

Switched Ethernet For Real-Time Industrial Communication: Modelling And Message Buffering Delay Evaluation

Yeqiong SONG and Anis KOUBAA
LORIA – INRIA Lorraine – UHP Nancy 1
ENSEM – 2, av. de la Forêt de Haye
54516 Vandoeuvre – France
Email : song@loria.fr; akoubaa@loria.fr

François SIMONOT
IECN – ESSTIN – UHP Nancy 1
ESSTIN – Parc R. Bentz
54500 Vandoeuvre – France
Email : simonot@esstin.uhp-nancy.fr

Abstract

Switched Ethernet is now considered as an attractive enabling technology for supporting factory communication needs. This paper deals with the modelling of an Ethernet switch and the performance evaluation of its real-time features. For multiple priority periodic and single priority aperiodic input traffic, methods for calculating respectively the worst case buffering delay and buffering delay probability distribution are given, allowing thus to estimate both hard real-time and soft real-time guarantees. Moreover for aperiodic traffic, Binomial input is compared to the Poisson one and we numerically showed that the Poisson case could be used as an approximation model to upper bounds the buffering delay of the actual switch. This allowed us to further consider the priorities for buffering delay evaluation using classic results of M/G/1 queue.

1. Introduction

During the last two decades many fieldbus protocols have been developed and successfully implemented for mainly meeting the real-time communication needs in the low layer of the industrial process control hierarchy. However the implementation of DCS (Distributed Control System) concept led to the expansion of intra-plant data quantity, the creation of inter-plant communication needs and the willing of taking advantages of the standard Internet protocols require higher bandwidth networks with guaranteed real-time QoS (Quality of Service). Whilst most of popular fieldbusses like Profibus, WorldFIP [1] and CAN [2] only offer low bit rate (1 to 5 Mbps). To be interconnected to Internet world most fieldbusses proposed to encapsulating IP packet but their short layer 2 frame lengths can not efficiently carry the large IP data flow. On the other hand Ethernet offers larger bandwidth but is not designed for supporting real-time communications. As CSMA/CD used in classic shared Ethernet does not provide deterministic medium access

delay because of collisions, An early approach has been to modify Ethernet to handle real-time traffic. Solutions range from implementing Master-slaves communication model over Ethernet to Token-passing or TDMA to schedule the access of the nodes, or even by modifying CSMA/CD to CSMA/DCR[3].

Switching technology makes easier all the above mentioned tentatives since an Ethernet switch can easily reduce collisions by segmenting the whole network into sub-segments or even completely eliminate the collisions by only connecting one node per full-duplex switch port. In this sense CSMA/CD can be no longer used. Using widely spread high speed Ethernet as a common enterprise-wide network infrastructure to carry both factory automation and office automation communications becomes feasible.

Ethernet is incontestably the most cost-effective solution (very low price, component maturity and hence reliability, stability since widely used by IT industry). In addition, no special staff training is needed since almost all the network engineers know well Ethernet as well as Internet related high layer protocols. Today's switched Ethernet offers following main interesting features for factory communication:

- Large bandwidths (10Mbps, 100Mbps, 1Gbps, 10Gbps)
- Deterministic network access delay thanks to the switching principle and full-duplex links
- Priority handling (IEEE802.1p) which provides a basic mechanism for supporting real-time communications
- Broadcast traffic isolation, scalability and enhanced security by configuring the network in terms of VLAN (Virtual LAN).
- Reliability improved by deploying STP (Spanning tree protocol) on redundant paths
- Deployment facility with WLAN (Wireless LAN, i.e.: IEEE802.11 LAN)
- *De facto* standard supporting many widely spread upper stacks (IP and socket-based UDP and TCP) for file transfer (FTP), remote login or virtual terminal (telnet), network management (SNMP),

Web-based access (HTTP), email (SMTP), which makes very easy its use and allows integrating many COTS (Commercial Off-The Shelf) API and middlewares like OPC (OLE for Process Control <http://www.opcfoundation.org>), Microsoft DCOM, Corba, Java/RMI, etc.

This is why switched Ethernet is more and more considered as an attractive enabling technology for supporting time-constrained communication.

But since Ethernet is not initially designed for factory communication three main problems should be resolved: i) hardware and software components should be enhanced for handling real-time requirement and for being used in industrial hostile environment (vibration, temperature variation, EMI, explosive atmosphere where the direct node powering is forbidden, ...), ii) new real-time process control protocol stack beside TCP/IP should be added, iii) system-wide temporal performance should be predictable (either for a priori static validation knowing the applications it supports or on-line dynamic request admission control).

For the two first points many industrial efforts have been made. Many PLC and field device suppliers propose now specially protected cables and enhanced RJ45 connectors. The remote powering of the nodes is also under active study. In our opinion, this should not be a difficult problem if the remote powering principle in IEEE802.11 wireless LAN for the access points can be adopted. A similar architecture with added control application oriented profile is defined in IEA (Industrial Ethernet Alliance, www.industrialethernet.com), in IDA (Interface for Distributed Automation, www.ida-group.org) and in EtherNet/IP (Ethernet/Industrial Protocol, www.odva.org). This is a real-time middleware with specific API over Ethernet TCP/IP. IEA proposes to use fieldbus high layer protocol like modbus, PROFINET of Profibus over Ethernet. IDA's profile is added over UDP while EtherNet/IP's profile over multicast IP (IGMP). Fieldbus Foundation (www.fieldbus.org) proposes an architecture in two levels with H1 as fieldbus and HSE (High Speed Ethernet) as backbone. IAONA (Industrial Automation Open Networking Alliance, www.iaona-eu.com) defined 4 real-time classes according to the timeliness requirement of applications. Class 1 covers standard Ethernet components, Class 2 covers standard Ethernet components but optimized for real-time demands (e.g. optimised "zero-copy" implementation of IP/UDP stack, i.e. a packet stays in the same buffer rather than copied to another memory zone every time it travels through TCP/IP layers). Class 3 are components with new added functions implemented in software (e.g. real-time middleware). Class 4 is the hardware version of class 3.

As for the 3rd point, there does not exist dedicated work on the real-time guarantee from distributed system point of view. This may be a reason still preventing the larger deployment of industrial Ethernet. Many say that switched Ethernet is determinist since the collision can

be eliminated and moreover with a so large bandwidth if the network load can be kept very low (under 1%) all traffic should travel through fluently. While others say it is not determinist as the messages could suffer a random buffering delay in a switch. So in-depth analysis is necessary for evaluating the message buffering delay in a switch and for providing guidelines for the network-wide configuration. There exists some measurement-based performance tests according to IETF benchmarks (RFC2285) at switch suppliers' web sites or for example in (www.NetworkComputing.com/815/815f1.html). The main performance parameters are RFC1944 throughput, frame/packet loss probability, many to one congestion handling capability and RFC2285 HOL (Head Of Line) blocking, X-stream performance, address-handling tests and illegal frame filtering tests. Jasperneite and Neumann[4] evaluated by simulation the message response time of a totally switched Ethernet structure in bus and star topology with periodic and Poisson input traffic patterns and with different priority levels. In our earlier work, Song [5] has given a detailed analysis of the real-time traffic handling features of the switched Ethernet and evaluated the average buffering delay by assuming a binomial input flow without priority. Koubâa and Song [6] have extended the previous work by introducing the priorities and dealing with a case study. But for simplifying the analysis a Poisson arrival pattern is assumed. In the above work, two problems remain to resolve: one is the buffering delay distribution in Binomial input case and the another one is the justification of using a Poisson arrival to approximate the Binomial one.

In this paper we give firstly an in-depth analysis of the components of the message response time and then the methods to evaluate the buffering delay in a switch. The rest of the paper is organized as follows. Section 2 describes the modelling of a switch by a M/G/1-like queue. Section 3 evaluates the buffering delay in three different input traffic patterns: the worst case buffering delay (upper bound) of periodic/D/1 queue which can be used for providing HRT (Hard Real-Time) guarantee; the probability distribution of the buffering delay of Poisson/D/1 (or M/D/1) and Binomial/D/1 queues for providing probabilistic SRT (Soft Real-Time) guarantee. Section 4 gives a numerical comparison between M/D/1 and Binomial/D/1. Section 5 concludes this paper and points out some future research directions.

2. Modelling

The number of LAN switches continues to proliferate. Since there is no standard for Ethernet switch implementation, there exists now many different kinds of switch internal architectures called switch fabrics. The three main architectures of today are matrix, bus and shared-memory [7].

Matrix based switch fabric (e.g.: crossbar) was issued from the telecommunication switches. The interest of

this architecture is its great number of ports that neither bus nor shared-memory can currently achieve. But this architecture is more problematic when broadcast or multicast and unicast occur simultaneously. In fact, no unicast can be transmitted when a broadcast or a multicast is taking place.

Bus architecture has a very high-speed core bus (collapsed backbone) shared by modules (i.e., input/output ports). The bus access control is often based on TDMA. An advantage of the bus comparing to the matrix architecture is that the bus architecture naturally supports broadcast traffic. One of the problems of this architecture is the output buffer overflow when many inputs should be forwarded to a same output port.

In practice, some commercially available switches use a mixed bus and matrix architectures such as Cisco Catalyst 5500 and 6000 series (www.cisco.com) and Allayer ROX series (www.allayer.com).

The third widely used and most popular (due to the lowest cost) architecture is based on a ultra rapid simultaneous multiple access memory shared by all ports. A packet entering to the switch is firstly stored in memory. The packet forwarding is performed by ASIC engine which looks up the destination MAC address in the forwarding table, finds it, then sends it to the appropriate output port. Output buffering is used instead of input buffering to avoid the HOL (head-of line) blocking. Output buffer overflow can be minimized by using shared-memory queuing since the buffer size is dynamically adjusted. In fact, all output buffers share a same global memory reducing thus the buffer overflow comparing to the per-port queuing. Typical commercially available examples are Cisco Catalyst 4000 and 8500 series (www.cisco.com).

All recent Ethernet switches are announced operating with wire-speed and non-blocking. Wire-speed means that all ports of a switch can simultaneously transmit or receive at their full bit rates. This requires that the switch fabric can operate at a bit rate equalling to the aggregate speeds of all the ports. For example, a 24 full-duplex ports fast Ethernet switch needs a fabric forwarding at 4.8 Gbps (24x2x100Mbps). A switch is non-blocking when it can forward a message to the destination port as long as that port is free, while a blocking one may be not able to forward a message although the destination port is free because of internal conflict in the switch fabric (one example is the HOL blocking in input buffering switches). Switches with output buffering are non-blocking.

Buffering and thus buffering delay exists in a switch whatever the switch is with full wire-speed or not. In fact message buffering occurs whenever the output port cannot forward all input messages at time. This corresponds to the burst traffic arrival. The analysis of the buffering delay depends on knowledge on the input traffic pattern. The only true no buffering switch is that with each output port bit rate always higher than or equal

to the sum of all possible input ports' bit rate! But considering that communication is essentially bi-directional, such a switch should not exist.

One can distinguish a fully switched Ethernet from those including both switches and hubs. In a fully switched Ethernet there is only one equipment (station or switch) per switch port. In case that wire-speed full-duplex switches are used, the end-to-end delay can be minimized by decreasing at maximum the message buffering. A frame travelling through the switches in its path without experienced any buffering has the minimum delay. The total delay introduced by a switch is composed of :

- The *switching latency* (traffic classification according to IEEE802.1p mapping table, destination port look-up and switch fabric set-up time),
- The *frame forwarding time* which depends on the forwarding mode (cut-through or store and forward) and eventually on the frame length if the "store & forward" mode is running,
- The *buffering delay* when the frame is queued.

The switching latency is a fixed value, which depends on the switch performance and often provided by the switch vendor (e.g. 11 μ s between 100Mbps ports and 70 μ s between 10Mbps ports for Cisco Catalyst 1900, 2820). The frame forwarding time can be obtained knowing in which forwarding mode the switch is running. The technique for analysing the buffering delay depends on the knowledge on the input traffic pattern. For periodic input traffic (or (σ , ρ)-bounded or sporadic majoring by a periodic one by taking the minimum inter arrival time as the data emission period), classic scheduling analysis or (σ , ρ)-related analysis [8] can give the worst-case buffering delay, providing thus the hard real-time guarantee. But for aperiodic input, since only few on traffic characteristics is known, a stochastic analysis is needed.

Generally a full-duplex Ethernet switch of N ports of the same bite rate (typically 100Mbps) with output buffering can be modelled as shown in Fig. 1.

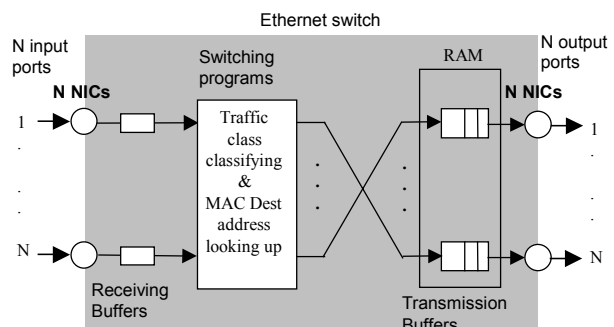


Fig. 1 Model of an Ethernet switch

As all N physical ports are in full-duplex mode we logically have thus N input lines and N output lines. The

switch running with $2N$ times of the line speed is considered since we are only interested by wire-speed switches. We further assume that each output buffer is of infinite capacity (in practice, all buffers dynamically share the same RAM zone) so that the congestion control mechanisms (back pressure or pause command of IEEE802.3x) will never be activated (otherwise real-time guarantees will not be possible). The switching program periodically polls (with period T) the N input ports according to the TDMA principle to classify and switch the arrived packet (stored in the one-place receiving buffer) to its corresponding transmission buffer (called also output buffer). If we note l as switching latency, m as packet length and c as line bit rate, the condition $Nxl \leq T \leq \max(m)/c$ should be met for a wire-speed switch. A packet arriving at an empty output buffer is directly forwarded to the NIC (Network Interface Card) for starting its transmission on the medium during at maximum m/c . Of course if more than one input packets ($\leq N$ packets per T) should be forwarded to a same output port, these packets will suffer an additional buffering delay. We note that in the worst case, the minimum value of T denoted by T_{min} corresponds to the sum of the transmission time of a 64-bytes minimum Ethernet packet and the 96-bits IFS (Inter Frame Spacing, equivalent to $9.6\mu s$ at 10Mbps and $0.96\mu s$ at 100Mbps). In fact, the minimum inter-arrival time at each input is T_{min} since no stations with an Ethernet NIC or shared segments can send packets faster than that to a switch port.

In what follows for analysis convenience we assume a constant packet length with $m/c + IFS = T$. This length can be of 64 bytes if Ethernet is used for transporting small data packets in field device level or can be of 1518 bytes which corresponds to the worst-case. With the above assumptions the analysis of the buffering delay is reduced to analyze one output queue behaviour, which can be seen as an M/G/1-like queue. According to the input traffic pattern, we will analyze the buffering delay in the following cases: Periodic/D/1, Binomial/D/1 and Poisson/D/1.

It is worth noting that in practice, if the core processor can not run at such a high speed to poll all input ports with T , specially designed ASIC often allows parallel processing (reading) of all the N input ports, just as like as a certain interrupt mechanism for signalling a packet arrival.

3. Performance Evaluation

In this section the necessary equations allowing calculating the worst-case buffering delay in case of periodic input traffic and the buffering delay distribution in case of aperiodic input traffic are derived.

3.1. Hard Real Time Guarantee Analysis

For a switch which implements P priorities ($P \leq 8$ according to IEEE802.1p), if M user-priorities should be supported, and if $M > P$, they will be re-mapped according to IEEE802.1p mapping table to the P switch priorities (ensured by traffic classifier).

3.1.1 Scheduling-based analysis

Assuming that M periodic frame-sources (with M or less different priorities) are mapped to P switch priorities. K_i is the number of periodic sources classified to priority i with $i = 1, 2, \dots, P$ representing the priority in decreasing order, and of course $K_1 + K_2 + \dots + K_P = M$ with $0 \leq K_i \leq M$. Then, a periodic source mapped to priority i is modelled by $\{C_i(k_i), T_i(k_i), D_i(k_i)\}$ with $k_i = 1, 2, \dots, K_i$ representing the index of the source generating priority i frames and $C_i(k_i)$ the transmission duration of the frames ($= m_i(k_i)/c$), $T_i(k_i)$ the frame generation period $\leq T$, $D_i(k_i)$ the associated deadline.

Assuming a non pre-emptive fixed priority scheduling in the switch, the worst-case buffering delay can be calculated for each priority by extending the results of Joseph and Pandia [9] and of Lehoczky [10] as following.

Because of the K_i multiple periodic sources per priority i and non pre-emption facts, it is not clear when a frame of priority i with index k_i experiences the worst-case (i.e. maximum response time). This is why we should compute the response time of all k (initialised to 1) consecutive transmissions of messages denoted by $R_{i,k}(k_i)$ generated by source $\{C_i(k_i), T_i(k_i), D_i(k_i)\}$ during a busy period. The maximal $R_{i,k}(k_i)$ will be the worst case response time. This busy period is bounded with the following condition:

$$\sum_{j=1}^i \rho_j = \sum_{j=1}^i \sum_{k_j=1}^{K_j} \frac{C_j(k_j)}{T_j(k_j)} \leq 1$$

which only means, that the normalised load generated by the priority i and all higher priorities should not be greater than 1.

To start a busy period we assume that when a frame of the source $\{C_i(k_i), T_i(k_i), D_i(k_i)\}$ is generated, all frames of higher and equal priorities (i.e., frames of priority 1, ..., $i-1$, and frames generated by the other $K_i - 1$ sources of priority i) arrive at the same time and must be transmitted before the tagged frame and moreover the longest frame of lower priority just began its transmission and will last $B_i = \max_{j>i, k_j} (C_j(k_j))$. So we start the computation to get the initial response time :

$$I_i^0(k_i) = B_i + \sum_{j=1}^i \sum_{k_j=1}^{K_j} C_j(k_j)$$

This value can be underestimated as shorter-period frames might be transmitted more than once during that period. Following iterative equation should be used. In this equation we also take into account the frame number k within this busy period.

$$I_i^{n+1}(k_i) = B_i + kC_i(k_i) + \sum_{j=1}^i \sum_{\substack{k_j=1 \\ k_j \neq k_i}}^{K_j} \left[\frac{I_i^n(k_i)}{T_j(k_j)} \right] C_j(k_j) \quad (1)$$

which could be calculated starting by $I_i^0(k_i)$ until the convergence $I_i^n(k_i) = I_i^{n+1}(k_i) = I_i(k_i)$.

The response time denoted by $R_{i,k}(k_i)$ is given by:

$$R_{i,k}(k_i) = I_i(k_i) - (k-1)T_i(k_i) \quad (2)$$

If $R_{i,k}(k_i) > T_i(k_i)$, during the response time of the k^{th} frame, there is at least another frame which has been generated by the same source. To be sure that it is not the $(k+1)^{\text{th}}$ frame which experiences the longest delay, we have to calculate equation (1) for the next frame ($k = k+1$). If $R_{i,k}(k_i) \leq T_i(k_i)$, we are sure that the busy period is finished and we can stop the calculation.

The worst case response time is then given by:

$$R_i(k_i) = \max_k \{R_{i,k}(k_i)\} \quad (3)$$

This result is also available for the case where aperiodic traffic is present but has less priority than i in order to be eventually considered as blocking factor B_i . Otherwise we should consider an aperiodic traffic as periodic with T_{min} as its period.

3.1.2 (σ, ρ) -based vs. scheduling-based analysis

Another approach that could be used to evaluate the worst-case response time is the CND (Calculus Network Delay) method developed by Cruz [8] and that supposes a more flexible arrival pattern characterized by (σ, ρ) parameters where σ is the maximum burst size (bits) and ρ is the long-term average rate (bit/s). A periodic arrival pattern could be represented by (σ, ρ) parameter where $\sigma = m$ and $\rho = m/T$ where T is the period with which the frame is generated by a periodic source, and m is the frame length.

It has been shown in [8] that for a periodic arrival pattern set (σ_j, ρ_j) $j=1 \dots N$ to a server of c bit/s, the worst case response time for a message of the i^{th} priority is :

$$R_i = \frac{\sum_{j=1}^i \sigma_j + \max_{i+1 \leq j \leq N} (m_j)}{c - \sum_{j=1}^{i-1} \rho_j} \quad (4)$$

A comparative study with the scheduling-based method in [11] showed that the scheduling-based worst-case response-time analysis is more precise than the CND

approach. In fact CND approach gives a larger bound for a given periodic set, but is easier to get. Also, this approach shows its importance for non-regular traffic (bursty) where stochastic and deterministic analyses show their limits.

3.2. Soft Real Time Guarantee Analysis

We assume now that messages are randomly and independently generated by sources (stations directly connected to switch). Messages have SRT constraints and the same frame-length m with $m/c + IFS = T$. This hypothesis of a constant frame-length is usually verified when Ethernet is used for low-level factory communications when the frame length seldom exceeds 64 bytes (minimal Ethernet Frame Size). In Addition, we assume that each incoming frame has a probability of $1/N$ of being destined to any specific output port (This hypothesis could be omitted for a Poisson arrival pattern). Then a buffer is modelled by an M/D/1 queue (or M/G/1 if the frame length is not constant). The contribution of this paper is to compute analytically the distribution for the waiting time in the queue which can be directly used to estimate the probabilistic guarantee for SRT constrained traffic, i.e. the probability that a message will not wait for more than t seconds in a switch, formally denoted by $P[W_q \leq t]$.

3.2.1 Buffering delay distribution in x/D/1 queue

We first give here the general approach for evaluating the waiting time distribution of a x/D/1 queue and then apply it to the case of $x = \text{Poisson}$ and $x = \text{Binomial}$.

Assuming there is n customer in the queue (that in the server is excluded), the waiting time of a customer in x/D/1 queue, denoted by W_q , is given by:

$$W_q = \begin{cases} 0, & \text{if 0 customer in the queue} \\ T', & \text{if 1 customer in the queue} \\ T' + T, & \text{if 2 customers in the queue} \\ \vdots & \vdots \\ T' + (n-1)T, & \text{if n customers in the queue} \end{cases}$$

where T is the service time which is a constant for our x/D/1 queue and T' is the residual service time of the in service customer when the n^{th} customer arrives in the queue.

The waiting time distribution depends thus on that of the customer number n and of the residual service time T' . If $(k-1)T \leq t \leq kT$, we have:

$$P[W_q \leq t] = P[n=0] + \dots + P[n=k-1] + P[n=k]P[T' \leq t - (k-1)T] \quad (5)$$

where $p_k = P[n = k]$ is the steady-state customer number distribution at the customer arrival instants and $P[T' \leq t]$ has the following distribution:

$$P[T' \leq t] = \begin{cases} 1, & t \geq T \\ \frac{t}{T}, & t \leq T \end{cases}$$

and

$$P[T' \leq t - (k-1)T] = \begin{cases} 1, & t - (k-1)T \geq T \\ \frac{t}{T}, & t - (k-1)T \leq T \end{cases} \quad (6)$$

We have finally for $(k-1)T \leq t \leq kT$:

$$P[W_q \leq t] = \sum_{n=0}^{k-1} p_n + p_k \frac{t - (k-1)T}{T} \quad (7)$$

This equation can be directly used for providing probabilistic guarantee on response time.

In the following, we will deal with two cases: **Poisson input** and **Binomial input**. The waiting time distribution for Poisson arrival is given by H. Kobayashi [12] while Binomial waiting time distribution is given by Karol et al. [13]. The motivation to study the Binomial input case is that Binomial arrival models more precisely the input of a switch of N ports than Poisson arrival. Since a Poisson arrival has a non zero probability to have more than N input messages during one slot T while a Binomial one has a finite distribution which has never more than N messages during a slot. Nevertheless the result of Poisson input case could be considered as an upper bound of the Binomial case. So, it can be used to study a switch handling priorities and different port bit rates.

3.2.2 Case of Binomial input

For aperiodic inputs following hypothesis are made:

- All frames are of the same length (so same transmission duration, called a slot and is normalized to 1 time unit),
- Arrival flow to each input port follows independent and identical Bernoulli process of p frames per slot,
- Each incoming frame has a probability of $1/N$ of being destined to any specific output port.

The analysis of the switch is reduced to analysing one of the output buffers. It can be modelled by a M/G/1 queue and studied based on the results of Karol et al. [13].

Equation 8 gives the number of frames in the buffer at the end of the m^{th} slot but just **after** the departure of a frame (if there is one) :

$$Q_{m+1} = \max(0, Q_m + A_{m+1} - 1) \quad (8)$$

Equation 9 (Lindley Equation) gives the number of frames in the buffer at the end of the m^{th} slot but just **before** the departure of a frame (if there is one) :

$$X_{m+1} = X_m + A_{m+1} - U(X_m) \quad (9)$$

Equations 8 and 9 are linked by:

$$Q_m = X_m - U(X_m) \quad (10)$$

where $U(x) = \begin{cases} 0, & x \leq 0 \\ 1, & x > 0 \end{cases}$ and A_{m+1} is the number of

arriving frames at the tagged buffer during the m^{th} slot with the Binomial distribution a_i with $i = 0, 1, \dots, N$

$$a_i = \text{Prob}[A=i] = \binom{N}{i} \left(\frac{p}{N}\right)^i \left(1 - \frac{p}{N}\right)^{N-i} \quad (11)$$

This distribution is independent of the number of the slot. The index m can thus be ignored. The g.f. (Generating function) of a_i is:

$$A(z) = \sum_{i=0}^N z^i a_i = \left(1 - \frac{p}{N} + z \frac{p}{N}\right)^N \quad (12)$$

The number of arrivals to an output buffer during one slot is $Np/N = p$. This leads to a buffer load $\rho = \text{arrival rate} \times \text{service duration} = px1 = p$.

It is worth noting that when $N \rightarrow \infty$, the distribution of A approaches a Poisson one with $\rho = \lambda \times \text{slot_time} = \lambda x1 = p$.

From Equation 10 one can notice that the knowledge on X_m allows to deduce Q_m but on the other hand, if $Q_m = 0$, we can not deduce the value of X_m as it can takes 0 or 1. This is why it is preferred to deal with X_m and then deduce Q_m . Moreover, Equation 9 is the same than that of a M/G/1 queue. The g.f. of X denoted by $X(z)$ is given by equation 13 [14]:

$$X(z) = \frac{(1-\rho)(1-z)A(z)}{A(z)-z} \quad (13)$$

with $P_0 = P[X_m = 0] = 1 - \rho = 1 - p$.

Since $Q_m = X_m - U(X_m)$, the g.f. of Q_m denoted by $Q(z)$ can be obtained as following:

$$\begin{aligned} Q(z) &= E[z^{Q_m}] = E[z^{X_m - U(X_m)}] = \sum_{k=0}^{\infty} z^{k-U(k)} P[X_m = k] \\ &= P_0 + \sum_{k=1}^{\infty} z^{(k-1)} P[X_m = k] = P_0 + z^{-1} [X(z) - P_0] \end{aligned}$$

Substituting $X(z)$ and $\rho = p$ results in:

$$Q(z) = \frac{(1-p)(1-z)}{A(z)-z} \quad (14)$$

Theoretically, Equation 14 gives all performance measures. The probability density function of the number of frames Q can be obtained by successively differentiating Equation 14:

$$P[Q = i] = \frac{1}{i!} \left. \frac{d^i Q(z)}{d^i z} \right|_{z=0} \quad (15)$$

This work can be achieved by using symbolic calculus software such as Maple. But this approach can be not numerically effective for great i .

Karol et al. [13] gave the Embedded Markov Chain of Q as well as $P[Q = i]$:

$$p_0 = P[Q = 0] = \frac{1-p}{a_0}$$

$$p_1 = P[Q = 1] = \frac{1-a_0-a_1}{a_0} p_0$$

...

$$p_n = P[Q = n] = \frac{1-a_1}{a_0} p_{n-1} - \sum_{i=2}^n \frac{a_i}{a_0} p_{n-i} \text{ for } n \geq 2$$

It can be noticed that $P[Q=0] \neq P[X=0] = 1-p$ because these two distributions are related by:

$$P[Q = k] = \begin{cases} P[X = 0] + P[X = 1], & k = 0 \\ P[X = k + 1], & k > 0 \end{cases} \quad (16)$$

Unfortunately, neither Q (Number of frames in the queue at the beginning of a time-slot) nor X (Number of frames in the queue at the end of a time-slot) gives the number of frames in the queue at the arriving point of a frame, which is fundamental for the computation of equation (7). To overcome this problem, Karol and al. [13] gave the distribution of queuing time for the tagged frame which is the "average frame" or randomly chosen one among all frames arrived during the same slot than the tagged one:

$$P[W'_q = kT] = \sum_{n=0}^k p_n \cdot \left(\frac{1}{p} \sum_{i=k+1-n}^{\infty} a_i \right)$$

But, they considered that output ports are synchronized with the input ones. That is not always true for an Ethernet Switch. We add the residual time according to equation 5 or 7 and we get for $(k-1)T \leq t \leq kT$:

$$P[W_q \leq t] = \sum_{m=0}^{k-1} \left(\sum_{n=0}^{m-1} p_n \left(\frac{1}{p} \sum_{i=k-n}^{\infty} a_i \right) \right) + \left(\sum_{n=0}^k p_n \left(\frac{1}{p} \sum_{i=k+1-n}^{\infty} a_i \right) \right) \frac{t - (k-1)T}{T} \quad (17)$$

Figures 2 and 3 show a numerical illustration of this distribution for a 2-port and 8-ports switches

respectively (with $T_c = T$). Figure 4 gives a comparative view of these two switches for a load $\rho = p = 0.9$.

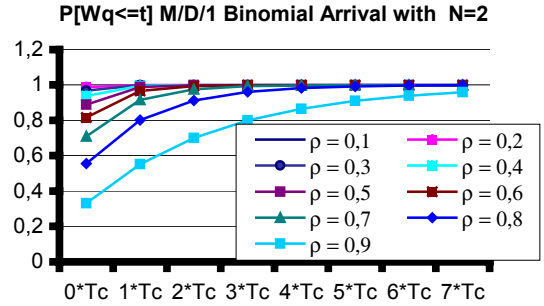


Fig.2 Waiting Time Distribution: Binomial arrival with N=2

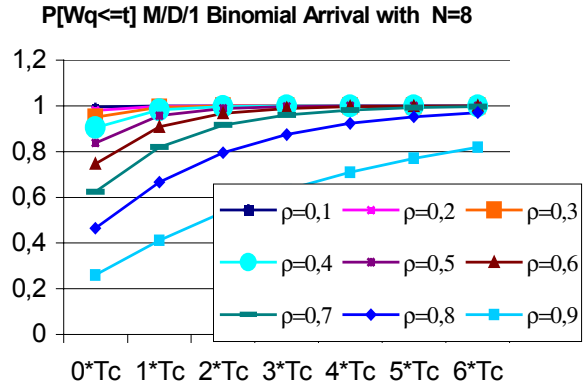


Fig.3 Waiting Time Distribution: Binomial arrival with N=8

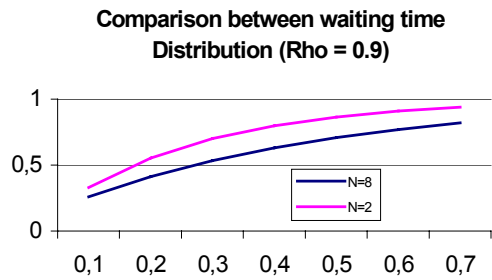


Fig. 4 Comparison of waiting time distributions

3.2.3 Case of Poisson input

For aperiodic inputs following hypothesis are made:

- All frames are of the same length (so same transmission duration, called a slot and is normalized to 1 time unit)
- Arrival flow to each input port follows Poisson process of $\lambda (= p)$ frames per time unit (so per slot)

- Each incoming frame has a probability of $1/N$ of being destined to any specific output port (this hypothesis can be relaxed as the sum of Poisson flows always results in a Poisson flow)

The analysis of the switch is then reduced to analysing one of the output buffers. It can be modelled by a classic M/D/1 queue (the switch is asynchronous, otherwise it should be modelled by a synchronous TDMA system which is studied in [15]). We report here the related result only for self-containing reason.

Equation 9 of the previous section gives the number of frames in the buffer at the frame departure instants with A_{m+1} as the number of arriving frames at the tagged buffer during the m^{th} slot (i.e. during the transmission time of a frame) with the Poisson distribution $a_i = P[A=i / \text{during 1 slot}] = e^{-\lambda} \frac{(\lambda)^i}{i!}$

This distribution is independent of the number of the slot. The index m can thus be ignored. The g.f. of a_i is:

$$A(z) = e^{\lambda(z-1)}$$

The g.f. of X denoted by $X(z)$ takes the form of that of M/G/1 queue and is given by [14]:

$$X(z) = \frac{(1-\rho)(1-z)A(z)}{A(z)-z} \quad (18)$$

with $P_0 = P[X_m = 0] = 1 - \rho = 1 - \lambda \tau$

For M/D/1, we have thus:

$$X(z) = \frac{(1-\rho)(z-1)e^{\lambda(z-1)}}{z - e^{\lambda(z-1)}} \quad (19)$$

Although X represents the frame number at the frame departure points but it has been shown [14] that for Poisson arrival, X represents also the frame number at any randomly chosen instants. For our case, X represents the frame number at frame arriving points.

The probability distribution of the number of frames X can be obtained by successively differentiating Equation 19 using equation 15.

Kobayashi [12] gave the following explicit expression.

$$\begin{aligned} p_0 &= 1 - \rho \\ p_1 &= (1 - \rho)(e^\rho - 1) \\ p_n &= (1 - \rho) \sum_{j=1}^n \frac{(-1)^{n-j} (j\rho)^{n-j-1} (j\rho + n - j) e^{j\rho}}{(n-j)!} \\ &\text{for } n = 2, 3, \dots \end{aligned}$$

which allows us to obtain the buffering delay distribution according to equation 7.

Fig.5 shows the waiting time distribution inside the queue (with $T_c = T$).

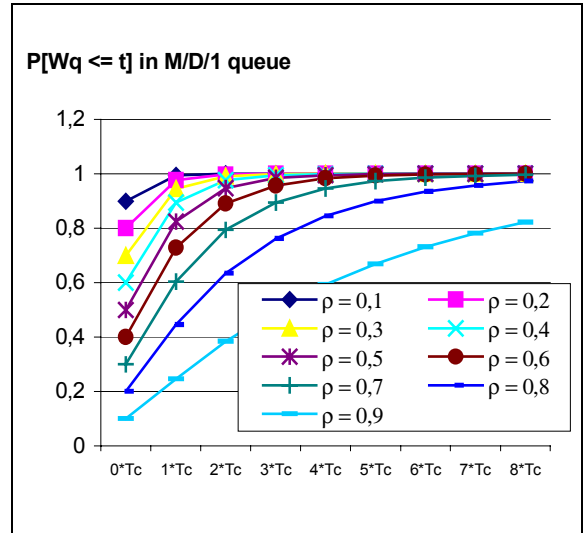


Fig.5 Waiting Time Distribution with Poisson arrival

4. Poisson/D/1 vs. Binomial/D/1

We have obtained the waiting time distribution for the queuing waiting time in the output ports of a switch using two different input stream scenarios.

Fig.6 and Fig.7 show the comparison between a Poisson arrival pattern and the binomial case with a parameter $N=2$ and $N=8$ for loads $\rho = \{0.9; 0.5\}$.

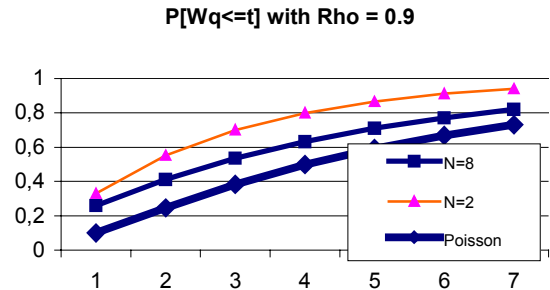


Fig. 6 Comparison for $\rho = 0.9$

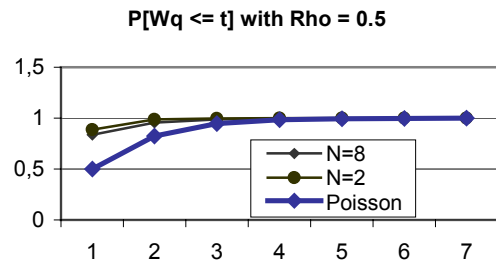


Fig.7 Comparison for $\rho = 0.5$

We have chosen to illustrate the results for high load (≥ 0.5) as we believe that switches guarantee a very low

waiting time for low loads and there is only for a given threshold that the queuing time will cause waiting problems and will be costly. Over the threshold waiting latency will be higher. In practice, increasing output port bit rate enables to smooth this problem.

We noticed from the comparative study, that Poisson case could be efficient to upper bound the binomial arrival pattern. It is clear according to these figures, that the Poisson traffic gives a more pessimistic bound for soft real time guarantee. That means, that Poisson distribution gives more restrictive guarantees but binomial model gives more fine result for the waiting time in the output port of a switch. The disadvantage of the binomial model is that it depends on the number of switch ports. However, it is shown by figures that waiting time for a binomial pattern approaches that of Poisson arrival one when the number of port gets greater. Actually, switch port number could reach 8, 16 and 24 ports; in this case, it is possible to have a good approximation of the waiting time distribution function by using the Poisson pattern instead of binomial one. This approximation enables to get the distribution of waiting time more easily and to extend the study for a priority-enabled switch [6].

5. Conclusion and future work

In this paper we firstly proposed a model of Ethernet switch to analyse the real-time guarantees and then evaluated the message buffering delay in a switch. For periodic messages with fixed priority an extension of the result of classic schedulability analysis is proposed to evaluate the worst-case response time. We also pointed out that the CND method proposed by Cruz [8] can also be used resulting in a less tight upper bound but more easy to calculate (so more suitable for on-line implementation). As for aperiodic messages the equations for computing the probabilistic response time guarantee for single priority class are derived based on the early work of Kobayashi [12] for M/D/1 queue and of Karol et al. [13] for Binomial/D/1 queue. A numerical comparison between M/D/1 and Binomial/D/1 showed that the buffering delay of the Binomial input case can be upper bounded by that of a Poisson one. This allows us to relax many hypothesis (forwarding probability will not have to be $1/N$ to each output port, constant message length) and makes easier the taking into account of multiple priorities and different output port speeds.

In parallel with a theoretic on going work on the stochastic comparison of Poisson/D/1 and Binomial/D/1, our future work aims to extend the present results to a mixed shared and switched Ethernet (as shared Ethernet efficiently supports multicast), to include layer 3 switch (router) and to implement an admission control mechanism over SBM/RSVP for on-line computing the real-time guarantees of the transmission demands [16].

References

- [1] CENELEC European Standard EN50170: *Fieldbus: Vol.1 P-Net, Vol.2 PROFIBUS, Vol.3 WorldFIP*, 1996.
- [2] ISO, *Road vehicles – Interchange of digital information - Controller area network for high-speed communication*, ISO standard 11898, 1994.
- [3] LeLann, G., and N. Rivierre, "Real-time Communications over Broadcast Networks: The CSMA-DCR and The DOD-CSMA-CD Protocols", *INRIA Report RR-1863*, 1993.
- [4] Jasperneite, J. and P. Neumann, "Performance Evaluation of Switched Ethernet in real-time Applications", *Proc. of 4th IFAC FeT'2001*, pp169-176, Nancy (France), Nov. 15-16, 2001.
- [5] Song, Y.Q., "Time Constrained Communication Over Switched Ethernet", *Proc. of IFAC Fet'2001*, pp.177-184, Nancy (France), Nov. 15-16, 2001
- [6] Koubâa, A. and Y.Q. Song « Evaluation de performances d'Ethernet commuté pour des applications temps réel » *Proc. RTS'2002*, Paris (France), 26-28 March 2002.
- [7] Seifert, R., *The Switch Book: The Complete Guide to LAN Switching Technology*, John-Wiley, 2000.
- [8] Cruz, R. L., "A calculus for network delay, Part I", *IEEE Trans. on Information Theory*, 37(1):114-131, Jan. 1991.
- [9] Joseph, M. and P. Pandia, "Finding response times in a real-time system", *The computer journal (British computing society)*, pp.390-395, Vol.29, No.5, Oct. 1986.
- [10] Lehoczky, J.P., "Fixed priority scheduling of periodic task sets with arbitrary deadlines", *Proc. of IEEE Real-time systems symposium, IEEE Computer Press*, pp.201-209, Los Alamitos, CA (USA), 1990.
- [11] Koubâa, A. and Y.Q. Song, "(σ, ρ)-Calculus Based Analysis vs. Worst-Case Response Time Analysis, Which bound is better?" *Internal Report TRIO-LORIA-INRIA* 2002.
- [12] Kobayashi, H., *Modelling and analysis*, Addison-Wesley, Reading MA, 1978.
- [13] Karol, M.J., M.G. Hlchij and S.P. Morgan, "Input vs. output queuing on a space-division packet switch", *IEEE Trans. on Commun.*, Vol. COM-35, No. 12, pp1347-1356, 1987.
- [14] Gross, D. and Carl M. Harris, *Fundamentals of queuing theory*, (2nd edition), John-Wiley.
- [15] Simonot, F., Y.Q. Song and J.P. Thomesse, "On message sojourn time for TDMA schemes with any buffer capacity", *IEEE Trans. on Common.* Vol. 43, No. 2/3/4, pp1013-1021, March 1995.
- [16] Koubâa, A., A. Jarraya and Y.Q. Song, SBM protocol for providing real-time QoS in Ethernet LANs, *Preprint Proc. of Euromicro RTLIA02*, pp45-49, Vienna (Austria), June 18th, 2002.