Effect of class clustering on delay differentiation in priority scheduling

T. Maertens, H. Bruneel and J. Walraevens

Priority scheduling is the most viable way to implement QoS differentiation in telecommunication networks. Most studies on priority scheduling do not take into account possible class clustering. In particular, they assume that different classes occur randomly and independently in the arrival stream of packets. In reality, however, packets of the same class may have the tendency to arrive in clusters. By using existing results, we show in this letter that class clustering may have a severe impact on the achievable delay differentiation in priority scheduling.

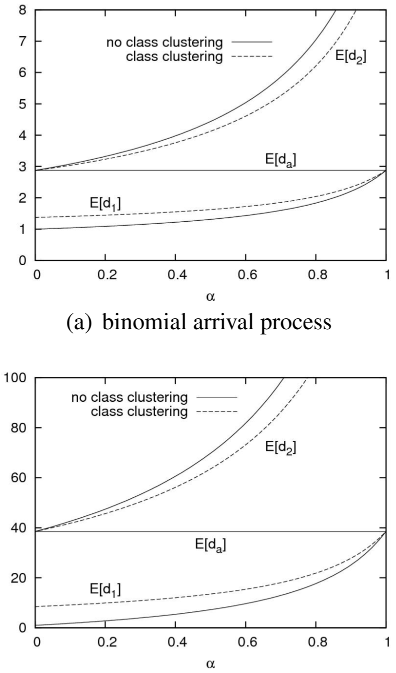
Introduction: Priority(-based) scheduling has been (see, e.g., [1]) and still is (see, e.g., [4]) a popular topic in the domain of telecommunication. Modern telecommunication systems must be capable of supporting voice, data, radio and video/television services. Different services, however, have extremely diverse *Quality-of-Service* (QoS) requirements. Real-time services, like teleconferencing and internet telephony, do not tolerate *delay* but can sustain some *loss*, while non-real-time services, like sending data files, allow for some delay but are quite vulnerable to loss. Here, we focus on delay as QoS measure. Prioritizing *delay-sensitive* services provides delay differentiation among different (types of) services.

To assess the impact of a priority scheduling discipline on the performance, in general, and on the delay differentiation, in particular, of a telecommunication system, one can make use of *multi-class* queueing models. A multi-class queueing model basically describes the way, the pace and the order in which information units or *packets* of all classes, consecutively, arrive at the system, are stored in the system, and are served by or transmitted out of the system. Since these processes are of an uncertain and unpredictable nature, it is common to express them in a stochastic or probabilistic way. A vast literature on multi-class queueing models exist, both in a continuous-time setting (see, e.g., [4]) and in a discrete-time setting (see, e.g., [2, 3]). While most continuous-time models assume that the arrival processes of the different priority classes are mutually independent, many discrete-time models incorporate the possibility of correlation between the numbers of arrivals of the different classes within one time unit or *slot* (see, e.g., [2, 3]).

In the numerical examples of the latter papers, it is usually assumed that the class of an arbitrary packet is independent of the class of other packets arriving in the same slot. The binomial arrival process is commonly used (see, e.g., [**?**, 2, 3]), as this is the arrival process to an output queue of an output-queueing switch. In reality, however, packets of the same class may have the tendency to arrive in *clusters*, i.e., the classes of packets arriving in the same slot may be non-independent. In this letter, we use existing results to demonstrate that so-called (extreme) *class clustering* may have a severe impact on the delay differentiation achieved by a priority scheduling discipline, and can reduce this delay differentiation considerably.

Queueing model: We consider a basic multi-class queueing models, i.e., a discrete-time two-class, named 1 and 2, *static* priority queue with one server, one-slot service times for both classes and class-1 packets having service priority over class-2 packets. The arrival process is characterised in two steps. First, we model the total (aggregated) arrival stream of packets from slot to slot by means of a sequence of i.i.d. non-negative random variables (denoting the total numbers of arrivals in consecutive slots) with common probability generating function (pgf) $A_T(z)$. The total mean number of arrivals per slot, i.e., the total arrival rate, is denoted by $\lambda_T \triangleq A'_T(1)$.

Secondly, the joint pgf $A(z_1, z_2)$ describes the numbers of arrivals of both classes within one slot, and is characterised in terms of $A_T(z)$. We look at two extreme cases. First, when both classes occur randomly and independently in the arrival stream of packets during a slot, $A(z_1, z_2)$ can be written as $A_T(\alpha z_1 + (1-\alpha) z_2)$, with α defined as the fraction of class-1 packets in the overall arrival stream, and thus, equal to the probability that a randomly chosen arriving packet is of class 1. We compare this popular case (see references above) to a second extreme case, i.e., the case where all arriving packets during a slot are of the same class. In that case, $A(z_1, z_2) = \alpha A_T(z_1) + (1-\alpha) A_T(z_2)$. It is obvious that, on average, the same number of (class-1 and class-2) packets arrive in both cases, but that packets are much more clustered in the second case.



(a) binomial arrival process

(b) deterministic batch arrival process

Fig. 1. Mean delays versus α, with $\lambda_T = 0.8$ and $N = 16$

Effect of class clustering: The queueing model described in the previous section has been studied in detail in [3], for general $A(z_1, z_2)$. In [3], the authors derive expressions for the pgfs of the queue contents and the packet delays. These pgfs further lead to expressions for some interesting performance measures involving these quantities. For the mean packet delays, for example, they obtain

$$\mathrm{E}\,[d_1] = 1 + \frac{\lambda_{11}}{2\lambda_1(1-\lambda_1)}, \tag{1}$$

and

$$\mathrm{E}\,[d_2] = 1 + \frac{2\lambda_{12} + \lambda_{22}}{2\lambda_2(1-\lambda_T)} + \frac{\lambda_{11}}{2(1-\lambda_T)(1-\lambda_1)}, \tag{2}$$

where λ_j $(j = 1, 2)$ denotes the class-j arrival rate and $\lambda_{ij} \triangleq \left.\frac{\partial^2 A(z_1, z_2)}{\partial z_i \partial z_j}\right|_{z_i=1, z_j=1}$ $(i, j = 1, 2)$. It is common knowledge that the static priority scheduling discipline achieves maximum delay differentiation between the different classes (see, e.g., [2, 3]). We use the above expressions to demonstrate the impact of (extreme) class clustering on the delay differentiation caused by the static priority scheduling discipline.

Let us first consider the popular binomial distribution for the total arrival stream:

$$A_T(z) = \left(1 - \frac{\lambda_T}{N}(1-z)\right)^N, \tag{3}$$

with N indicating the maximum (total) number of arrivals during a slot. Then Fig. 1(a) shows the mean delays of both classes of packets and of an arbitrary packet ($\mathrm{E}\,[d_a]$) as a function of α (recall that α is defined as the fraction of class-1 packets in the overall arrival stream), for $\lambda_T = 0.8$ and $N = 16$. The delay of an arbitrary packet in a work-conserving system with single-slot service times does not depend on whether the scheduling discipline is First-In-First-Out (FIFO) or there is preferential treatment for one class of packets (see, e.g., [2]):

$$\mathrm{E}\,[d_a] = 1 + \frac{\lambda_{TT}}{2\lambda_T(1-\lambda_T)}, \tag{4}$$

with $\lambda_{TT} = A''_T(1)$. Furthermore, when $\alpha = 0$, $\mathrm{E}\,[d_a] = \mathrm{E}\,[d_2]$, and when $\alpha = 1$, $\mathrm{E}\,[d_a] = \mathrm{E}\,[d_1]$. Indeed, in these cases, all packets are of the same class (i.e., of class 2 and class 1, respectively). The figure clearly shows the influence of class clustering. Without class clustering, class-1 packets have the tendency to arrive spread in time, especially when the class-1 load is small compared to the class-2 load (i.e., when α is small). As a consequence, they hardly influence each other. With class clustering, however, class-1 packets arrive closely together, so a class-1 packet suffers by the other class-1 packets that belong to the same cluster. This results in a higher value for $\mathrm{E}[d_1]$ than in the case of no class clustering. For example, for $\alpha = 0.2$, $\mathrm{E}[d_1]$ increases with approximately 50%. When class-2 packets arrive in clusters, on the other hand, they do not have to endure class-1 arrivals in their arrival slot, so they have a larger probability to be served fast(er), yielding a smaller $\mathrm{E}[d_2]$.

Secondly, we assume a deterministic batch arrival process: there are either no or N arrivals during a slot, with λ_T (still) denoting the mean number of arrivals per slot. Hence,

$$A_T(z) = 1 - \frac{\lambda_T}{N} + \frac{\lambda_T}{N} z^N. \tag{5}$$

Fig. 1(b) illustrates the corresponding mean delays, again as a function of α and for $\lambda_T = 0.8$ and $N = 16$. We notice that the mean delays have much higher values than in the case of a binomial arrival process. This is due to the much higher variance of the (total) number of arrivals for the deterministic batch arrival process. In particular, arriving packets (of both classes) suffer by the other packets of the batch. Absolutely speaking, furthermore, class clustering has a larger impact on the mean delays of both classes when the aggregated arrival process has a deterministic batch distribution than when the total number of arrivals is binomially distributed. For $\alpha = 0.2$ and a deterministic batch arrival process, for example, the value of $\mathrm{E}[d_1]$ in the case of class clustering is more than three times higher than the value of $\mathrm{E}[d_1]$ in the case of no class clustering, while this was only 50% for the binomial arrival process.

By using Exprs. (1) and (2), and applying the definitions of the arrival process, we finally obtain the following formula for the proportion of the delay differentiation in the case of (extreme) class clustering to the delay differentiation in the case of no class clustering:

$$\frac{(\mathrm{E}[d_2] - \mathrm{E}[d_1])_{\text{c.c.}}}{(\mathrm{E}[d_2] - \mathrm{E}[d_1])_{\text{no c.c.}}} = \lambda_T, \tag{6}$$

with c.c. an abbreviation for class clustering. This means that the delay differentiation in the case of class clustering is $(\lambda_T \times 100)\%$ of the delay differentiation in the case of no class clustering. Or, class clustering reduces the delay differentiation with $((1 - \lambda_T) \times 100)\%$. For instance, when $\lambda_T = 0.5$, the delay differentiation in the clustered case is only half of that in the independent case. So when the load is low and packets of the same class have the tendency to arrive in clusters, there is not much use to adopt a priority scheduling. In other words, when the load is low, it is better to "decluster" packets of the same class. This can, for instance, be achieved by avoiding packets of the same class to follow the same path in a telecommunication network.

Conclusion: This letter shows that (extreme) class clustering may have an impact on the achievable delay differentiation in priority scheduling. Existing studies on priority scheduling are thus somewhat deceptive, because they do not take into account possible class clustering.

Acknowledgment: The last author is a Postdoctoral Fellow with the Fund for Scientific Research, Flanders (F.W.O.-Vlaanderen), Belgium.

T. Maertens, H. Bruneel and J. Walraevens (*Ghent University, Department of Telecommunications and Information Processing, SMACS Research Group, Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium*)

E-mail: tmaerten@telin.UGent.be

References

1 Bae, J.J. and Suda, T.: 'Survey of traffic control schemes and protocols in ATM networks', *Proceedings of the IEEE*, 1991, **79**, pp. 170-189.

2 Maertens, T., Walraevens, J., Moeneclaey, M., and Bruneel, H.: 'Performance analysis of a discrete-time queueing system with priority jumps', *International Journal of Electronics and Communications*, 2009, **63**, pp. 853-858.

3 Walraevens, J., Steyaert, B., and Bruneel, H.: 'Performance analysis of a single-server ATM queue with a priority scheduling', *Computers and Operations Research*, 2003, **30**, pp. 1807-1829.

4 Wang, L., Min, G., Kouvatsos, D.D., and Jin, X.: 'Analytical modeling of an integrated priority and WFQ scheduling scheme in multi-service networks', *Computer Communications*, 2010, **33**, pp. S93-S101.