} P_{3,1,s_2} + \mu_2 p_{s_1} P_{3,2,s_2}\end{aligned} \qquad (22)$$

$$(\lambda + \mu_{s_1}) P_{1,s_1,0} = \lambda_{s_1} P_{0,0,0} + \mu_1 P_{2,s_1,1} + \mu_2 P_{2,s_1,2} \qquad (23)$$

$$(\lambda + \mu_{s_2}) P_{1,0,s_2} = \mu_1 P_{2,1,s_2} + \mu_2 P_{2,2,s_2} \qquad (24)$$

$$\lambda P_{0,0,0} = \mu_1 P_{1,1,0} + \mu_2 P_{1,2,0} + \mu_1 P_{1,0,1} + \mu_2 P_{1,0,2} \qquad (25)$$

$$\sum_{i=0}^{\infty} \sum_{s_1=0}^{2} \sum_{s_2=0}^{2} P_{i,s_1,s_2} = 1. \qquad (26)$$

See Remark 4.1 for additional work on the subject of steady state probabilities.

Let $D_j(a, b, s_1, s_2)$ be a random variable, denoting the discrimination experienced by a class j customer, through a walk starting at $\mathcal{S}_{a,b,s_1,s_2}$, and ending at its departure. Then

$$E[D_j | k, s_1, s_2] = \begin{cases} E[D_j(k, 0, s_1, s_2)] & k > 1 \\ E[D_j(1, 0, s_1, j)] & k = 1, s_2 = 0 \\ E[D_j(0, 1, j, s_2)] & k = 1, s_1 = 0 \\ E[D_j(0, 0, j, 0)] & k = 0 \end{cases}$$

$$E[D_j^2 | k, s_1, s_2] = \begin{cases} E[D_j^2(k, 0, s_1, s_2)] & k > 1 \\ E[D_j^2(1, 0, s_1, j)] & k = 1, s_2 = 0 \\ E[D_j^2(0, 1, j, s_2)] & k = 1, s_1 = 0 \\ E[D_j^2(0, 0, j, 0)] & k = 0 \end{cases}.$$

Let $d_j(a, b, s_1, s_2)$ and $d_j^{(2)}(a, b, s_1, s_2)$ be the first two moments of $D_j(a, b, s_1, s_2)$.

Assume customer C, of class j, is in $\mathcal{S}_{a,b,s_1,s_2}$ at slot i. The slot length is exponentially distributed with first two moments $t^{(1)}_{s_1,s_2}$ and $t^{(2)}_{s_1,s_2}$, which equal

$$t^{(1)}_{j_1,j_2} = \frac{1}{\lambda + \mu_{j_1} + \mu_{j_2}} \qquad t^{(2)}_{j_1,j_2} = \frac{2}{(\lambda + \mu_{j_1} + \mu_{j_2})^2} = 2(t^{(1)}_j)^2, \qquad (27)$$

where $\mu_0 \overset{def}{=} 0$.

The probability that a slot where class j_1 and class j_2 customers are being served ends with an arrival of a class k customer is denoted $\tilde{\lambda}_{(j_1,j_2),k}$. The probability that such a slot ends with an arrival of any customer is denoted $\tilde{\lambda}_{(j_1,j_2)}$. The probability that it ends with the departure of a class k customer is denoted $\tilde{\mu}_{(j_1,j_2),k}$.

$$\tilde{\lambda}_{(j_1,j_2),k} = \frac{\lambda_k}{\lambda + \mu_{j_1} + \mu_{j_2}} \qquad \tilde{\lambda}_{(j_1,j_2)} = \frac{\lambda}{\lambda + \mu_{j_1} + \mu_{j_2}} \qquad \tilde{\mu}_{(j_1,j_2),k} = \frac{\mu_k}{\lambda + \mu_{j_1} + \mu_{j_2}}. \qquad (28)$$

Event		Probability	Next State
$a > 2$			
Customer arrival		$\tilde{\lambda}_{(s_1,s_2)}$	$\mathcal{S}_{a,b+1,s_1,s_2}$
Class s_1 departure, next customer of class 1		$\tilde{\mu}_{(s_1,s_2),s_1}p_1$	$\mathcal{S}_{a-1,b,1,s_2}$
Class s_1 departure, next customer of class 2		$\tilde{\mu}_{(s_1,s_2),s_1}p_2$	$\mathcal{S}_{a-1,b,2,s_2}$
Class s_2 departure, next customer of class 1		$\tilde{\mu}_{(s_1,s_2),s_2}p_1$	$\mathcal{S}_{a-1,b,s_1,1}$
Class s_2 departure, next customer of class 2		$\tilde{\mu}_{(s_1,s_2),s_2}p_2$	$\mathcal{S}_{a-1,b,s_2,2}$
$a = 2$			
Customer arrival		$\tilde{\lambda}_{(s_1,s_2)}$	$\mathcal{S}_{a,b+1,s_1,s_2}$
Class s_1 departure		$\tilde{\mu}_{(s_1,s_2),s_1}$	$\mathcal{S}_{0,b+1,j,s_2}$
Class s_2 departure		$\tilde{\mu}_{(s_1,s_2),s_2}$	$\mathcal{S}_{1,b+1,s_1,j}$
$a = 1$			
Customer arrival		$\tilde{\lambda}_{(s_1,s_2)}$	$\mathcal{S}_{a,b+1,s_1,s_2}$
Other customer leaves	$b = 0$	$\tilde{\mu}_{(s_1,s_2),s_1}$	$\mathcal{S}_{0,0,0,j}$
	$b > 0$, next customer of class 1	$\tilde{\mu}_{(s_1,s_2),s_1}p_1$	$\mathcal{S}_{1,b-1,1,j}$
	$b > 0$, next customer of class 2	$\tilde{\mu}_{(s_1,s_2),s_1}p_2$	$\mathcal{S}_{1,b-1,2,j}$
C leaves		$\tilde{\mu}_{(s_1,s_2),j}$	
$a = 0, b > 1$			
Customer arrival		$\tilde{\lambda}_{(s_1,s_2)}$	$\mathcal{S}_{a,b+1,s_1,s_2}$
Other customer leaves, next customer of class 1		$\tilde{\mu}_{(s_1,s_2),s_2}p_1$	$\mathcal{S}_{0,b-1,j,1}$
Other customer leaves, next customer of class 2		$\tilde{\mu}_{(s_1,s_2),s_2}p_2$	$\mathcal{S}_{0,b-1,j,2}$
C leaves		$\tilde{\mu}_{(s_1,s_2),j}$	
$a = 0, b = 1$			
Customer arrival		$\tilde{\lambda}_{(s_1,s_2)}$	$\mathcal{S}_{a,b+1,s_1,s_2}$
Other customer leaves		$\tilde{\mu}_{(s_1,s_2),s_2}$	$\mathcal{S}_{0,0,j,0}$
C leaves		$\tilde{\mu}_{(s_1,s_2),j}$	
$a = 0, b = 0$			
Customer of class 1 arrives	$s_2 = j$	$\tilde{\lambda}_{(s_1,s_2),1}$	$\mathcal{S}_{1,0,1,j}$
	$s_1 = j$	$\tilde{\lambda}_{(s_1,s_2),1}$	$\mathcal{S}_{0,1,j,1}$
Customer of class 2 arrives	$s_2 = j$	$\tilde{\lambda}_{(s_1,s_2),2}$	$\mathcal{S}_{1,0,2,j}$
	$s_1 = j$	$\tilde{\lambda}_{(s_1,s_2),2}$	$\mathcal{S}_{0,1,j,2}$
C leaves		$\tilde{\mu}_{(s_1,s_2),j}$	

Table 1: Possible Events and State Transitions for the Dual Server Single Queue FCFS System with Two Customer Classes

Table 1 summarizes the possible events encountered in the beginning of the next slot, their probabilities, and the state that C will observe in the next slot.

This leads to the following recursive expressions

$$d_j(a,b,s_1,s_2) = t^{(1)}_{s_1,s_2} c(a,b) +$$
$$\begin{cases} \tilde{\lambda}_{(s_1,s_2)} d_j(a,b+1,s_1,s_2) + \tilde{\mu}_{(s_1,s_2),s_1} p_1 d_j(a-1,b,1,s_2) + \tilde{\mu}_{(s_1,s_2),s_1} p_2 d_j(a-1,b,2,s_2) + \\ \quad \tilde{\mu}_{(s_1,s_2),s_2} p_1 d_j(a-1,b,s_1,1) + \tilde{\mu}_{(s_1,s_2),s_2} p_2 d_j(a-1,b,s_1,2) & a>2 \\ \tilde{\lambda}_{(s_1,s_2)} d_j(a,b+1,s_1,s_2) + \tilde{\mu}_{(s_1,s_2),s_1} d_j(0,b+1,j,s_2) + \tilde{\mu}_{(s_1,s_2),s_2} d_j(1,b,s_1,j) & a=2 \\ \tilde{\lambda}_{(s_1,s_2)} d_j(a,b+1,s_1,s_2) + \tilde{\mu}_{(s_1,s_2),s_1} p_1 d_j(1,b-1,1,j) + \tilde{\mu}_{(s_1,s_2),s_1} p_2 d_j(1,b-1,2,j) & a=1, b>0 \\ \tilde{\lambda}_{(s_1,s_2)} d_j(a,b+1,s_1,s_2) + \tilde{\mu}_{(s_1,s_2),s_2} p_1 d_j(0,b-1,j,1) + \tilde{\mu}_{(s_1,s_2),s_2} p_2 d_j(0,b-1,j,2) & a=0, b>1 \\ \tilde{\lambda}_{(s_1,s_2)} d_j(a,b+1,s_1,s_2) + \tilde{\mu}_{(s_1,s_2),s_1} d_j(0,0,0,j) & a=1, b=0 \\ \tilde{\lambda}_{(s_1,s_2)} d_j(a,b+1,s_1,s_2) + \tilde{\mu}_{(s_1,s_2),s_2} d_j(0,0,j,0) & a=0, b=1 \\ \tilde{\lambda}_{(s_1,s_2),1} d_j(0,1,j,1) + \tilde{\lambda}_{(s_1,s_2),2} d_j(0,1,j,2) & a=0, b=0, \\ \tilde{\lambda}_{(s_1,s_2),1} d_j(1,0,1,j) + \tilde{\lambda}_{(s_1,s_2),2} d_j(1,0,2,j) & a=0, b=0, \end{cases}$$

A recursive expression for $d_j^{(2)}(a,b,s_1,s_2)$ is similar to (17) and is not brought for the sake of conciseness.

B.1 Computational Aspects

(21)-(26) provide a set of linear equations for calculating P_{k,s_1,s_2}, leading to a sparse set of $4K$ linear equations.

The method described in Section 4.1.1, for evaluating $E[D_j]$ and $E[D_j^2]$, works only for $a>1$, as for $a \leq 1$, $d_j(a,b,s_1,s_2)$ can depend on both $d_j(a,b+1,s_1,s_2)$ and $d_j(a,b-1,s_1,s_2)$. An efficient solution to this is to solve for $a \leq 1$ using the method described in Section 4.2.1, and then continue with $a>1$. The first part of the solution has time complexity of $O(IK)$ and space complexity of $O(K)$. The second part has time complexity of $O(K^2)$ and space complexity of $O(K)$, since it can be done in a "column-by-column" manner.