Computer Science Journal of Moldova, vol.**15**, no.2(44), 2007

# Generalized Priority Models for QoS and CoS Network Technologies *

Gh. Mishkoy, S. Giordano, A. Bejan, O. Benderschi

### Abstract

The variety of priority queueing systems with random switchover times is suggested in this paper. Such systems represent generalized models for a wide class of phenomena which involve queueing and prioritization and are considered in QoS and CoS network problems. The classification of such systems is given and methods of their analysis are discussed. Specialists in QoS and CoS technologies may find such models adequate and appropriate for the network traffic analysis.

**1991 Mathematics Subject Classication:** Primary 90B22; Secondary 68M22, 68M10.

**Keywords:** Priority queues, switchover times, Quality of Service (QoS), Class of Service (CoS), prioritization, traffic characteristics

*Quality of Service* (QoS) and *Class of Service* (CoS) technologies play nowadays a crucial role in the analysis of a network traffic, which is highly diverse and may be characterized in terms of *bandwidth*, *delay*, *loss*, and *availability*. Some more specific characteristics can also be considered.

Most of the network traffic is IP-based today. On the one hand it is beneficial, as it provides a single transport protocol and it simplifies maintaining of the hardware and software products. However, IP-based technologies have some drawbacks. First of all, under the IP protocol network packets are delivered through the network without taking any specific path. This results in the unpredictability of the quality of service in such networks.

*This work was done under support of the SCOPES grant IB7320-110720

However, today, networks deal with so many types of traffic, that these may interact in a very unfavorable manner while being transmitted through the network. QoS and CoS technologies serve to ensure that diverse applications can be properly supported in an IP-network, i.e. see [21]. This is achieved by distinguishing between different types of data and by managing them using the mechanisms of data prioritization.

We consider in this paper a diverse class of priority queueing systems involving switching to describe, model and analyze phenomena which involve prioritized queueing and may take place in the studied or designed network. We suggest that some performance characteristics of such priority queueing systems can be used for estimating and providing a respective Quality of Service.

In the following section we discuss briefly the QoS and CoS methodologies and their applications in analyzing and modeling networks.

We further discuss the priority queueing disciplines in details, then introduce the most important characteristics of such systems and indicate on the methods of their analytical and numerical study. We also give a brief description of the imitation modeling of such priority systems.

In the last section we consider an example of usage of such systems, and, in particular, we discuss the benefits of using them for obtaining QoS in WLANs.

# 1 QoS and CoS methodology in network traffic analysis

## 1.1 Quality of Service and Class of Service

Quality of Service is a general concept referring to the capability of a network to provide better service to selected network traffic over various technologies, including *Frame Relay*, *Asynchronous Transfer Mode* (ATM), *Ethernet* and *802.1 networks*, *SONET*, and IP-routed networks that may use any or all of these underlying technologies (e.g. [7]).

Define by a *flow* in a broad sense a combination of packets passing through a network. Basically, QoS enables to provide in a network a better service to certain flows by assigning the higher priority of a flow or limiting the priority of another. This can be done in different ways: mostly by designing corresponding queue management mechanisms.

One can represent the basic QoS architecture by the following three components and steps [7]:

- QoS marking techniques for coordinating QoS from end-to-end between network elements

- QoS within a single network element (e.g. queueing, scheduling, traffic-shaping tools)

- QoS policy, management and accounting functions to control and administer end-to-end traffic across a network

We refer in this paper mostly to a QoS of a single network element (i.e. to a second step of the QoS providing architecture scheme given above).

QoS within a single network element, or node, can be specified by a *congestion management*, *queue management*, *link efficiency*, and shaping/policing tools.

The Class of Service concept is a concept of the flow network traffic division into different classes. This concept provides class-dependent service to each packet in a flow, depending on which priority class it does belong to (see [24]). CoS provides end-to-end prioritization for frame relay and ATM traffic over IP networks. In a framework of CoS traffic is prioritized by setting the *Differentiated Services* code in the header of an IP data packet.

## 1.2 Prioritization in Information Systems

As we saw, the prioritization plays the crucial role in QoS and CoS technologies.

In information networks it is desirable to provide shorter waiting times for control packets (packets that contain information about network status), voice connection packets, and packets associated with messages which should be delivered urgently.

There are many ways to attribute preferences. However, on a conceptual level, there are not so many ways to provide preferential service in a queueing system or queueing network. In Section 2 we describe a wide range of service disciplines in priority queueing systems involving switching between flows.

For examples and more account on prioritization and its forms the reader is referred to [2]. Description of some queueing disciplines implemented at nodes of an ad hoc network can be found in [16]. QoS in ad hoc networks and mechanisms of data prioritization in such networks is discussed in [1], [25] and references therein.

### 1.3 Priority Queueing Systems in QoS and CoS analysis, modeling and design of networks

The mathematical models of queueing systems play an important role in analysis, modeling and design of various networks, including Wireless Local Area Networks (WLAN). The IEEE 802.11 standards, widely used in WLAN, are playing a more and more important role in building of the concepts of the Next Generation of Mobile Networks. Some specific queueing models are still proposed for network management and performance analysis based on mentioned technologies (see, e.g., [20]).

It appears that one of the important problems on the way to next generations of Mobile Networks will be a problem of providing enhanced mechanisms for the delivery of QoS and CoS facilities. The QoS is very relevant in WLAN, due to the growing demand, even in the case of mobile users, for multimedia applications, such as streaming video and teleconferencing. Recently pursued standardization efforts in IEEE 802.11e attempt to provide a level of service differentiation by statically associating different QoS parameters for pre-defined traffic classes, while CoS enables more predictable traffic delivery by assigning different delivery status for each application. For example, a first priority label can be assigned to data application which requires faster turnaround, such as mission-critical data transaction, video or voice transmission, etc. A lower priority label is assigned to less time sensitive traffic, such as e-mail or web-surfing.

To summarize, there are two ways of achieving a certain level of quality of service in networks: (i) by increasing bandwidth (which is not always possible), and (ii) by adding complicated QoS and CoS traffic management mechanisms.

What do we offer? We offer the modeling of the processes which take place at the nodes of a network (or any other phenomena involving prioritized queueing) by generalized priority queueing systems with random switchover times, where appropriate. QoS parameters defined to measure service quality include traditional parameters such as *latency* (delay and delay jitter), *packet loss-rate*, and *throughput* (allocated bandwidth). There are also parameters that are more related to wireless networks, as *varying channel conditions*. We believe, that these parameters can be estimated more appropriately by representing all the processes involving queueing and waiting phenomena and taking place in a network (network nodes' processes, switching). Analysis of the performance characteristics of such queueing systems can significantly help in understanding of the network design, analysis and modeling in order to provide higher QoS level.

Thus, we do not point any attention on the traffic management mechanisms. Assuming that a certain mechanism is chosen to be considered we only provide a way of representing any prioritized queueing process and suggest that performance characteristics of the service process in such queueing system may be used in estimation of the end QoS at the level of a network by estimating QoS's within network nodes.

# 2 Priority queueing systems with switchover times

## 2.1 Introduction

Priority queueing systems form a large class of queueing systems where the incoming requests are to be distinguished by their importance. Such systems represent adequate models of many aspects of everyday life, when a preferential service is to be granted to certain kinds of requests (demands or customers). Priority queueing systems have also found important applications in the modeling and analysis of computer and communication systems: packets transfer and routing in computer networks, distributed operations and calculations (multiprocessor OS's, etc), telephone switching systems and mobile phone networks. Some civil services (surgeries, ambulances, fires, etc.) can also be modeled using the concept of priority queueing systems.

The general rule of service in priority queueing systems is as follows: the requests which are in the system and have a higher priority should be served before those that have lower priorities. However, the mode of the device's behavior in such systems may essentially diversify them. In addition, there are systems where device needs some time to switch itself from the servicing of one kind of requests to another. All this gives a great variety of the considered systems. Accordingly to these phenomena the description and classification of the priority queueing systems is given below in the great generality.

## 2.2 Notations, systems description and classification

The classification given here takes its origin from the works of Klimov and Mishkoy [17], and Bejan and Mishkoy [4].

Consider a queueing system with a single device and $r$ classes of incoming requests, denoted by *class 1*, *class 2*, ..., *class r*, each having its own flow of arrival and waiting line. Requests of a particular class are served on one of the two following bases within their own line:

- a first-in-first-out basis (FIFO);

- a last-in-first-out basis (LIFO).

Suppose that the time periods between two consecutive arrivals of the requests of the class $i$ are distributed identically and have a cumulative distribution function (cdf) $A_i(t)$, $i = 1, \ldots, r$. Similarly, suppose that the service time of a customer of the class $i$ is a random variable (rv) $B_i$ with a cdf $B_i(t)$, i.e.

$$\mathbb{P}(B_i \leq t) = B_i(t), \ i = 1, \ldots, r.$$

For conciseness let us call the requests of the class $i$ by *$i$-requests.* We say that $i$-requests have a higher priority than $j$-requests if $1 \leq i < j \leq r$. Thus, 1-requests are the requests of the highest priority, whereas $r$-requests are of the lowest one. Device gives a preference in service to the requests of the highest priority among those presented in the system.

However, some time is needed for the device to proceed with a switching from one line of requests to another. This time is considered to be a random variable and we say that $C_{ij}$ is the time to switch from the service of $i$-requests to the service of $j$-requests, $1 \leq i \leq r$, $1 \leq j \leq r$, $i \neq j$. Refer further to $C_{ij}$ as *$ij$-switchover time* with a cdf $C_{ij}(t)$.

Sometimes it is plausible to view the temporal structure of the switchover time $C_{ij}$ as a sum of two independent periods:

$$C_{ij} = T_i + S_j, \ i \neq j, \tag{2.1}$$

where $T_i$ is a (random) time of **t**ermination of all service procedures referring to the class $i$, and $S_j$ is a (random) time of the arrangements the device may need to **s**tart servicing the $j$-requests. Technically, this phenomenon may be imagined as device's passing through a special *neutral* or *null state* – while proceeding with the $ij$-switching the device needs the time $T_i$ to get to the neutral state from class $i$, and it needs the time $S_j$ to get further to the class $j$ from the neutral state. We shall call such switching policy by *neutral state switching.* Under this policy the cdf's of rv's $\{T_i\}_{i=1}^r$ and $\{S_i\}_{i=1}^r$ will be some known families of functions $\{T_i(t)\}_{i=1}^r$ and $\{S_i(t)\}_{i=1}^r$.

### 2.2.1 Disciplines of service

Consider two disciplines of service — both traditional in the theory of priority queues: *preemptive service discipline* and *non-preemptive service discipline.* It is assumed under the former discipline that any request of the priority higher than the one that is being served interrupts the service process and requires device's switching to its class immediately. Under the latter discipline, the request of a lower priority level will receive a complete service after which the device will proceed with the switching, if needed. In both cases, on completion of service of the requests of some class, the device will be ready to move to the non-empty queue corresponding to the class of the highest priority level presented in the system at that moment.

**Preemptive service discipline.** Consider different scenarios in regard to the request whose service was interrupted:

1. *preemptive resume policy* — the interrupted request will be served the residuary period of time after device's return, i.e. the time which this request would have been served, if its service was not interrupted, from the moment of the interruption.

2. *repeat again* policies:

   - *preemptive identical repeat policy* — the interrupted request will be served again after device's return. The service time will coincide with the complete time this request would have been served if its service was not interrupted.
   - *preemptive non-identical repeat policy* — exactly as in the previous policy, but the repeat service time is new, though distributed in accordance with corresponding service law, i.e. having cdf $B_i(t)$ if the request to be served again is from class $i$.

3. *preemptive loss policy* — the interrupted requests will be lost and removed from the system.

**Non-preemptive service discipline.** There will be no immediate interruptions of requests' services under this discipline. Yet, on completion of service of each request (of several requests within a line), the device is ready to move to the non-empty queue with the highest priority level requests, if any are presented in the system and are waiting to be served. Instead of the term *non-preemptive service discipline* one can, following Gaver [9], use another name for this discipline — *postponable priority service discipline.*

The postponable priority service discipline can be of different kinds, as how the switching to higher priority requests is postponed:

1. *request postponable priority service discipline* — on completion of service of any request, the device is ready to switch to the non-empty queue of the higher priority requests.
2. - *exhaustive postponable priority service discipline* — the device will be ready to switch to the non-empty queue of the higher priority requests only and only when the queueing line of requests, which are being served at the moment, becomes empty.
   - *gated postponable priority service discipline* — exactly as in the *exhaustive postponable* discipline with the difference that the device will only serve those requests which came in the system before the interrupting ones.

### 2.2.2 Switching

One should take into account that some of the incoming demands may find the device switching to the requests of lower priority. Therefore, by analogy with the service process disciplines, distinguish between the following switching process disciplines: *preemptive switching discipline*, *preemptive neutral state switching discipline*, *non-preemptive switching discipline*, *non-preemptive neutral state switching discipline.*

**Preemptive switching.** Under the *preemptive switching* and *preemptive neutral state switching* disciplines any $ij$-switching will be immediately interrupted by $k$-requests, if and only if $k < j$, i.e., if some higher priority requests enter the system. After interruption a new switching to these requests is initiated. The two switching disciplines differ only in the absence/presence of the special null state — an intermediate device's state while switching (see definition of a neutral state on p. 223).

Sometimes it is plausible to consider the preemptive type of switching involving neutral state, formally as in (2.1), where the termination works $T_i$ are never interrupted. Call such type of switching *pseudo-preemptive switching.*

**Non-preemptive switching.** Under the *non-preemptive switching* and *non-preemptive neutral state switching* disciplines no switching can be interrupted by higher priority demands. The latter discipline differs from the former one in the existence of an intermediate switching state — neutral state, as introduced above.

Consider the *non-preemptive neutral state switching discipline* and recall that the structure of the switching consists in this case of two paths, as given by (2.1). Suppose that the device was found by a $k$-request switching to the $j$-requests, where $k < j$, i.e. realizing some $ij$-switching of the length $C_{ij}$. This moment could fall either on one of the following two periods: switching to the null state (of the length $T_i$) or switching from the null state (of the length $S_j$). Therefore consider the following two subdisciplines:

- *normal switching* — the switching to the $k$-requests will be made either after switching to the null state from $i$-requests (and then its duration will be $S_k$) or after the switching from the null state to the $j$-requests (and then its duration will be $C_{jk}$).

- *postponable switching* — the switching to the interrupting $k$-requests is possible only after the $ij$-switching is completed (and lasts then the time $C_{jk}$).

### 2.2.3 Behavior of the device in the idle state

We move now to the specifications of the device's regimes in the idle state. First, regardless the regime, let us assume that the device needs some *warming time* to proceed with the switching or servicing when the first customer comes in the empty system, i.e. after a *period of* *idleness.* This warming time is a random variable $W$ with a cdf $W(t)$. If the warming time is equal to zero (the device requires no warming), then $W(t) \equiv H(t)$, where $H(t)$ is the Heaviside function.

Following the tradition which takes its origin from the work of Gaver [10] differ within the following modes of behavior of the device when the system becomes empty:

- *set to zero* — upon the completion of service of the last request in the system the device switches immediately to the *neutral state.* If the first request which enters the empty system is a request of the priority $i$, then the device proceeds with the switching of the duration $S_i$. Obviously, this regime is well defined in the systems with the neutral state switching disciplines. However, one can define the *set to zero* regime for the systems with the "neutral state free" switching processes. For this, consider a neutral state as a special state of device's relaxation while being idle. Additional random times $\{C_i\}_{i=1}^{r}$ of post-warming switching will be required to be specified then.

- *look ahead* — the device switches itself to the 1-requests' line at the moment the system becomes empty.

- *wait and see* — the device remains switched to the queueing line of the last served request.

- *wait for the most probable* — the device switches to the flow of the most likely to appear customers. To clarify, this can be understood as follows. Let $a_i(t) = A_i'(t)$ be the density of the $i$-requests' inter-arrival times, $i = 1, \ldots, r$. Then, by the flow of the most likely to appear customers understand the $p$-requests' flow, where $p = \arg\max\limits_{i} a_i(t_0+)$, where

  $$t_0 = \min_{i=1,\ldots,r} \sup_{a_i(t)=0} t.$$

  If $p$ is not determined uniquely, then some additional considerations may be taken into account — for instance, $p$ may be taken as follows:

$$p = \min\{\arg\max_i a_i(t_0+)\}. \tag{2.2}$$

A large class of priority queueing systems is described. Essentially, it comprises the systems defined by the following information and identifiers:

- arrival flows — distributions of inter-arrival times (for each flow);
- service times — distributions of service times (for each flow);
- switching times — specification of the switching type (neutral state or not) and distributions of switching times;
- warming time — distribution of waiting times;
- order of service within a line (FIFO, LIFO);
- service discipline;
- switching discipline;
- behavior of the device in the idle state.

Adopt the generalization of the standard Kendall notation $A_r|B_r|1$ for such systems with writing of an additional information on the identifiers listed above, which specify the system.

**Example 1.** *The queueing systems with the Poissonian incoming flows are of great importance in the theory and practice. In this case the inter-arrival times are exponentially distributed, i.e.* $A_i(t) = 1 - e^{-\lambda_i t}$, $i = 1, \ldots, r$, *where* $\lambda_1, \lambda_2, \ldots, \lambda_r$ *are some non-negative real numbers with the physical meaning of the flow arrival rates. A typical system with the Poissonian incoming flows may be specified then as follows:* ***FIFO*** $M_r|G_r|1$ ***"neutral state"-"request postponable service discipline"-"preemptive switching" priority queueing system with the "wait for the most probable" device's regime****.*

*Here, the most probable requests are the $p$-requests, where $p$ is determined from (2.2), i.e.* $p = \min\{\arg \max\limits_{i=1,\ldots,r} \lambda_i\}$, *and it has a clear physical meaning — $p$ is the highest priority level of the requests among those which have the greatest arrival rate.*

## 2.3 Characteristics of system performance

One can specify many stochastic processes taking place in the described queueing systems. Some of the characteristics of these stochastic processes are of special interest and may well serve as system performance characteristics.

Begin with the notions of *busy period* and *idle period* (or *vacation period*). Call by the *busy period* the period of time during which the device is occupied either with servicing of the requests or with the switching. The notion of busy period is intuitively absolutely clear. We shall call the periods of time which alternate busy periods by *idle periods*. It is clear that a busy period follows some idle period and vice versa.

Let $\mathit{\Pi} = \{\Pi_1, \Pi_2, \ldots\}$ be consecutive busy periods of the system. Note that in $M_r|G_r|1$ models $\mathit{\Pi}$ is a sequence of independent and identically distributed (iid) random variables with some cdf $\Pi(t)$, unless it is the model with the *"wait and see"* mode of behavior of the device in the idle state. Therefore, denote the random variable which has a cdf $\Pi(t)$ by $\Pi$ and refer to it as a busy period. Note that its distribution $\Pi(t)$ does not depend on the order of requests' service (FIFO, LIFO). We conjecture that all this is also true for the scheme *"wait and see"*.

Describe by vector $\mathbf{m}(t) = \{m_1(t), m_2(t), \ldots, m_r(t)\} \in {\mathbb{N}^*}^r$ the state of the system at time $t$, where $m_i(t)$ is the number of $i$-requests in the system at time $t$. Here $\mathbb{N}^* \overset{\partial ef}{=} \mathbb{N} \cup \{0\}$. Denote by $m(t)$ the number of all requests in the system at time $t$. Thus,

$$m(t) = \sum_{i=1}^{r} m_i(t).$$

Introduce also the following notations for cdf's of $\mathbf{m}$ and $m$:

$$P_{\mathbf{m}}(t) \overset{\partial ef}{=} \mathbb{P} \text{ (there are } m_i \ i\text{-requests in the system at time } t),$$

where $\mathbf{m} = (m_1, \dots, m_r)$; and

$$P_m(t) \overset{\partial ef}{=} \mathbb{P} \text{ (there are } m \text{ requests in the system at time } t).$$

The following rather abstract notions of *virtual waiting time* and *virtual sojourn time* are very important in the theory of queueing systems and its applications. Consider $i$-requests and ask the question: what time should wait an $i$-request to get start served if it arrived in the system at time $t$? This time period can obviously be considered as a random variable. Denote it by $W_t^{(i)}$ and call *the virtual waiting time of i-requests*. Denote the cdf of $W_t^{(i)}$ by $W^{(i)}(t, \tau)$, i.e.

$$W^{(i)}(t, \tau) = \mathbb{P}(W_t^i \leq \tau).$$

Analogously, the time that an $i$-request would spend in the system if it entered the system at time $t$ is a random variable denoted by $V_i$ with cdf $V_i(t, \tau)$, i.e.

$$V^{(i)}(t, \tau) = \mathbb{P}(V_t^i \leq \tau).$$

Note that the virtual waiting and sojourn times essentially depend on the requests' service order (FIFO, LIFO).

Introduce also the notion of a *loss probability.* Let

$$P_{loss}^{(i)} \overset{\partial ef}{=} \mathbb{P} \text{ (an } i\text{-request will be lost)}$$

for the scheme "with losses".

**Stationarity.** The notion of stationarity is very important in the study of time-evolving stochastic systems. Usually the system is considered to be stationary if its behavior becomes stable and, in some sense, settled down. Many system characteristics have stationary analogues then and often these are very convenient for describing the settled system behavior after some, may be quite long, period of time.

More formal means for the study of the stationarity in the family of the systems considered here are provided by the theory of regeneration processes and embedded Markov processes. General methodology here is to discern some underlying, embedded process, say a (continuous) Markov chain, in the main stochastic process, described, for example, by vector $\mathbf{m}(t)$, and then to impose some restrictions on the system parameters to obtain the condition of stationarity. Often such condition is just sufficient and usually it can be formulated in terms of some quantity $\rho$ which is then to be called a *system workload*, or *traffic coefficient.* We shall call it a *node traffic coefficient.* Standard form of expressing the stationarity of the system is an inequality of the following form:

$$\rho < 1. \tag{2.3}$$

In the following section we will again point out the importance of this characteristic for the network traffic analysis.

# 3 Performance characteristics of priority systems with switchover times

## 3.1 Busy period

The definition of the busy period in priority queueing models involving switching is given in §2.3.

### 3.1.1 Motivation

The notion of the busy period is a very important notion. It is really important to know how busy periods are distributed in order to evaluate the system performance and the load of the device.

It may also be useful and necessary to evaluate the busy periods when we want to find some other characteristics of a queueing system, such as queue length or server's state, for instance.

Let $P_{\mathbf{m}}(t)$ be the probability of the event "there are $\mathbf{m} = (m_1, \ldots, m_r)$ requests in the system at time $t$." Define $P(z,t) \overset{\partial ef}{=}$

$\sum\limits_{\mathbf{m}\geq 0} P_{\mathbf{m}}(t)z^{\mathbf{m}}$, where $z^{\mathbf{m}} = z_1^{m_1} \dots z_r^{m_r}$, $z_i \in [0,1]$. Then, the Laplace-Stieltjes transform

$$p(z,s) = \int\limits_0^\infty e^{-st} dP(z,t)$$

of this generating function may be determined with a help of the following

**Theorem 2.** *([19]) The Laplace-Stieltjes transform $p(z,s)$ of $P(z,t)$ in $M_r|G_r|1$ can be found as follows:*

$$p(z,s) = \frac{1+\sigma\pi(z,s)}{s+\sigma-\sigma\pi(s)},$$

*where $\sigma\pi(z,s) = \sigma_r\pi_r(z,s)$ may be determined from the following recurrent equation*

$$\begin{aligned} \sigma_k\pi_k(z,s) &= \sigma_{k-1}\pi_{k-1}(z,s) + \gamma_{k-1}(s,z)\nu_k(z,s) & (3.1)\\ &+ \frac{h_k(z,s)}{z_k - h_k(s+[\sigma-\lambda z]_k)}[\gamma_{k-1}(s,z)\nu_k(s+[\sigma-\lambda z]_k) \\ &+ \sigma_{k-1}\pi_{k-1}(s+\lambda_k) - \sigma_k\pi_k(s)], \textit{where} \\ \gamma_{k-1}(s,z) &= \sigma_{k-1}[\pi_{k-1}(s+[\sigma-\lambda z]_k) - \pi_{k-1}(s+\lambda_k)] + \lambda_k z_k, & (3.2) \end{aligned}$$

*and $[\sigma-\lambda z]_k := \sum\limits_{i\leq k}\lambda_k(1-z_k)$; $h_k$ and $\nu_k$ should be specified for a certain discipline (e.g., see Theorem 3). Here $\sigma_k := \sum\limits_{i=1}^{k}\lambda_i$.*

In this theorem $h_k$ is a LST of a *$k$-service period $H_k$* — the time which starts when a $k$-request enters the server and finishes when the server is ready to serve the next $k$-request queueing in a respective waiting line; $\nu_k$ is a LST of *$k$-switching period $N_k$* — the period of time starting from the switching to $k$-requests' waiting line and ending when the server is ready to serve $k$-requests.

To summarize: to know how the busy periods are distributed is to be able to evaluate many other system performance characteristics.

### 3.1.2 Traffic coefficient and the generalized Kendall equation

We give here more details on node traffic coefficient and its connection with busy period in systems $M_r|G_r|1$. We assume, that $C_{ij} \equiv C_j$, independently on $i$.

The following result is due to Mishkoy [19].

**Theorem 3.** *For the system $M_r|G_r|1$ under preemptive discipline and scheme "with losses" the following equations hold*

$$\begin{aligned}
\pi_k(s) &= \frac{\sigma_{k-1}}{\sigma_k}[\pi_{k-1}(s+\lambda_k) + \delta_{k-1}(s)\nu_k(s+\lambda_k[1-\bar{\pi}_k(s)])] \qquad (3.3)\\
&\quad + \frac{\lambda_k}{\sigma_k}\pi_{kk}(s),\\
\pi_{kk}(s) &= \nu_k(s+\lambda_k[1-\bar{\pi}_k(s)])\bar{\pi}_k(s), \qquad (3.4)\\
\bar{\pi}_k(s) &= h_k(s+\lambda_k[1-\bar{\pi}_k(s)]), \qquad (3.5)\\
\nu_k(s) &= \frac{c_k(s+\sigma_{k-1})}{1-\frac{\sigma_{k-1}}{s+\sigma_{k-1}}[1-c_k(s+\sigma_{k-1})]\pi_{k-1}(s)}, \qquad (3.6)\\
h_k(s) &= \beta_k(s+\sigma_{k-1}) \qquad (3.7)\\
&\quad + \frac{\sigma_{k-1}}{s+\sigma_{k-1}}[1-\beta_k(s+\sigma_{k-1})]\pi_{k-1}(s)\nu_k(s),\ k=1,\dots,r,\\
\pi_0(s) &= 0. \qquad (3.8)
\end{aligned}$$

*The condition of stationarity is*

$$\rho_r = \sum_{k=1}^{r} \lambda_k b_k < 1, \tag{3.9}$$

*where $b_1 = \frac{\beta_{11}+c_{11}}{1+\lambda_1 c_{11}}$, and*

$$\begin{aligned}
b_i &= \Phi_1 \dots \Phi_{i-1}\frac{1}{\sigma_{i-1}c_i(\sigma_{i-1})}[\frac{1}{\beta_i(\sigma_{i-1})}-1], \qquad (3.10)\\
\Phi_1 &= 1, \qquad (3.11)\\
\Phi_i &= 1+\frac{\sigma_i-\sigma_i\pi_{i-1}(\lambda_i)}{\sigma_{i-1}}[\frac{1}{c_i(\sigma_{i-1})}-1]. \qquad (3.12)
\end{aligned}$$

Here $\rho_r$ is nothing but the node traffic coefficient $\rho$.

The condition (3.9) means that $\Pi(t)$ is a proper cdf, i.e. busy periods are almost surely of finite length. This is an important condition for the QoS traffic analysis, as it is useful for nodes' overloading control.

Note, that the moments of both cycles and busy period may be easily obtained by differentiating their Laplace-Stieltjes transforms at zero. Note also, that the equations (3.3) and (3.5) can be viewed as generalizations of the classical Kendall equation for the LST of busy period in $M_r|G_r|1$. It is really necessary to solve the system (3.3)-(3.8) for the values of $\pi_i(\lambda_i)$ as these are required for the evaluation of the traffic coefficient. Moreover, if one has to get more complete information about distribution of busy periods, then one should be able to solve the mentioned system at any non-negative point $s$ and to invert (numerically) the Laplace-Stieltjes transform.

### 3.1.3 Examples and numerical methods

Consider the systems of described above type with "degenerated" (i.e. null, zero) orientation time. The following typical result is known from [11] (we give it in a short form, more suitable for our needs now).

**Theorem 4.** *In $M_r\,|G_r|\,1$(scheme "with losses") the following system of functional equations*

$$\begin{cases} h_k(s) = \beta_k(s+\sigma_{k-1}) + \frac{\sigma_{k-1}}{s+\sigma_{k-1}}[1-\beta_k(s+\sigma_{k-1})]\pi_{k-1}(s), \\ \pi_{kk}(s) = h_k(s+\lambda_k-\lambda_k\pi_{kk}(s)), \\ \pi_{ki}(s) = \pi_{k-1,i}(s+\lambda_k-\lambda_k\pi_{kk}(s)),\ i=1,\ldots,k-1 \\ \sigma_k\pi_k(s) = \sum\limits_{j=1}^{k}\lambda_j\pi_{kj}(s) \end{cases}$$

*determines unique functions $h_k(s), \pi_{ki}(s), \pi_k(s)$ $(i,k=1,\ldots,r)$, which are analytical in the half-plane $\Re s > 0$, where $|h_k(s)|<1$, $|\pi_{ki}(s)|<1$, $|\pi_k(s)|<1$. Moreover, if $\rho := \lambda_1\beta_{11} + \sum\limits_{j=1}^{k-1}\frac{\lambda_{j+1}}{\sigma_j}[\frac{1}{\beta_{j+1}(\sigma_j)}-1] \le 1$ then $h_{k+1}(0)=\pi_{ki}(0)=\pi_k(0)=1$, and no one equality holds otherwise. Here $\beta_{11} := \int\limits_0^\infty t dB_1(t)$ and $\pi_0(s)\equiv 0$.*

Functions $\pi_k(s), \pi_{ki}(s)$ included in the expressions above are LST's of cdf's of some supplementary time intervals. Specifically, $\pi_r(s)$ is nothing but a $\mathcal{LS}[\Pi(t)]$, i.e. the Laplace-Stieltjes transform of the cdf $\Pi(t)$. Note, that this theorem can be easily derived from the more general result provided by Theorem 3.

These examples of relatively simple priority queueing models show that it is necessary to develop numerical methods of their analytical description.

For some of the described schemes such numerical algorithms have already been developed and applied in [3]. The work is in progress to provide the numerical algorithms for all the schemes from the classification given in Section 2.

## 4 The problem of the input flow type and imitation modeling of priority queueing systems

In the previous section we have proposed a wide range of priority queueing models to describe processes taking place in communication and information networks. As it has been already pointed out, in order to design better communication network and to provide higher level of quality of service, it is really important to be able to evaluate network performance parameters. We concentrated ourselves on node traffic characteristics and we described complex priority queueing models with switching.

One of the crucial cornerstones of queueing theory traditionally was the assumption that queues and incoming requests can be modeled as continuous-time Markov chains. Alternatively, one can distinguish an embedded Markov chain and still perform the analysis of a system. This allowed to make extensive use of the exponential distributions and memoryless properties in the study of such systems.

However, it has been recently discovered that, in practice, flows of incoming requests in queueing systems may exhibit some additional statistical properties that cannot be ignored in the theory. For instance, it has been found that traffic in communication networks can exhibit such phenomena like *self-similarity*, *long-range dependence* and *burstiness.* In such cases development of traffic models is more sophisticated and analytical methods became less powerful. Zwart [28] notes that a careful statistical analysis in [18] showed that Ethernet LAN traffic at Bellcore exhibits these properties. It also behaves extremely bursty on a wide range of time scales. Among other sources that confirm that discussed phenomena take place in today traffic we mention [5], [22], [23], [26], [27].

Yet, one of the alternative ways of study of such systems is the method of *imitation modeling.* One can choose different tools and methodologies to use this method in the context of telecommunication technologies, and, particularly, in the context of wireless systems: [8], [13] (using OPNET), [14] (using stochastic Petri nets), etc.

We only concentrate ourselves here on the priority systems described in Section 2. As it has already been pointed out, the assumption about non-Poissonian nature of arrival flows makes analytical methods to be less efficient in providing information on the system performance characteristics.

Let us assume that instead of $M_r|G_r|1$ priority queueing system with switchover times a $G_r|G_r|1$ system is studied and it is the system of interest in providing a corresponding node QoS. The simulation package of classes $PQSST$ by Botezatu and Bejan [6] can be efficiently used for these purposes. It was designed to provide simulation tools of the performance analysis of systems $G_r|G_r|1$, supporting all the disciplines described in the previous section. In this package the interarrival and service times for each flow can be chosen to be of one of the following probabilistic laws: *Arcsine*, *Beta*, *Chi Square*, *Constant*, *Erlang*, *F-Ratio*, *Gamma*, *Logarithmic*, *Lognormal*, *Parabolic*, *Pareto*, *Power*, *Rayleigh*, *Triangular*, *Uniform*, *Weibull.* The package is implemented as Java applet which is accessible online at the following address: `http://vantrix.net/queues/applet.htm`

Original data and system representation algorithms were used in the package $PQSST$ which are based mostly on an object-oriented approach in modeling of such systems (e.g., see [12]).

The package $PQSST$ allows to obtain full chronology of the system under study. Additionally, it provides summary on busy periods statistics, idle periods statistics, mean waiting times of requests, loss probabilities (see § 2.3).

It is believed that the package *PQSST* will be of real interest for those interested in performance analysis of priority queueing systems with switchover times, and particularly, in the context of QoS provision in communication traffic systems.

## 5 Example of network modeling with priority queueing systems

We continue with an example of usage of the described systems. This example is based on a Cisco Priority Queueing technology which is described in [15].

Priority queueing is useful for making sure that mission-critical traffic traversing various WAN links gets priority treatment. For example, Cisco uses priority queueing to ensure that important Oracle-based sales reporting data gets to its destination ahead of other, less-critical traffic. Priority queueing uses static configuration mechanism and does not automatically adapt to changing network requirements. In this example prioritization represents the process of placing data into four levels of queues: high, medium, normal and low. This is shown schematically in Figure 1.

It is easy to see that this process of prioritization can be modeled as $G_4|G_4|1$ priority queueing system with postponable priority service discipline and correspondingly chosen densities $a_i(t)$ and $b_i(t)$ of inter-arrival and service times, respectively (arrival process can be complex and exhibit such properties as self-similarity, long-range dependence, or burstiness, as discussed above). The discipline of switching can also be appropriately chosen.

However, one might prefer to consider a service discipline other than non-preemptive one (as *postponable service discipline* is, accordingly to the classification given in the previous section) in order to minimize mean waiting times of the packets, for instance. The package *PQSST*

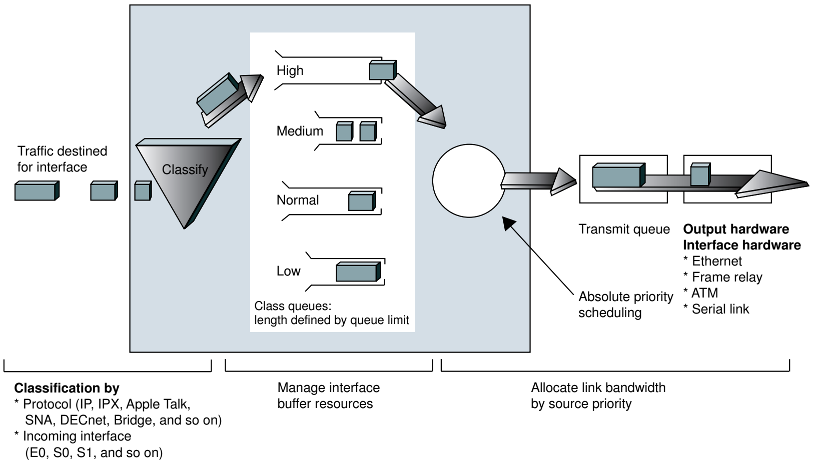

Figure 1. Priority Queueing Places Data into Four Levels of Queues: High, Medium, Normal, and Low (reproduced from CISCO documentation [15]).

may be useful for these purposes, unless the incoming flows are of Poissonian type (analytical methods can be applied then).

## 6 Concluding remarks

We described in this paper a large class of priority queueing systems involving switching as a class of adequate models of the phenomena which take place in a network. The performance analysis of such systems may essentially influence the ways of estimating and providing a respective level of QoS in networks via estimating nodes QoS's.

One of the most important characteristics of the priority queueing systems is the node traffic coefficient $\rho$. This quantity plays the crucial role in estimation of the node QoS. The role of the stationarity condition of the form (2.3) (or, for instance, of the condition (3.9) for the system $M_r|G_r|1$ under preemptive discipline and scheme "with losses" with zero switchover times) has been discussed. This is an important condition on a way of providing network QoS. Note, that if at least one of the node traffic coefficients of a network is equal or greater than zero, than the corresponding nodes becomes overloaded (busy periods are of infinite length with probability one).

It has been pointed out that special numerical algorithms and schemes should be elaborated in order to estimate node traffic coefficients in the systems of general type. As it may be easily seen from the results of Theorem 3 and Theorem 4 the problem of estimation of node traffic coefficients is closely related to the problem of the busy periods' estimation.

It will be shown in further research that the *blocking probability* (which is one of the main QoS characteristics) can also be expressed with the help of the system of functional equations of the form (3.3) - (3.5). It has been mentioned that the system (3.3) - (3.5) represents a generalization of the well-known Kendall equation. Similarly, the result of Theorem 2 can be viewed as a generalization of the classical Pollaczek-Khintchine formula.

Yet, an alternative method of study of the considered system is the method of imitational modeling, which was applied to the described systems: the package $PQSST$ has been designed to imitate such systems and estimate empirically their most important performance characteristics.

In this paper we suggested to relate network QoS characteristics to node QoS characteristics on a qualitative level. It is a matter of future work to propose such a connection on a quantitative level.

## References

[1] Aad, I., C. Castelluccia. 2003. Priorities in WLANs. *Computer Networks* **41** 505–526.

[2] Anderson, R. H. et al. 2003. *Securing the U.S. Defense Information Structure*. Rand Monograph Report.

[3] Bejan, A. 2004. On algorithms of busy time period evaluation in priority queues with orientation time. *Communications of the Second Conference of the Mathematical Society of the Republic of Moldova* 32–36.

[4] Bejan, A., Gh. Mishkoy. 2004. Switchover time regularities in priority queueing systems. *Communications of The Second Conference of the Mathematical Society of the Republic of Moldova* 36–39.

[5] Beran, J., R. Sherman, M. S. Taqqu, W. Willinger. 1995. Long-range dependence in variable-bit-rate video traffic. *IEEE Transactions in Communications* **43** 1566–1579.

[6] Botezatu, M., A. Bejan. 2006. Priority Queueing Systems with Switchover Times: Modeling and Analysis in Java. *Communications of The XIV Conference on Applied and Industrial Mathematics (Satellite Conference of ICM 2006)* 49–52

[7] Cisco Systems. *Cisco IOS 12.0 Quality of Service.* Indianapolis: Cisco Press, 1999.

[8] Chow, J. 1999. Development of channel models for simulation of wireless systems in OPNET. *Transactions of the Society for Computer Simulation International* **16**(3) 86–92.

[9] Gaver, D. P. 1962. A waiting line with interrupted service, including priorities. *J. Roy. Stat. Soc. B* **24** 73–90.

[10] Gaver D. P. 1963. Competitive queueing: idleness probabilities under priority disciplines. *J. Roy. Stat. Soc. B* **25**(2) 489–499.

[11] Gnedenko, B.V. et al (1973) *Priority Queueing Systems.* Moscow State University Press, in Russian.

[12] Grama, I., G. Mishkoy 1993. The Object Oriented Programming for Queueing Systems. *Computer Science Journal of Moldova* **1**(1) 85–104.

[13] Green, D. B., M. S. Obaidat. 2003. Modeling and simulation of IEEE 802.11 WLAN mobile ad hoc networks using topology broadcast reverse-path forwarding (TBRPF). *Computer Communications* **26** 1741–1746.

[14] Heindl, A., R. German. 2001. Performance modeling of IEEE 802.11 wireless LANs with stochastic Petri nets. *Performance Evaluation* **44** 139–164.

[15] Cisco Systems. *Internetworking Technology Handbook.* Documentation.
http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/qos.htm

[16] Kakaraparthi, R., et al. 2000. Efficient message scheduling in ad hoc networks. *Proc. IEEE Wireless Communications Networking Conf.* 1226–1231.

[17] Klimov, G. P., G. K. Mishkoy. 1979. *Priority Queueing Systems with Orientation.* Moscow "Nauka" (in Russian).

[18] Leland, W. E., M. S. Taqqu, W. Willinger, D. V. Wilson. 1994. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking* **2** 1–15.

[19] Mishkoy, G. K. 1990. Priority queueing involving orientation and the problems of their software implementation. *Computers Math. Applic.* **19**(1) 109-113.

[20] Miorandi, D., A. A. Kherani, E. Altman. 2005. A queueing model for HTTP traffic over IEEE 802.11 WLANs. Preprint.

[21] Nortel Networks. 2003. *Introduction to Quality of Service.* Technical document. White Paper.

[22] Park, K., W. Willinger, eds. 2000 *Self-Similar Network Traffic and Performance Evaluation.* Wiley, New-York.

[23] Paxson, V., S. Floyd. 1995. Wide area traffic: the failure of Poisson modeling. *IEEE/ACM Transactions on Networking* **3** 226–244.

[24] Peuhkuri, M. 1999. *IP Quality of Service.*
http://www.netlab.tkk.fi/u/puhuri/htyo/Tik-110.551/iwork/iwork.html

[25] Stine, J. A., Gustavo de Veciana. 2004. A paradigm for Quality-of-Service in Wireless Ad Hoc Networks Using Synchronous Signaling and Node States. *IEEE Journal on Selected Areas in Communications* **22**(7) 1301–1321.

[26] Willinger, W., M. S. Taqqu, W. E. Lelenad, D. V. Wilson. 1995. Self-similarity in high-speed packet traffic: analysis and modelling of Ethernet traffic measurements. *Statistical Science* **10** 67–85.

[27] Willinger, W., M. S. Taqqu, R. Sherman, D. V. Wilson. 1997. Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level. *IEEE/ACM Transactions on Networking* **5** 71–86.

[28] Zwart, A. P. 2001. Queueing systems with heavy tails. *PhD thesis.* Eindhoven: Eindhoven University of Technology.

Gh. Mishkoy, S. Giordano, A. Bejan, O. Benderschi, Received May 8, 2007

Gh. Mishkoy
Academy of Sciences of Moldova,
Free International University of Moldova
E–mail: *gmiscoi@ulim.md*

S. Giordano
University of Applied Sciences of Southern Switzerland

A. Bejan
State University of Moldova, Heriot-Watt University (Edinburgh)
E–mail: *a.i.bejan@ma.hw.ac.uk*

O. Benderschi
State University of Moldova