Gradual Delay Differentiation in Priority Scheduling

Tom Maertens, Joris Walraevens and Herwig Bruneel

Ghent University – UGent
Department of Telecommunications and Information Processing
SMACS Research Group
Address: Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium
Phone: +32-9-2648901, Fax: +32-9-2644295
E-mail: {tmaerten,jw,hb}@telin.ugent.be

Extended abstract

Modern telecommunication networks are designed to offer a wide variety of services, such as information access, e-mail, internet telephony, file sharing, and streaming media. Different services, however, may have extremely diverse *Quality-of-Service* (QoS) requirements. *Real-time* services, like internet telephony and streaming media, do not tolerate delay, but can sustain some loss, while *non-real-time* services, like e-mail and file sharing, allow for some delay, but are quite vulnerable to loss and require a large throughput. The traffic that flows through telecommunication devices nowadays can thus more or less be classified into two types. In our research, we only focus on delay as QoS measure. Regarding their different delay requirements, real-time and non-real-time traffic are then categorised as *delay-sensitive* and *delay-tolerant* respectively, and to achieve the required *delay differentiation* between both types of traffic, the delay-sensitive traffic is *favoured* (or *prioritised*) in *scheduling* the packets for *transmission*.

At its simplest, priority is *always* given to delay-sensitive packets, i.e., delay-tolerant packets can only be transmitted when there are no delay-sensitive packets present in the system. The priority levels of both types of traffic thus never change during time. This *static* priority scheduling discipline provides low delays for delay-sensitive packets, but the performance for the delay-tolerant traffic can be degraded severely. Specifically, when the network is highly loaded and a large portion of the network traffic consists of delay-sensitive traffic, static priority scheduling may cause excessive delays for delay-tolerant packets (see e.g., [7]). Although this type of traffic allows for some delay to a certain extent, excessive delays have to be avoided as much as possible. The Transmission Control Protocol (TCP), for example, could consider a delay-tolerant packet with a too big delay as being lost, and would consequently decrease its transmission rate. This decreases the throughput, which is detrimental to data tranfer services. The decrease of the transmission rate, however, is unnecessary, since the delay-tolerant packet is not lost. The ability to differentiate between both types of traffic with respect to their delay has contributed to the success of static priority scheduling, but the impact of the scheme on the performance of specific services may thus be too disadvantageous in some cases.

To obtain a more *gradual* delay differentiation, we introduce *priority jumps* in the priority scheduling: the priority level of delay-sensitive packets is fixed, but the priority level of delay-tolerant packets may increase in the course of time. In the assumption that the two types of packets arrive in separate queues, this means that packets of the so-called *low-priority queue* can *jump* to the (tail of the) *high-priority queue*. Jumped packets are then treated in this queue as if they are delay-sensitive packets. From the transmission channel's point of view, nothing changes in comparison with static priority scheduling: the packet at the *head* of the highest non-empty priority queue is chosen next for transmission. Priority schemes with priority jumps thus build upon the simplicity and efficiency of the static priority scheme, but as opposed to the latter, they prevent delay-tolerant packets from *starving*.

Many criteria can be used to decide if and when packets jump: a maximum queueing delay in the low-priority queue (see e.g., [2, 3]), a queue-length-threshold of the high- or low-priority queue (see e.g., [1, 5]), a random jumping probability per time unit (see e.g., [4]), an arrival characteristic of one type of traffic (see e.g., [6]), . . . Via analyses based on probability generating functions, we study the effect of various jumping criteria on the performance of a discrete-time queueing system. In all cases, some boundary functions need to be determined during the solution process, and the probability generating function approach usually provides an efficient and fast method for this purpose. Once the probability generating functions of the queue contents and the packet delays are calculated, expressions for the mean values (and for higher moments) of these quantities are easy to obtain. These expressions are very suitable for studying the effect of a jumping mechanism. The probability generating functions, moreover, prove to be very useful in deriving approximate expressions for the tail probabilities of the corresponding quantities. We thereby show that determining the tail behaviour of a quantity from its probability generating function can be rather complex in a priority queueing system with priority jumps. Finally, we notice that subtle differences between jumping mechanisms may not only cause large differences in their results, but can also yield a major shift in the solution process.

References

[1] Jang, J., Shim, S., and Shin, B. (1997). Analysis of DQLT scheduling policy for an ATM multiplexer. *IEEE Communications Letters*, 1(6):175–177.

[2] Kleinrock, L. (1976). *Queueing systems volume II: computer applications.* Wiley & Sons, New York.

[3] Lim, Y. and Kobza, J. (1990). Analysis of a delay-dependent priority discipline in an integrated multiclass traffic fast packet switch. *IEEE Transactions on Communications*, COM-38(5):659–685.

[4] Maertens, T., Walraevens, J., and Bruneel, H. (2006). On priority queues with priority jumps. *Performance Evaluation*, 63(12):1235–1252.

[5] Maertens, T., Walraevens, J., and Bruneel, H. (2007a). A modified HOL priority scheduling discipline: performance analysis. *European Journal of Operational Research*, 180(3):1168–1185.

[6] Maertens, T., Walraevens, J., and Bruneel, H. (2008a). Performance comparison of several priority schemes with prioriy jumps. *Annals of Operations Research*, 180(3):1168–1185.

[7] Walraevens, J., Steyaert, B., and Bruneel, H. (2003). Performance analysis of a single-server ATM queue with a priority scheduling. *Computers and Operations Research*, 30(12):1807–1829.