Performance Evaluation of Priority based Schedulers in the Internet

LASSAAD ESSAFI[1] and GUNTER BOLCH[1]
[1]Institute of Computer Science
University of Erlangen-Nuremberg
Martensstrasse 3, D-91058 Erlangen, Germany
ldessafi@aol.com, bolch@informatik.uni-erlangen.de

31-Mar-2003

Abstract

This paper shows how the delay of jobs/packets using priority based scheduling mechanisms can be computed analytically. Static priorities and different types of time dependent priorities are considered. This paper also shows how these results are used in modern router design. Results for mono-processor and multi-processor architectures are given.
Keywords: priority queuing, multi-processor systems, quality of service

1 Introduction

In a router architecture queuing occurs when packets are received by a device's interface processor (input queue), and queuing may also occur prior to transmitting the packets to another interface (output queue) on the same device. A basic router is a collection of input processes that assemble packets as they are received, checking the integrity of the basic packet framing, one or more processors that determine the destination interface to which the packet should be passed and output processors that frame and transmit the packets on their next hop.

Priority scheduling represents a class of scheduling disciplines which can be used to provide differentiated services in the Internet. In addition to strict priority scheduling, already implemented in several router architectures, recent research on proportional differentiated services has shown that the waiting time priority scheduler is a promising mechanism for approximating the proportional delay differentiation model [1].

The aim of this paper is to give an overview of different priority based scheduling mechanisms and investigate their properties analytically. Two classes of priority based disciplines are discussed: static priorities in Section 2 and time dependent priorities with different variants in Section 3. We finally conclude in Section 4.

2 Static Priorities

One of the first queuing variations to be widely implemented was priority queuing. Here a fairly general model based on M/G/1 is used [9], but can in some cases be approximatively extended to the more general case of G/G/m. The results hold only in case of stationarity. In this queuing model we assume that an arriving packet belongs to a priority class r ($r = 1, 2, \ldots, R$). The priority of a packet is constant during its whole sojourn time in the router, that is why we refer to this class of priorities as static priorities. The next packet to be sent is the packet with the highest priority r. Within a priority class the queuing discipline is FCFS (First-Come-First-Served).

The mean waiting time $\overline{W}_r$ of an arriving packet C_r of the priority class r has three components [5, 6]:

1. The mean remaining service time $\overline{W}_0$ of the packet being served (if any).

2. The mean service time of the packets, found in the queue by the tagged packet on arrival, and that are served before it. These are the packets in the queue of the same and higher priority as the tagged packet.

3. Mean service time of packets that arrive at the system while the tagged packet is in the queue and are served before him. These are packets with higher priority than the tagged packet.

Note that we consider only the case where a packet being served is *not preempted* by an arriving packet with higher priority. Preemption is not considered in this paper because it is less relevant in the context of packet queuing.

We define:

$\overline{N}_{ir}$: Mean number of packets of class i found in the queue by the tagged packet C_r (with priority r) and being served before it,

$\overline{M}_{ir}$: Mean number of packets of class i which arrive during the waiting time of the tagged packet and being served before it.

Proceedings 17th European Simulation Multiconference
 ISBN 3-936150-25-7

Then the mean waiting time of class r packets in an M/G/1 system can be written as the sum of three components:

$$\overline{W}_r = \overline{W}_0 + \sum_{i=1}^{R} \overline{N}_{ir} \cdot \frac{1}{\mu_i} + \sum_{i=1}^{R} \overline{M}_{ir} \cdot \frac{1}{\mu_i} . \quad (1)$$

where μ_i is the service rate of the packets of class i[1]. For a multi-procesor system with m processors M/G/m ($m > 1$):

$$\overline{W}_r = \overline{W}_0 + \sum_{i=1}^{R} \frac{\overline{N}_{ir}}{m} \frac{1}{\mu_i} + \sum_{i=1}^{R} \frac{\overline{M}_{ir}}{m} \cdot \frac{1}{\mu_i} , \quad (2)$$

where $\overline{N}_{ir}$ and $\overline{M}_{ir}$ are given by:

$$\begin{aligned} \overline{N}_{ir} &= 0 \quad i < r , \\ \overline{M}_{ir} &= 0 \quad i \leq r , \end{aligned} \quad (3)$$

and, with Little's theorem resp. the formula for the mean number of jobs of class i which arrive during the mean waiting time $\overline{W}_r$:

$$\begin{aligned} \overline{N}_{ir} &= \lambda_i \overline{W}_i \quad i \geq r , \\ \overline{M}_{ir} &= \lambda_i \overline{W}_r \quad i > r, \end{aligned} \quad (4)$$

where λ_i denotes the arrival rate of class-i packets. Considering the utilization factor ρ_r in class r as

$$\rho_r = \frac{\lambda_r}{\mu_r} \quad (5)$$

equations (1) and (2) can be solved to obtain:

$$\overline{W}_r = \frac{\overline{W}_0}{(1-\sigma_r)(1-\sigma_{r+1})} , \quad (6)$$

where:

$$\sigma_r = \sum_{i=r}^{R} \rho_i . \quad (7)$$

Obviously the sum of utilization factors in all classes has to fulfill the condition (stability condition):

$$\sum_{i=1}^{R} < 1, \quad (8)$$

otherwise the system is unstable and the queue lengths become infinite.

Mean Remaining Service Time $\overline{W}_0$: The mean remaining service time $\overline{W}_0$ for poisson arrival [8] is given by:

$$\overline{W}_0 = P(\text{server is busy}) \cdot \overline{R} + P(\text{server is idle}) \cdot 0 , \quad (9)$$

with the main remaining service time $\overline{R}$ of a busy server (the remaining service time of an idle server is obviously zero). When a packet arrives, the packet in service needs $\overline{R}$ time units on the average to be finished. This quantity is also called the mean residual life [6] and is given by:[2]

$$\overline{R} = \frac{\overline{T_B^2}}{2\overline{T}_B} = \frac{\overline{T}_B}{2}(1 + c_B^2). \quad (10)$$

For an M/M/1 system ($c_B^2 = 1$), we obtain:

$$\overline{R}_{\mathrm{M/M/1}} = \overline{T}_B = \frac{1}{\mu} ,$$

which is related to the memoryless property of the exponential distribution. For multi-processor system with m processors we refer to the approximation in [6]:

$$\overline{R} \approx \frac{\overline{T_B^2}}{2m\overline{T}_B} \quad (11)$$

The probability that the link server is busy is for the mono-processor case given by the utilization ρ. For a muti-processor server with m processors, the probability P_m that all m processors are busy can be calculated as follows: Let p_k be the probability that k packets are being actually in the system

$$p_k = \begin{cases} p_0 \frac{(m\rho)^k}{k!} & \text{if} \quad k \leq m \\ p_0 \frac{m^m \rho^k}{m!} & \text{if} \quad k \geq m \end{cases} \quad (12)$$

$$p_0 = \left(\left(\sum_{k=0}^{m-1} \frac{(m\rho)^k}{k!} \right) + \frac{(m\rho)^m}{m!(1-\rho)} \right)^{-1} \quad (13)$$

For a system with m processors:

$$P_m = \sum_{k=0}^{\infty} p_k = \frac{(m\rho)^m}{m!(1-\rho)} p_0 \quad (14)$$

The exact solution is only valid for exponentially distributed service times, e.g. M/M/m-systems. For M/G/m-systems several approximations have been given in [5]:

1. $P_m = 1 - e^{-mp} \sum_{k=0}^{m-1} \frac{(m\rho)^k}{k!}$ for a utilization $\rho \leq 0.3$
2. $P_m = \rho^{m+1}/2$ for $0.3 \leq \rho \leq 0.7$
3. $P_m = \frac{\rho + \rho^m}{2}$ for a utilization higher than 0.7
4. $P_m = P_{m(M/M/m)}$ for all utilization values

We conclude this section by giving the mean remaining service time for different queuing systems [6][3]:

$$\overline{W}_{0,\mathrm{M/M/1}} = \sum_{i=1}^{R} \rho_i \frac{1}{\mu_i} \quad (15)$$

[1] throughout this paper we use the same notation as in Bolch et.al. [6]

[2] $\overline{T}_B$ is the mean service time and c_B^2 is the coefficient of variation

[3] please note that the presented results and the results in the following sections have been validated using simulations in [10, 11]

$$\overline{W}_{0,\mathrm{M/G/1}} = \sum_{i=1}^{R} \rho_i \cdot \frac{1+c_{B_i}^2}{2\mu_i} \tag{16}$$

$$\overline{W}_{0,\mathrm{GI/G/1,AC}} \approx \sum_{i=1}^{R} \rho_i \cdot \frac{c_{A_i}^2+c_{B_i}^2}{2\mu_i} \tag{17}$$

$$\overline{W}_{0,\mathrm{GI/G/1,KLB}} \approx \sum_{i=1}^{R} \rho_i \cdot \frac{c_{A_i}^2+c_{B_i}^2}{2\mu_i} \cdot G_{\mathrm{KLB}} \tag{18}$$

$$\overline{W}_{0,\mathrm{GI/G/1,KUL}} \approx \sum_{i=1}^{R} \rho_i \cdot \tag{19}$$

$$\frac{c_{A_i}^{f(c_{A_i},c_{B_i},\rho_i)}+c_{B_i}^2}{2\mu_i} \tag{20}$$

$$\overline{W}_{0,\mathrm{M/M/m}} = \frac{P_m}{m\rho} \sum_{i=1}^{R} \rho_i \cdot \frac{1}{\mu_i} \tag{21}$$

$$\overline{W}_{0,\mathrm{M/G/m}} \approx \frac{P_m}{2m\rho} \cdot \sum_{i=1}^{R} \rho_i \cdot \frac{1+c_{B_i}^2}{\mu_i} \tag{22}$$

$$\overline{W}_{0,\mathrm{GI/G/m,AC}} \approx \frac{P_m}{2m\rho} \cdot \sum_{i=1}^{R} \rho_i \cdot \frac{c_{A_i}^2+c_{B_i}^2}{\mu_i} \tag{23}$$

$$\overline{W}_{0,\mathrm{GI/G/m,KLB}} \approx \frac{P_m}{2m\rho} \sum_{i=1}^{R} \rho_i \cdot \tag{24}$$

$$\frac{c_{A_i}^2+c_{B_i}^2}{\mu_i} G_{\mathrm{KLB}} \tag{25}$$

$$\overline{W}_{0,\mathrm{GI/G/m,KUL}} \approx \frac{P_m}{2m\rho} \sum_{i=1}^{R} \rho_i \cdot \tag{26}$$

$$\frac{c_{A_i}^{f(c_{A_i},c_{B_i},\rho_i)}+c_{B_i}^2}{\mu_i} . \tag{27}$$

For $f(c_{A_i}, c_{B_i}, \rho_i)$,

$$f(c_A, c_B, \rho) = \begin{cases} 1, c_A \in \{0,1\}, \\ +\Big[\rho(14.1c_A - 5.9) + (-13.7c_A + 4.1)\Big] c_B^2 \\ +\Big[\rho(-59.7c_A + 21.1) + (54.9c_A - 16.3)\Big] c_B \\ +\Big[\rho(c_A - 4.5) + (-1.5c_A + 6.55)\Big] , \\ 0 \le c_A \le 1, \\ -0.75\rho + 2.775 \, , c_A > 1, \end{cases} \tag{28}$$

for $G_{\mathrm{KLB,GI/G/1}}$

$$G_{\mathrm{KLB}} = \begin{cases} exp\left(-\frac{2}{3} \cdot \frac{1-\rho}{\rho} \cdot \frac{(1-c_A^2)^2}{c_A^2+c_B^2}\right) , & 0 \le c_A \le 1 , \\ exp\left(-(1-\rho)\frac{c_A^2-1}{c_A^2+4c_B^2}\right) , & c_A > 1 . \end{cases} \tag{29}$$

and for $G_{\mathrm{KLB,GI/G/m}}$

$$G_{\mathrm{KLB}} = \begin{cases} exp\left(-\frac{2}{3}\frac{1-\rho}{P_m}\frac{(1-c_A^2)^2}{c_A^2+c_B^2}\right) , & 0 \le c_A \le 1, \\ exp\left(-(1-\rho)\frac{c_A^2-1}{c_A^2+4c_B^2}\right) , & c_A > 1, \end{cases} \tag{30}$$

3 Time Dependent Priorities

Static priorities are simple to implement in sofware and hardware because, to a make a scheduling decision, the scheduler needs only to determine the highest priority nonempty queue. On the other hand, a static priority scheme allows a misbehaving connection at highest priority to increase the delay and decrease the available bandwidth of connections at all lower priority levels. This leads in extreme cases to starvation of lower priority classes.

In many cases it is advantageous for a packet priority to increase with the time. This possibility can be considered if we use a priority function:

$$q_r(t) = \text{Priority of class } r \text{ at time } t \, .$$

Such systems are more flexible but need more expense for the administration. In this section, we investigate different types of time dependent priorities.

3.1 Class Dependent Increasing Rate

We refer to the same queuing model and assign each priority class a parameter b_r, which can be interpreted according to the the priority function

$$q_r(t) = (t - t_0) b_r \tag{31}$$

as the increasing rate (slope) of the priority in the class r, where $0 \le b_1 \le b_2 \le \ldots \le b_R$. This means that the priority of a higher class increases faster that the priority of a lower class. A packet enters the system at time t_0 and then increases its priority at the rate b_r.

In order to determine the mean waiting time of a packet C_r which arrives to the system at time t_0, we follow the same approach as for static priorities (Eqn. 1). We have to determine the mean number of packets $\overline{N}_{ir}$ of class i found in the queues by the tagged packet (belonging to the priority class r) upon its arrival and being served before it, and the mean number of packets $\overline{M}_{ir}$ of class i which arrive during the waiting time of the tagged packet and being served before it. For the mean remaining service times we can use the same formulae as for static priorities.

The set $\overline{\mathbf{N}}_{\mathbf{ir}}$: We first consider the packets of lower priority classes ($i < r$), which arrive to the queuing system before the tagged packet C_r (before t_0) and which are served before it. These packets are characterized by the following (see Figure 1):

- these packets arrived to the system at some point $-t_1$
- waiting time: $w_i(t_1)$ with $t_1 < w_i(t_1) < t_1 + t_2$
- have at time t_2 the same priority as C_r, which means $b_r t_2 = (t_1 + t_2) b_i$.

Now we compute $t_1 + t_2$, which is the period where the lower priority packets in N_{ir} in are served before C_r:

$$t_1 + t_2 = \frac{b_r t_1}{b_r - b_i}$$

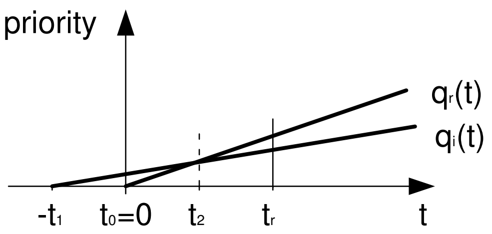

Figure 1: Priority functions with constant slopes (for the sets $\overline{N}_{ir}$)

$\overline{N}_{ir}$ can be written as:

$$\int_0^\infty \lambda_i P\{t < w_i(t) \leq \frac{b_r}{b_r - b_i} t_1\} dt \tag{32}$$

Eqn. (32) can be simplified using the substitution:

$$y = \frac{b_r}{b_r - b_i} t$$

and the fact

$$W_i = E[w_i] = \int_0^\infty 1 - P(w_i \leq x) dx$$

We finally get:

$$\overline{N}_{ir} = \lambda_i \int_0^\infty 1 - P(w_i \leq t) dt$$

$$-\lambda_i \int_0^\infty 1 - P(w_i \leq y) dy (1 - \frac{b_i}{b_r}) \tag{33}$$

hence

$$\overline{N}_{ir} = \lambda_i W_i \frac{b_i}{b_r} \text{ for all } i < r \tag{34}$$

$\overline{N}_{ir}$ is according to Little's theorem for all $i \geq r$

$$\overline{N}_{ir} = \lambda_i W_i \tag{35}$$

The set $\mathbf{\overline{M}_{ir}}$: Considering the tagged packet C_r (arrived at t_0), it is obvious that

$$\overline{M}_{ir} = 0 \text{ for all } i \leq r \tag{36}$$

because no packet of these lower priority classes will be served before C_r. For classes with higher priorities we have to consider all packets which arrive to the system after C_r, but which are served before it. These are according to Figure 2 all packets which arrive in the interval $[0, T_i)$. The crucial time T_i is determined by:

$$b_r W_r = b_i (W_r - T_i)$$

hence:

$$T_i = W_r \left(1 - \frac{b_r}{b_i}\right)$$

and:

$$\overline{M}_{ir} = \lambda_i T_i = \lambda_i \overline{W}_r \left(1 - \frac{b_r}{b_i}\right) \text{ for all } i > r \tag{37}$$

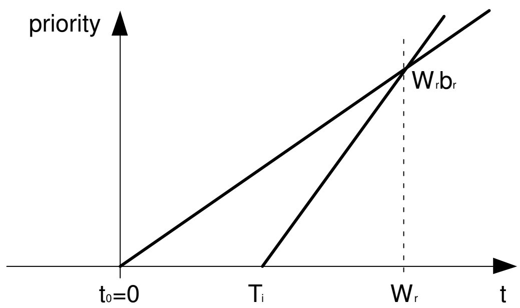

Figure 2: Priority functions with constant slope (for the sets $\overline{M}_{ir}$)

The mean waiting time $\mathbf{\overline{W}_r}$: After substituting the previous results (34), (35), (36) and (37) in (1) we get:

$$\overline{W}_r = \frac{\overline{W}_0 + \sum_{i=1}^{r-1} \rho_i \overline{W}_i \frac{b_i}{b_r} + \sum_{i=r}^{R} \rho_i \overline{W}_i}{1 - \sum_{i=r+1}^{R} \rho_i (1 - \frac{b_r}{b_i})} \tag{38}$$

We use the conservation law [6]:

$$\sum_{i=r}^{R} \rho_i \overline{W}_i = \rho \overline{W}_{FIFO} \tag{39}$$

and get a recursive formula for the mean waiting times:

$$\overline{W}_r = \frac{\overline{W}_{FIFO} - \sum_{i=1}^{r-1} \rho_i \overline{W}_i \left(1 - \frac{b_i}{b_r}\right)}{1 - \sum_{i=r+1}^{R} \rho_i (1 - \frac{b_r}{b_i})} \tag{40}$$

Relationship between $\mathbf{\overline{W}_0}$ and $\mathbf{\overline{W}_{FIFO}}$: In Section 2 the mean remaining service time $\overline{W}_0$ for a variety of queuing systems were given. The relationship between $\overline{W}_0$ and $\overline{W}_{FIFO}$ is given by:

$$\overline{W}_{FIFO} = \frac{\overline{W}_0}{1 - \rho} \tag{41}$$

Proof:
The mean waiting time of packet in a FIFO system has two components:

1. the mean remaining service time $\overline{W}_0$ of the packet in service (if any),
2. the sum of the mean service times of the packets in the queue.

This sum can be written as:

$$\overline{W} = \overline{W}_0 + \overline{Q} \cdot \overline{T}_B \tag{42}$$

According to Little's theorem, the mean number of packets in the queue is:

$$\overline{Q} = \lambda \cdot \overline{W}$$

From Eqn. (42) we obtain:

$$\overline{W} = \overline{W}_0 + \lambda \cdot \overline{W} \cdot \overline{T}_B = \overline{W}_0 + \rho \cdot \overline{W}$$

and finally Eqn. (41). q.e.d.

3.2 Variants of the Priority Function

The first variant of the priority function in Eqn. 31 consists in assigning an exponent r_s (i.e. 2) to the slope b_r. The resulting priority function is then:

$$q_r^{r_s}(t) = (t - t_0) b_r^{r_s} \tag{43}$$

The mean waiting time of a packet of the class r is:

$$\overline{W}_r = \frac{\overline{W}_{FIFO} - \sum\limits_{i=1}^{r-1} \rho_i \overline{W}_i \left(1 - \left(\frac{b_i}{b_r}\right)^{r_s}\right)}{1 - \sum\limits_{i=r+1}^{R} \rho_i \left(1 - \left(\frac{b_r}{b_i}\right)^{r_s}\right)} \tag{44}$$

The use of the exponent r_s leads to a better separation of the different priority classes and it is possible to theoretically cover the whole spectrum of queueing mechanisms: from a strict differentiation of the priority classes as done with static priorities to no differentiation as it is the case with FIFO.

The second variant is the priority function:

$$q_r^n(t) = (t - t_0)^n b_r \tag{45}$$

we get for the mean waiting time:

$$\overline{W}_r = \frac{\overline{W}_{FIFO} - \sum\limits_{i=1}^{r-1} \rho_i \overline{W}_i \left(1 - \left(\frac{b_i}{b_r}\right)^{1/n}\right)}{1 - \sum\limits_{i=r+1}^{R} \rho_i \left(1 - \left(\frac{b_r}{b_i}\right)^{1/n}\right)} \tag{46}$$

The exponent n is used to weight the waiting time in the system, which also leads to a fine differentiation of the classes. For the limits $n \to \infty$ and $n \to 0$, the system tend to static priorities and FIFO respectively. The combination of r_s and n, which does not cause any mathematical problems, allows more variation possibilities in the specified spectrum.

3.3 Class Dependent Starting Priorities

A possibility to reduce the waiting time of certain packets is to assign to each packet a class dependent starting priority r_r. With a slope 1 for the priority functions of all classes and with

$$0 \leq r_1 \leq r_2 \leq \ldots \leq r_R$$

we get the following form for the priority functions:

$$q_r(t) = r_r + t - t_0 \tag{47}$$

In this case again, the mean waiting time $\overline{W}_r$ is dependent on the mean remaining service time $\overline{W}_0$ and the sets $\overline{M}_{ip}$ and $\overline{N}_{ip}$, which have to be determined.

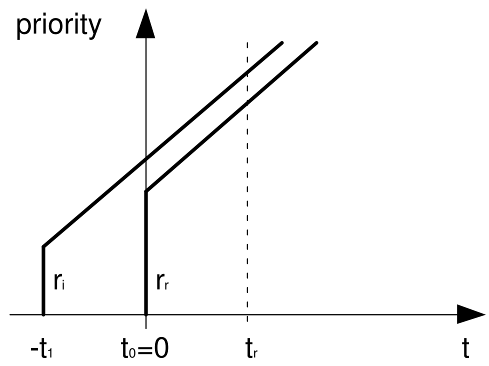

Figure 3: Determination of $\overline{N}_{ir}$ for class dependent starting priorities

The set $\overline{\mathbf{N}}_{\mathbf{ir}}$: A packet C_r, which arrives to the system at time t_0 with the starting priority r_r cannot be served before the packets already buffered in the queues of the same or higher classes, that's why:

$$\overline{N}_{ir} = \lambda_i \overline{W}_i \text{ for all } i \geq r \tag{48}$$

For lower priority classes, the packets (indexed with i) already buffered in the queues at time t_0 and which will be served before C_r are characterized by (see Figure 3):

- arrival time: $-t_1$
- starting priority: r_i
- priority at time t_0: $q_i(t_0 = 0) = r_i + t_1$
- waiting time: $w_i(t_1)$ with $t_1 < w_i(t_1) < \infty$
- $q_i(t_0 = 0) \geq r_r$, because they are served before C_r

These packets will get served before C_r, that's why:

$$t_1 \geq r_r - r_i$$

Hence for $(i < r)$ we get:

$$\overline{N}_{ir} = \int_{r_r - r_i}^{\infty} \lambda_i P\{t < w_i(t) \leq \infty\} dt \tag{49}$$

After substitution and using:

$$W_i = E[w_i] = \int_0^{\infty} 1 - P(w_i \leq t) dt \tag{50}$$

we finally get:

$$\overline{N}_{ir} = \lambda_i W_i - \lambda_i \int_0^{r_r - r_i} P(w_i > t) dt \text{ for all } (i < r) \tag{51}$$

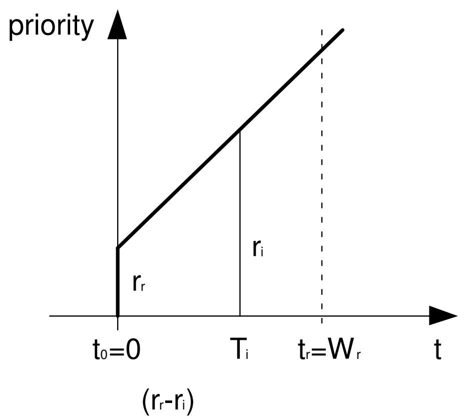

Figure 4: Determination of $\overline{M}_{ip}$ for class dependent starting priorities

The set $\overline{\mathbf{M}}_{\mathbf{ir}}$: Considering the tagged packet C_r (arrived at t_0), it is obvious that

$$\overline{M}_{ir} = 0 \text{ for all } i \leq r \tag{52}$$

because no packet of these lower priority classes will be served before C_r.

For classes with higher priorities we have to consider all packets which arrive to the system after C_r, but which are served before it. These are according to Figure 4 all packets which arrive in the interval $[0, T_i)$, because:

$$q_i(t_r) \geq q_r(t_r) = r_r + t_r$$

We use

$$r_i + W_r - T_i = r_r + W_r$$

to calculate T_i and get:

$$T_i = r_i - r_r$$

Hence:

$$\overline{M}_{ir} = \lambda_i \int_0^{r_i - r_r} P(w_r > t)dt \text{ for all } (i > r) \tag{53}$$

Substituting (48), (48) and (53) in (1) and using the conservation law (39) gives:

$$\overline{W}_r = \overline{W}_{FIFO} - \sum_{i=1}^{r-1} \rho_i \int_0^{r_r - r_i} P(w_i > t)dt +$$

$$\sum_{i=r+1}^{R} \rho_i \int_0^{r_i - r_r} P(w_r > t)dt \tag{54}$$

To determine the waiting probabilities $P(w_k < t)$ $(k = i, r)$, which cannot be calculated exactly, we refer here to two approximations proposed in [7] for M/G/1-systems:

Approximation 1: is a heavy traffic approximation of the mean waiting time is given by:

$$\overline{W}_r = \overline{W}_{FIFO} - P_m \sum_{i=1}^{R} \rho_i (r_r - r_i) \qquad \rho \to 1 \tag{55}$$

Approximation 2: This approximation of the mean waiting time is valid for $0 \leq \rho < 1$

$$\overline{W}_r \Big(1 - \sum_{i=r+1}^{R} \rho_i \big(1 - e^{P_m (r_r - r_i)/\overline{W}_r}\big)\Big) =$$

$$\overline{W}_{FIFO} - \sum_{i=1}^{r-1} \rho_i \overline{W}_i \big(1 - e^{P_m (r_i - r_r)/\overline{W}_i}\big) \tag{56}$$

Given $\overline{W}_1$ all other mean waiting times can be determined recursively, where for each i the solution of the equation (56) has to be computed numerically.

3.4 Starting Priorities r_p with Independent Increasing Rates b_q

We extend here the considered system, where not only a starting priority r_p for each class is defined, but also a class dependent increasing rate. If both parameters are dependent on each other, that means a class with a high starting priority also has a relatively high increasing rate, we then define the priority functions as:

$$q_r(t) = r_r + (t - t_0) b_r \tag{57}$$

These functions don't have any important differences if compared to the priority functions (31) and (47) [5]. If the two parameters are independent of each other, so we will characterize each class with the two parameters: p for the starting priority and q for the increasing rate, with

$$0 \leq r_1 \leq r_2 \leq \ldots \leq r_P \text{ and}$$

$$0 \leq b_1 \leq b_2 \leq \ldots \leq b_Q$$

A combination of a high starting priority and a low increasing rate (and vice versa) is here possible. This system is the most general one and covers all cases between FIFO- and static priority systems. The priority function is defined by:

$$q_{pq} = r_p + (t - t_0) b_q \tag{58}$$

All parameters like C_p, ρ_p and λ_p have now to extended to $C_{pq}, \rho_{pq}, \lambda_{pq}$ etc. Furthermore we define $\overline{N}_{ij,pq}$ and $\overline{M}_{ij,pq}$ as:

$\overline{N}_{ij,pq}$ mean number of packets with starting priority r_i and increasing rate b_j, which are already buffered in the queues and are will be served before the packet C_{pq}, with starting priority r_p and increasing rate b_q,

$\overline{M}_{ij,pq}$ mean number of packets with starting priority r_i and increasing rate b_j, which arrive during the waiting time of C_{pq} and which are to be served before the packet C_{pq}.

The mean waiting time of the packet C_{pq} is given by:

$$\overline{W}_{pq} = \overline{W}_0 + \sum_{i=1}^{P} \sum_{j=1}^{Q} \frac{N_{ij,pq} + M_{ij,pq}}{m} \overline{T}_{B_{ij}} \tag{59}$$

For $\overline{N}_{ij,pq}$ we have to consider four cases:

- a) $i \geq p$ and $j \geq q$

$$\overline{N}_{ij,pq} = \lambda_{ij} \overline{W}_{ij} \tag{60}$$

- b) $i \geq p$ and $j < q$ (see Figure 5):

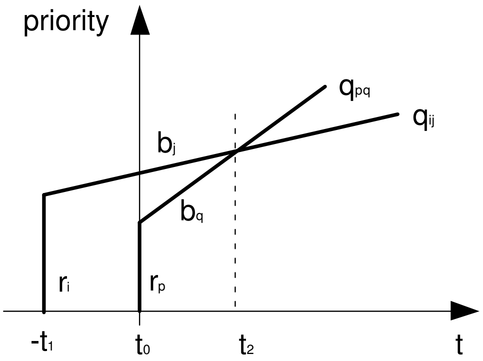

Figure 5: Determination of $N_{ij,pq}$ for $i \geq p, j < q$

The unit of time in which packets C_{ij} are served before C_{pq} can be determined using:

$$r_p + b_2 t_2 = r_i + b_j(t_1 + t_2)$$

Then:

$$\overline{N}_{ij,pq} = \int_0^\infty \lambda_{ij} P\{1 < w_{ij}(t)$$

$$\leq \frac{r_i - r_p}{b_q - b_j} + \frac{b_q}{b_q - b_j} t_1\} dt$$

After substitution and using:

$$\overline{W}_{ij} = E[w_{ij}] = \int_0^\infty 1 - P(w_{ij} \leq x) dx$$

we get:

$$\overline{N}_{ij,pq} = \lambda_{ij} \overline{W}_{ij} \frac{b_j}{b_q} \text{ for all } i \geq p, j < q \tag{61}$$

- c) $i < p$ and $j \geq q$ (see Figure 6):

It holds for the relevant packets at $t = 0$:

$$q_{ij}(t = 0) = r_i + t_1 b_j \geq r_p$$

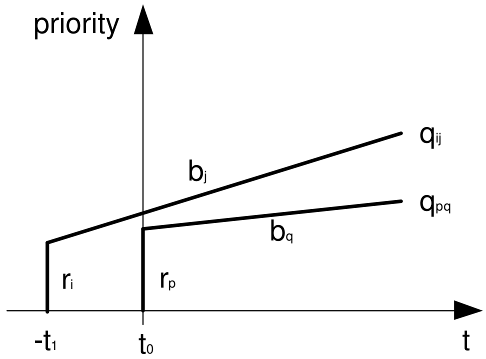

Figure 6: Determination of $N_{ij,pq}$ for $i < p, j \geq q$

with the crucial time point

$$t_1 \geq \frac{r_p - r_i}{b_j}$$

so that:

$$\overline{N}_{ij,pq} = \lambda_{ij} \overline{W}_{ij} - \lambda_{ij} \int_0^{\frac{r_p - r_i}{b_j}} P(w_{ij} > t) dt$$

$$\text{for all } i < p, j \geq q \tag{62}$$

- d) $i < p$ and $j < q$ (see Figure 7):

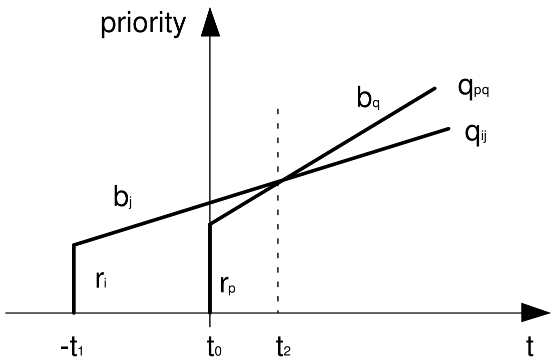

Figure 7: Determination of $N_{ij,pq}$ for $i < p, j < q$

It holds for the packets which are served before C_{pq}:

$$r_p + b_q t_2 = r_i + b_j(t_1 + t_2)$$

With

$$t_1 + t_2 = \frac{r_i - r_p}{b_q - b_j} + \frac{b_q}{b_q - b_j} t_1$$

we get with a similar way as in b):

$$\overline{N}_{ij,pq} = \lambda_{ij} \overline{W}_{ij} \frac{b_j}{b_q} \text{ for all } i < p, j < q \tag{63}$$

In order to determine the sets $\overline{M}_{ij,pq}$ we make here the difference between four cases whereas:

$$\overline{M}_{ij,pq} = 0 \text{ for all } i \leq p, j \leq q$$

All other cases can be treated similarly because the crucial time point t_{ij} is determined by

$$r_i + (W_{pq} - t_{ij})b_j > r_p - W_{pq}b_q$$

hence

$$t_{ij} < \frac{r_i - r_p}{b_j} + W_{pq}\left(1 - \frac{b_q}{b_j}\right)$$

The mean number of packets, which arrive in the time interval $[0, t_{ij})$ and which are served before C_{pq} is then:

$$\overline{M}_{ij,pq} = \lambda_{ij} \int_0^{t_{ij}} P(w_{pq} > t)dt$$

$$\text{for all } i > p \text{ and for all } i \leq p, j > q \qquad (64)$$

The determined sets $\overline{N}_{ij,pq}$ and $\overline{M}_{ij,pq}$ can now be substituted in (59) together with the conservation law to get the following form for the mean waiting time:

$$\begin{aligned}\overline{W}_{pq} &= \overline{W}_{FIFO} - \textstyle\sum_{i=1}^{P}\sum_{j=1}^{q-1} \rho_{ij}\overline{W}_{ij}\left(1 - \frac{b_j}{b_q}\right) \\ &\quad - \textstyle\sum_{i=1}^{p-1}\sum_{j=q}^{Q} \rho_{ij} \int_0^{\frac{r_p - r_i}{b_j}} P(w_{ij} > t)dt \\ &\quad + \textstyle\sum_{i=p+1}^{P}\sum_{j=1}^{Q} \rho_{ij} \int_0^{t_{ij}} P(w_{pq} > t)dt \\ &\quad + \textstyle\sum_{i=1}^{P}\sum_{j=q+1}^{Q} \rho_{ij} \int_0^{t_{ij}} P(w_{pq} > t)dt \quad (65)\end{aligned}$$

The unknown probabilities have to be substituted by the proposed approximations to get a recursive equation system, that can be solved numerically.

3.5 Dynamic Priorities with Class Dependent Deadlines

Another strategy for scheduling packets is based on deadlines which are assigned to each packet[4]. R packet classes are defined with a parameter G_i for each class, with:

$$G_1 > G_2 > \ldots > G_R$$

The parameters G_i define the time period, which may maximally be elapsed to serve a packet "in time". Packets with the lowest deadlines have the highest priority and vice versa. According to the priority function:

$$q_r(t) = \begin{cases} (t - t_0)/(G_r - t + t_0) & \text{if } t_0 < t \leq G_r + t_0 \\ \infty & \text{if } G_r + t_0 \leq t < \infty \end{cases}$$

the priorities increase faster, whenever the deadline of a packet nears. If the deadline is reached, the packet gets an infinite priority, which forces the system to serve the packet (see Figure 8). Packets with a positive infinite priority are served in a FIFO order.

The approach to calculate the mean waiting times is not very different from the methods used for the other strategies. To determine the set $\overline{N}_{ir}$ we divide the packets into two categories, which are considered separately:

[4]this strategy is also known as Earliest Deadline First (EDF)

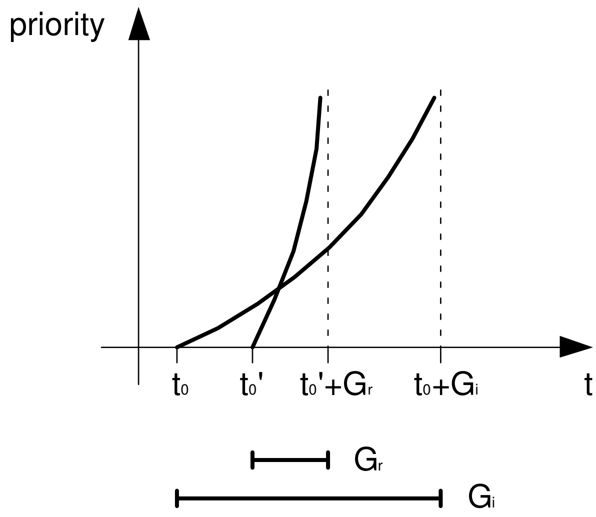

Figure 8: Class dependent deadlines

1. packets, which get an infinite priority after the considered packet C_r does. If $-t_1$ is their arrival time, so
$$G_i - t_1 > G_r$$
2. packets which get an infinite priority at the latest until G_r (at $t_0 = 0$, arrival of C_r), which means
$$G_i - t_1 \leq G_r$$

Using the priority functions we can deduce the sets $\overline{N}_{ir}(1)$ and $\overline{N}_{ir}(2)$. $\overline{M}_{ip}$ can also be determined according to the known methods we used for the other strategies. Substituting these sets in Eq. (1) and using the conservation law gives:

$$\begin{aligned}\overline{W}_r = \overline{W}_{FIFO} &+ \sum_{i=r+1}^{R} \rho_i \int_0^{T_i} 1 - P(w_r \geq G_r)dt \\ &- \sum_{i=1}^{r-1} \rho_i \frac{G_i - G_r}{G_i} \int_0^{G_i - G_r} P(w_i > x)dx \quad (66)\end{aligned}$$

with

$$T_i = \overline{W}_r \frac{G_r - G_i}{G_r} \qquad (67)$$

For the unknown probabilities we may again use one of the two proposed approximations.

4 Conclusion

In this paper we investigated two classes of priority based scheduling mechanisms: static priorities and time dependent priorities. In all cases we showed how the mean waiting times of packets of different classes can be computed analytically. We presented results for different arrival and service time distributions. Mono- and multi-processor routers were considered. These results have been successfully applied

for the characterization and analysis of proportional differentiated services [12, 13].
It is still important to investigate how the scheduler parameters have to be chosen, if quality of service profiles for the different traffic classes are given (design problem). It is also interesting to investigate the behavior of the priority based schedulers, when self similar traffic and correlated inter-arrival times are considered.

References

[1] *C.Dovrolis, D.Stiliadis, P.Ramanathan:* "Proportional Differentiated Services: Delay Differentiation and Packet Scheduling", ACM SIGCOMM '99, Cambridge, MA, Sep. 1999

[2] *P.Ferguson, G.Huston:* "Quality of Service: Delivering QoS on the Inernet and in Corporate Networks", Wiley Computer Publishing, 1998

[3] *D.Black, S.Blake, M.Carlson, E.Davies, Z.Wang and W.Weiss:* "An Architecture for Differentiated Services", IETF RFC 2475, Dec. 1998

[4] *K.Nichols, S.Blake, F.Baker, D.Black:* "Definition of the Differentiated Services Field (DS Field) in IPv4 and IPv6 Headers", IETF RFC 2474, Dec. 1998

[5] *G.Bolch, W.Bruchner:* Analytische Modelle symmetrischer Mehrpro-zessoranalagen mit dynamischen Prioritäten. Elektronische Rechenanlagen, 26(1), page 12-19, 1984

[6] *G.Bolch, S.Greiner, H.de Meer, K.S.Trivedi:* Queueing Networks and Markov Chains : Modeling and Performance Evaluation with Computer Science Applications, Wiley, New York, 1998

[7] *B.Walke:* Realzeitrechnermodelle; Theorie und Anwendungen. Oldenburg Verlag, München 1978

[8] *L.Kleinrock:* Queueing Systems - Volume I , Wiley, 1975

[9] *L.Kleinrock:* Queueing Systems - Volume II , Wiley, 1976

[10] *J.Kulbatzski, J,Liebeherr:* Entwicklung und Implementierung neuer Algorithmen für das Programmpaket PRIORI zur Leistungsbewertung von Mehrprozessorsystemen mit Prioritäten. Studienarbeit, University of Erlangen, 1987

[11] *J.Kulbatzski:* Das Programmsystem PRIORI - Erweiterung und Validierung mit Simulationen. Diplomarbeit, University of Erlangen, 1989

[12] *L.Essafi, G.Bolch, A.Andres:* "An Adaptive Waiting Time Priority Scheduler for the Proportional Differentiation Model", ASTC HPC'01, Seattle, Apr. 2001

[13] *L.Essafi, G.Bolch, H.de Meer:* "Dynamic Priority Scheduling for Proportional Delay Differentiated Services", MMB, Aachen, Sep. 2001