



HAL
open science

A design process of switched Ethernet architectures according to real-time application constraints

Jean-Philippe Georges, Nicolas Krommenacker, Thierry Divoux, Eric Rondeau

► To cite this version:

Jean-Philippe Georges, Nicolas Krommenacker, Thierry Divoux, Eric Rondeau. A design process of switched Ethernet architectures according to real-time application constraints. *Engineering Applications of Artificial Intelligence*, 2006, 19 (3), pp.335-344. 10.1016/j.engappai.2005.09.004. hal-00144780

HAL Id: hal-00144780

<https://hal.science/hal-00144780>

Submitted on 4 May 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A design process of switched Ethernet architectures according to real-time application constraints^{*}

Jean-Philippe Georges^{*}, Nicolas Krommenacker,
Thierry Divoux, Eric Rondeau

*Centre de Recherche en Automatique de Nancy (CNRS UMR 7039)
Université Henri Poincaré, Nancy, Faculté des Sciences
BP 239, F-54600 Vandœuvre lès Nancy, Cedex, France*

Abstract

Ethernet networks are based on a medium access method which is not deterministic. The use of such networks in factory environments (which are strongly time constraints) can absolutely not guarantee that the applications requirements will be respected. This paper presents a method based on genetic algorithms to minimize end-to-end delays by providing a good distribution of the devices on the network switches. The objective function is defined by using the network calculus which is a deterministic theory and enables to determine bounded delays. In this paper, a case study is described : theoretical results are verified by a real experimentation and compared with results obtained with a network simulator.

Key words: real-time communications, Ethernet, Genetic Algorithms

1 Introduction

The industrial communications are currently based on specific networks called fieldbuses such as FIP, Profibus (CENELEC (1996)), CAN. They interconnect Programmable Logical Controllers (PLC), CNC, robots ... to exchange

^{*} This paper is an extended version of the work presented at IFAC-INCOM 2004, April 2004, Brazil

^{*} Corresponding author.

Email address: <firstname.name>@cran.uhp-nancy.fr (Jean-Philippe Georges).

technical data for monitoring, controlling, and synchronizing industrial processes. Their main characteristic is to ensure that the end-to-end delays of messages remain limited, compared with the time-cycle of the applications. Thus, these networks are deterministic and some protocols satisfy the integrity constraints of the information. In opposition, the Ethernet networks based on the CSMA/CD protocol are more and more used to interconnect industrial devices. Some applications confirm this evolution in different industrial areas : cars (Jaguar), pharmaceuticals (Boehringer Ingelheim), avionics (Grieu et al. (2003)) ... Moreover, different organizations such as the Industrial Automation Networking Alliance (IAONA), the Industrial Ethernet Association (IEA) promote Ethernet as "the standard in the industrial environment". Finally research projects such as CIDER (Alves et al. (2000)) study the use of Ethernet as an enabling technology for future dependable real-time systems.

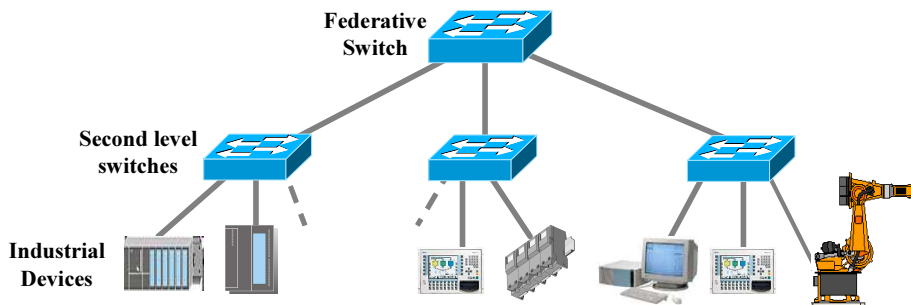


Figure 1. Hierarchical topology for switched Ethernet networks

To interconnect Ethernet switches, different topologies can be used. Rüping et al. (1999) shows that the hierarchical topology (figure 1) is the most efficient and therefore mainly used in factory networks. Industrial devices are distributed on second level switches which are directly connected to a federative switch. Moreover, this topology will support two supplementary mechanisms : full-duplex communications and micro-segmentation. The reason is that on shared Ethernet networks, collisions could occur, which is a non-deterministic issue. Now with micro-segmentation, each device is isolated in its own segment, so that with full-duplex, collisions are eliminated. In these conditions, the collision problem is shifted to congestion in switches. Thus the objective will be to control and to evaluate the switches congestion.

We propose to design a switched Ethernet architecture which satisfies the time requirements. Our objective is to optimize the network organization at the physical layer level. Switched networks design consists in distributing industrial devices on the different Ethernet switches in order to reduce the load of the backbone and of the switches. Figure 2 shows well that the distribution of the devices on the second level switches mathematically corresponds to the partitioning of the communication graph of these devices as described in section 2. As the graph partitioning is an NP-complete problem the Genetic Algorithms are suitable to solve this network segmentation problem. The

Genetic Algorithms have already been used for networks performance optimisation (Krommenacker et al. (2002, 2001)).

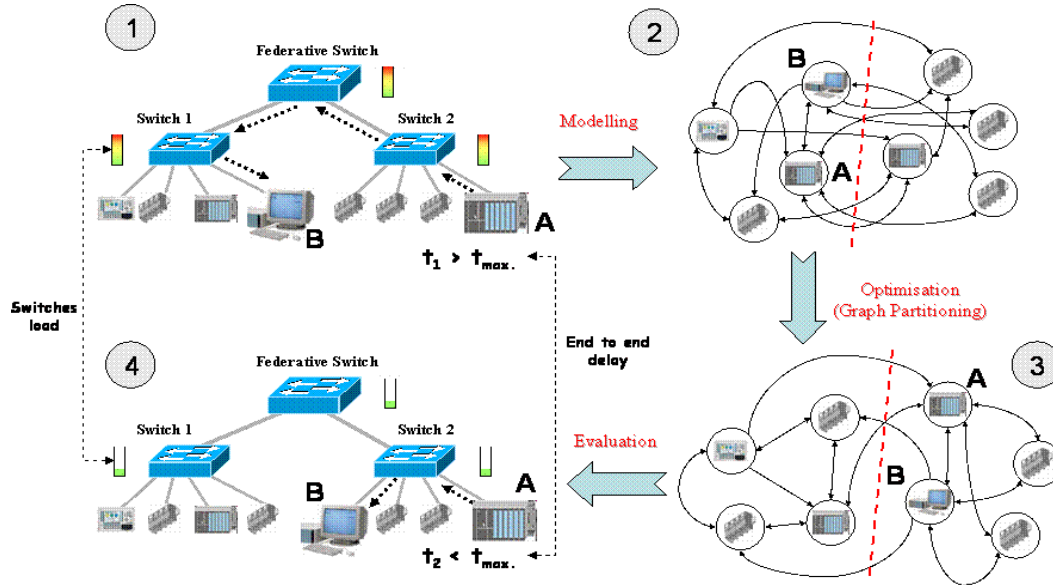


Figure 2. The segmentation problem is a graph partitioning one.

The objective is to collect on the same switches the devices which exchange information. Thus, the criterion to minimize is the number of the graph edges cut by the partition. Indeed the end-to-end delay of a message exchanged between two actors belonging to the same partition will be reduced, because the number of crossed switches is minimized. Nevertheless, even if this delay is optimized, it is not guaranteed that it is lower than the application time constraints. That is why the optimization criterion has to explicitly take into account this delay, and more particularly in the worst case. The Network Calculus theory introduced by R. Cruz (see Cruz (1991a), Cruz (1991b)) enables to compute an upper-bound of the delay. It is presented in the section 4 and validated with a real experimentation.

Finally, the network design approach described in this paper is illustrated on a factory communication scenario. The obtained solution is evaluated on a real experimentation and with a network simulator. The results show the segmentation impacts on the temporal performances. Therefore, it is important to have a method to design a switched Ethernet industrial network in order to make it deterministic under some specific hypothesis.

2 Partitioning problem

2.1 Problem

The graph theory is often used to study networks topology. It is obviously true for switched Ethernet networks as shown by the figure 3 where the dot line represents the distribution of the devices on the two second level switches. In the graph, each industrial device is represented by a vertex and the network communication by an edge between two vertices, i.e. two devices. A partition of this graph will correspond to the mapping of the industrial devices on the different switches.

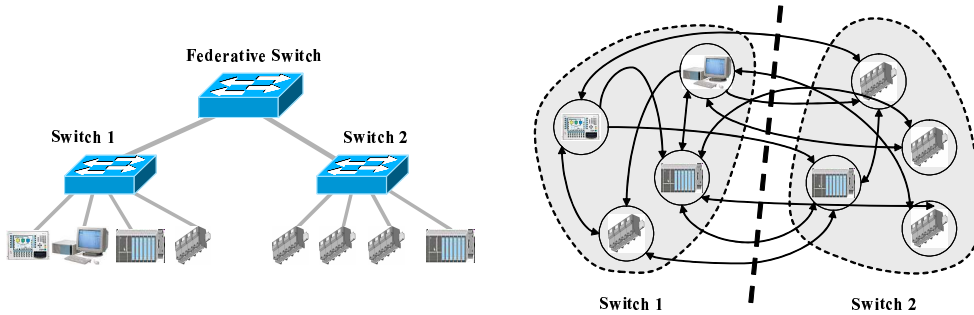


Figure 3. Representation of a switched Ethernet network by a graph.

If communications are not uniform across the graph, then it can be valued by means of weight on the edges (for example, to represent the amount of data exchanged between two devices each second). The weighted graph partitioning problem can be stated as follows :

Given a graph with vertex and edge weights, partition the vertices into disjoint subsets so that the sum of the vertex weights of each subset is closed to the average sum, and the total cost of the cut edges (edges that connect vertices in distinct subdomains) is minimized.

Let $G = (V, E, w)$ be an undirected graph :

$$\begin{cases} V = \{v_1, v_2, \dots, v_n\} & \text{set of vertices} \\ E \subseteq V \times V & \text{set of edges} \\ w : E \mapsto \mathbb{N} & \text{weights of the edges} \end{cases}$$

The k -way partitioning problem consists in dividing the vertices of a given graph into k disjoint subsets called partitions (P) that are subject to some constraints. The partitioning problem can be expressed as the optimization of a single objective (i.e. the edge-cut) :

$$\begin{aligned}
\textbf{Objective:} \quad & \min \sum_{i,j} w_{ij} = \{w(v_i, w_j) \mid v_i \in P_k, w_j \in P_l \forall k \neq l\} \\
\textbf{Constraints:} \quad & \begin{cases} \sum_{i=1}^k P_i = V & (1) \\ P_i \cap P_j = \{\emptyset\} \quad \forall i \neq j & (2) \\ |P_1| \approx |P_2| \approx \dots \approx |P_k| & (3) \end{cases}
\end{aligned}$$

The objective is also to minimize the edge-cut under the constraint that each vertex is associated to a single partition (constraints (1) and (2)) and that the number of vertices in each partition $|P_i|$ is balanced (constraint (3)).

The quality of a graph partitioning is a subjective measurement which depends on the nature of the application. Our objective is to minimize the traffic inside the federative switch and to balance the traffic on the second level switches : When we partition nodes into subgroups, it might happen that some subgroups support a higher traffic than other subgroups. To avoid this situation, we can partition nodes so that the traffic flows in subgroups are balanced. In the graph representation, it consists both in maximizing intra-group exchanges (the total amount of vertex weights assigned to each partition) and in minimizing the inter-group dialogues (minimizing the total weight of edge-cuts). For the balancing problem, the sum of the vertices weights in each sub-group is controlled.

Even if this approach leads to optimise the end-to-end delays, it does not guarantee that they respect a predefined bound time. That is why in the following sections, we propose to use an objective function in which the end-to-end delay is directly formulated.

3 Adaptation of the Genetic Algorithms to the network design

3.1 Introduction

The issue is to determine on which second level switch an industrial device has to be connected. The objective is to propose a segmentation scheme for which the maximum end-to-end delays are minimized in order to guarantee deterministic communications.

Before running a GA, it is necessary to define:

- the individuals on which it operates (*encoding*),
- the operators it uses,
- some parameters such as the population size ... ,

- an objective function.

3.2 Coding structure

For the segmentation problem, several encoding methods could be defined. The most often used is the N-string representation. Each chromosome corresponds to one vector in which the i^{th} element of an individual is j if the i^{th} industrial device is connected to the second level switch labeled j . There are as many elements in the vector as factory devices in the network. The k-ary alphabet $[1, 2, \dots, k]$ is used to code the switch label. For instance, the string $[112332231]$ represents the mapping that assigns devices 1, 2, 9 to switch 1, devices 3, 6, 7 to switch 2 and devices 4, 5, 8 to switch 3. The figure 4 illustrates the encoding of this solution and the decoding of this solution.

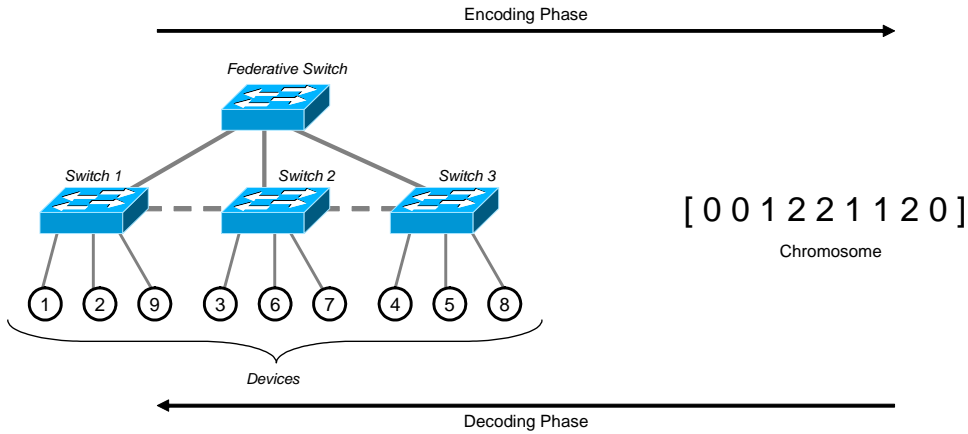


Figure 4. Encoding network topology

3.3 Crossover and Mutation operators

In GA, these operators are the two most important space exploration operators. A crossover operator creates a new offspring chromosome by combining parts of the parents' chromosomes. A two-point crossover is used in our algorithm.

The crossover operator can generate illegal chromosomes. In the figure 6, an offspring violated the number of partitions constraint (3 groups are expected). A repairing procedure is defined : permutations are applied out of the cut-points in order to preserve the scheme of parent chromosomes.

The mutation operator enables to introduce unexplored search space to the population. The most often used method is the *bit-flip* mutation. But it can

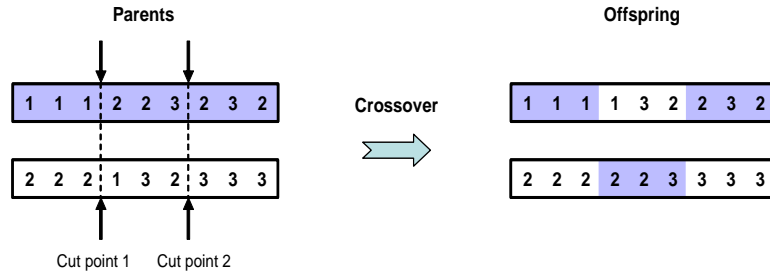


Figure 5. Two-point crossover operator

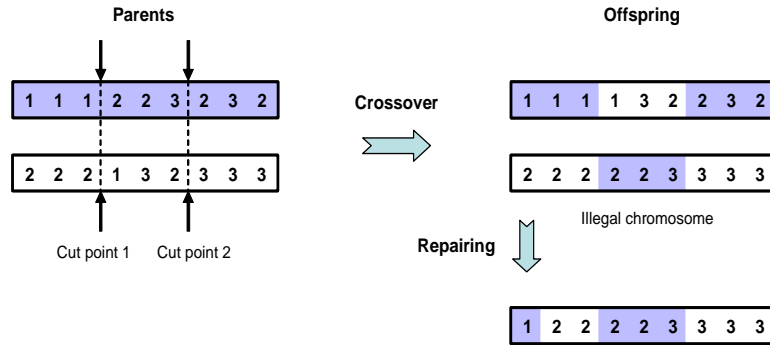


Figure 6. Repairing step

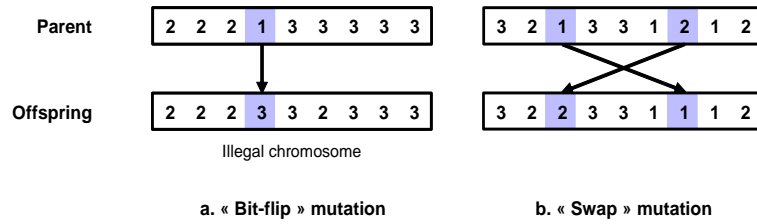


Figure 7. Mutation operators

also generate illegal chromosomes. For example, we can lose a switch in the encoding system (figure 7.a). So a swap mutation has been defined for preserving the switch number constraint (figure 7.b).

3.4 Parameters setting

Goldberg (1989) defined conventional GA parameters such as population size (pop_{size}), crossover (p_c) and mutation (p_m) probabilities. They influence the performance of the algorithm. For example, the size of the initial population influences on the quality of the final solution to the detriment of the execution time. For our problem, optimal values have been computed in Krommenacker (2002).

3.5 Objective function

The quality of a solution will be defined according to the respect of the time-constraints. Thus, to determine the quality of the segmentation, the evaluation function has to take into account the end-to-end delay. In previous works (Kamen et al. (1999), Torab (2000)), the objective consists in reducing the average end-to-end delays which are computed by using stochastic theory. As the question is to know if switched Ethernet guarantees the temporal constraints, a deterministic theory is necessary. The objective function is formulated as :

$$f = \max_i \{ \overline{D}_i - T_i \}$$

with i the message index, \overline{D}_i the message maximum end-to-end delay and T_i the message time constraint). It has to be minimized until it becomes negative (for example, a control message has to be periodically received in a time interval lower than the Programmable Logical Controller time-cycle). The network calculus theory presented in section 4 enables to determine the upper-bound delays \overline{D}_i for each message.

4 Network calculus theory

The network calculus is a deterministic theory of the queuing systems. The main contributions are Cruz (1991a), Le Boudec and Thiran (2001), Chang (2000). It is traditionally used for scheduling (Le Boudec (1998), Cruz (1995)) and for traffic regulation (Chang et al. (2002)) to improve the Quality of Service. The Network Calculus is also more and more used to study Ethernet networks temporal performances (Georges et al. (2002), Jasperneite et al. (2002)).

4.1 Traffic characterization

Since the industrial case gives concise information about communications, deterministic hypothesis on the traffic arrival can be supposed. The modelling of the incoming traffic is traditionally done by using a stochastic model (in Song (2001), it is a Bernoulli process). In this paper, the arrival curve (Le Boudec and Thiran, 2001, definition 1.3.1) notion is chosen. In this model, the traffic is unknown but it is assumed that its arrival satisfies a time constraint. Typically, the quantity of data arrived at a time t will not be more than the arrival curve value at the time t . In fact, the arrival of data is assumed to be constrained by a (σ, ρ) envelop $b(t)$ called leaky bucket (figure 8).

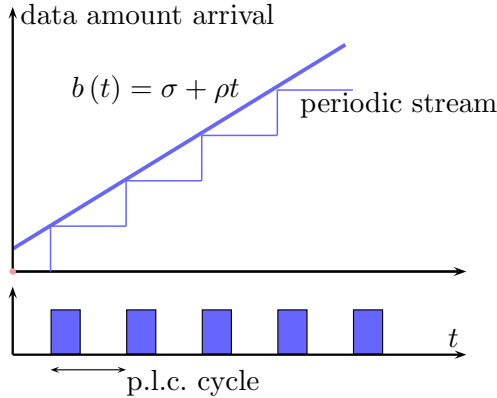


Figure 8. Traffic upper bounding

If $R(t)$ represents the instantaneous rate of data, C the capacity of the links on the network, σ the maximum amount of traffic that can arrive in a burst (initially, it would be the maximum frames length) and ρ an upper bound on the long-term average rate of the traffic flow, then $\forall x, y; y \geq x, x \geq 0$, the burstiness constraint is written $R \sim b$ which means :

$$\int_x^y R(t) dt \leq \min \{C(y - x), \sigma + \rho(y - x)\} \quad (1)$$

In the general case, the notion of arrival curves is usually associated with traffic smoothers because the traffic is unknown and has to be limited (for example for the Internet traffic Caponetto et al. (2002) uses (σ, ρ) regulators). In opposition, in an industrial context, the traffic is better known and it is possible to determine real constraints on the traffic arrival at the application layer such as the periodic messages transmissions during each PLC time-cycle.

4.2 Service modelling of the network devices

In this section, a switched network modelling is proposed. For this, it is necessary to describe the internal behavior of the network devices (the switches). A switch is a complex system which introduces different mechanisms and different technologies. Some models have been proposed : in Song (2001) and in Jasperneite et al. (2002), a blackbox approach is used. In opposition, our contribution is to model in a constructive way as done by Giaccone et al. (2004), but with a more detailed functional decomposition.

Phipps (1999) and Seifert (2000) decompose the switching architecture in three main functional components :

- the queuing model refers to the buffering and the congestion mechanisms located in the switch,

- the switching implementation refers to the decision making process within the switch (how and where a switching decision is made),
- and the switching fabric is the path that data takes to move from one port to another one.

Switches are modelled as presented on the figure 9. It is the result of a study described in Georges et al. (2003) in which several models are built and compared. The model is constituted of a sequence of three basic elements : one multiplexer, one queue and one demultiplexer. In order to take into account the internal speed of a switch, the three following capacities will be used : C , the throughput inside the switch, C_{in} , the throughput of arrival of data on input ports and C_{out} , the output speed.

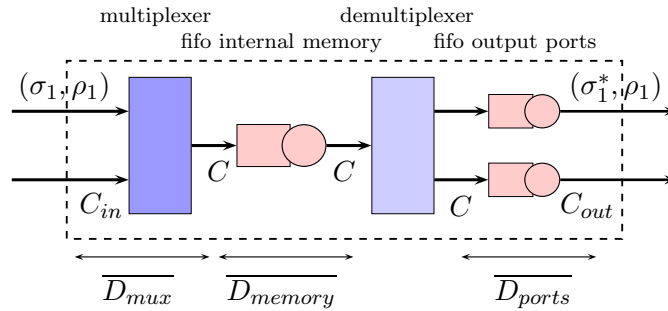


Figure 9. Switch model

In the figure 9, (σ_1, ρ_1) represents the arrival of data on the first input port. In a first step, this traffic will be multiplexed with the others incoming data. Next, data frames are buffered in the shared-memory queue which processes them with a FIFO scheme (representation of Ethernet fairness). As switches are connected themselves or to station by links, the model has to be completed with queues in order to represent the time spent to transmit data on a link which has a capacity C_{out} . In this work, the link transmission mode is supposed to be the full-duplex mode in order to avoid collision problems introduced in the half-duplex one. Thus, frames are finally transmitted on the medium at a rate (σ_1^*, ρ_1) .

The objective is now to mathematically formulate the service offered by each of these elements and to propose a bound for the time needed to cross them. For that, we introduce in a first time the network calculus approach of Cruz.

The quantity of data processed in the element is called the *backlog* ((Le Boudec and Thiran, 2001, definition 1.2.1)) and could be seen as the congestion value of the element. It is defined at time t by the amount of data which is already arrived at time t , minus the amount of data processes at the same time. In the network calculus theory, the delays correspond to the time needed to process the backlog. So, upper bounded delays depend in the worst case on the maximum backlog expression.

Let us consider now the first element used in the model of the figure 9. The multiplexer function consists in forwarding all the streams of the different input links in one output link. For L the maximum length of the frames and C_2 the capacity of the input link 2, (Cruz, 1991a, theorem 4.1) has defined that the transit delay of a 2 input-ports multiplexer FIFO is upper bounded by

$$\overline{D_{mux,1}(t)} = \frac{1}{C_{out}} \max_{t \geq 0} [b_1(t) + b_2(t + L/C_2) - C_{out}t] \quad (2)$$

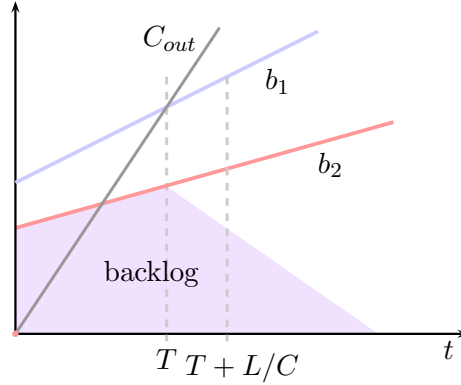


Figure 10. Backlog evolution inside a two-inputs multiplexer FIFO

In the equation (2) (illustrated on the figure 10 where the burst length of the stream 1 is the longest), the maximum backlog value (number of data arrived minus number of forwarded data) is divided by the multiplexer forwarding rate.

The extension of Cruz work in Georges et al. (2002) provides directly : the delay applied to any bit entering in a FIFO m -inputs multiplexer from the stream i is upper-bounded by :

$$\overline{D_{mux,i}} = \frac{1}{C_{out}} \min \{ \overline{B_{mux,i}}, \overline{B_{mux,k}} \} \quad (3)$$

where

$$\begin{aligned} \overline{B_{mux,i}} &= \sum_{z=1; z \neq i}^m b_z \left(\frac{\sigma_i}{C_i - \rho_i} + \frac{L_z}{C_z} \right) + (C_i - C_{out}) \left(\frac{\sigma_i}{C_i - \rho_i} \right) \\ \overline{B_{mux,k}} &= \min_{k; k \neq i} \left\{ \sum_{z=1; z \neq k}^m b_z \left(\frac{\sigma_k}{C_k - \rho_k} - \frac{L_k}{C_k} + \frac{L_z}{C_z} \right) + (C_k - C_{out}) \left(\frac{\sigma_k}{C_k - \rho_k} - \frac{L_k}{C_k} \right) \right. \\ &\quad \left. - \rho_i \frac{L_i}{C_i} + L_k \right\} \end{aligned}$$

The second basic element of the figure 9, the FIFO queue, is constituted of one input buffer in which data are queued, and of one server which forwards this data. This element can be considered as a regulator ($\sigma = 0$, $\rho = C_{out}$)

(Cruz (1991b)). Moreover, the FIFO queue can be viewed as a particular FIFO multiplexer, just with one input port. The analysis developed in Georges et al. (2003) will be used and therefore:

$$\overline{D}_{fifo} = \frac{1}{C_{out}} \frac{(C_{in} - C_{out})}{C_{in} - \rho_{in}} \sigma_{in} \quad (4)$$

This equation can be easily obtained from the previous one, using the fact that there is only one input port.

Finally, the central demultiplexer function is to split the streams that arrive on the input link and to route them to the appropriate output links. In the Ethernet, this work consists to read the destination address at the beginning of the frame and to select in the forwarding table the output port. Moreover the routing strategy is fixed. Therefore it is assumed that the routing step is instantaneously achieved.

4.3 Towards an end-to-end delay evaluation function

In the previous section, upper-bounded delay equations have been proposed for crossing one switch. In these equations, the maximum delay \overline{D} depends on the leaky bucket parameters : the maximum amount of traffic σ that can arrive in a burst and the upper-bound ρ on the long-term average rate. Consequently, we need to know the (σ, ρ) envelop at each point of the network. As shown by the figure 11, the problem is that we only know the initial arrival curve (σ^0, ρ^0) . The other arrival curves, for example after crossing the first switch (σ^1, ρ^1) , have to be determined.

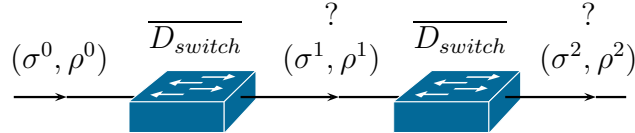


Figure 11. Burstiness along a switched Ethernet network.

In order to resolve the evolution of the burstiness constraint of a flow, Cruz extends the previous method. For a system for which the arrival of data is constrained by b_{in} ($R_{in} \sim b_{in}$) and for which the delay \overline{D} for crossing the system is finite ($\overline{D} < +\infty$), he shows that the output of data is constrained by b_{out} ($R_{out} \sim b_{out}$) with $b_{out}(x)$ defined for all x positive by:

$$b_{out}(x) = b_{in}(x + \overline{D}) \quad (5)$$

And if we develop the previous equation by using the relation (1), we have:

$$\begin{aligned} \sigma_{out} &= \sigma_{in} + \rho_{in} \overline{D} \\ \rho_{out} &= \rho_{in} \end{aligned} \quad (6)$$

In the case of the figure 11, the arrival curve after crossing the first switch will be also defined by $(\sigma^1, \rho^1) = (\sigma^0 + \rho^0 \overline{D}_{switch}, \rho^0)$.

This analysis is based on the fact that the arrival rate stays constant and that the delay is translated in a supplementary burst.

The method steps of the end-to-end delay resolution are :

- (1) To identify all streams on each station and to determine the initial leaky bucket values.
- (2) To identify the route of each stream (in switched Ethernet networks, paths are determined by the spanning tree).
- (3) On each switch, to formulate all streams output burstiness equations as described in the equation 6.
- (4) To define the equation systems under the mathematical form $a_n \sigma_1 + b_n \sigma_2 + \dots + z_n \sigma_m = \delta_n$.
- (5) To calculate the burstiness values.
- (6) To determine the end-to-end delay with

$$\overline{D}_i = \frac{\sigma_i^h - \sigma_i^0}{\rho_i}$$

where h represents the number of crossed switches minus one.

End-to-end delays over the complete network are obtained by computing the equation given at the last step of the method. For the evaluation function included in the Genetic Algorithm, all the maxima end-to-end delays are computed and the obtained value corresponds to the highest delay.

4.4 Validation of the method

In order to confirm the validity of the model, a set of experimental measurements has been carried out. In these experiments, the time needed to transmit one frame from the sender to the receiver on a switched Ethernet network is measured. These results will be then compared with the bound given by the network calculus applied on the model proposed in this paper and also compared with the delays provided by a network simulation tool, Comnet III (figure 15).

The platform is constituted of one *Cisco Catalyst 2912 XL* switch and three stations. Links are always configured at 10Mb/s in the full-duplex mode. *garros* periodically sends frames of 64 bytes of data (the minimum data length in an Ethernet frame) from the first Ethernet (`eth1`) to its second Ethernet interface (`eth0`). The period is fixed at 10 ms. In order to load the switch, a

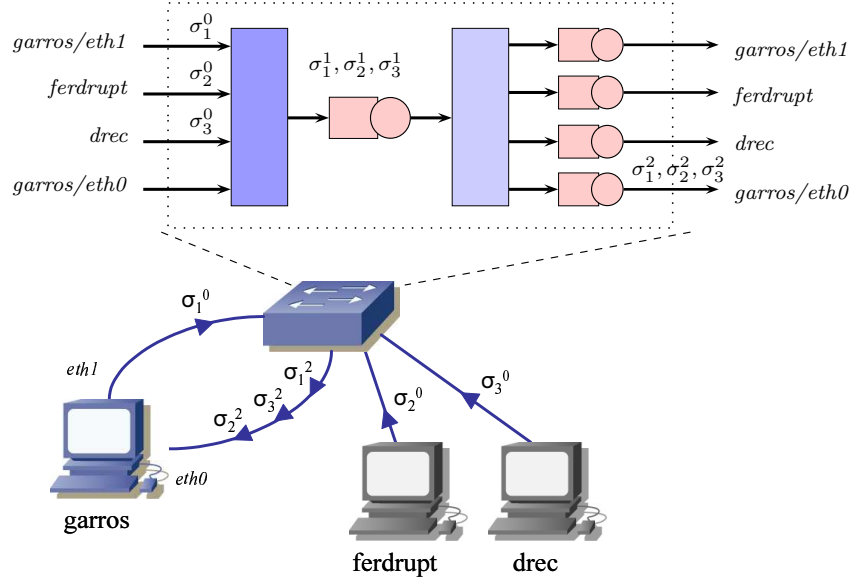


Figure 12. The experimental platform.

background traffic is generated : *ferdrupt* and *drec* send frames of 1500 bytes of data (the maximum data length in an Ethernet frame) each 5 *ms* to the Ethernet interface *eth0* of *garros*.

We apply now the method steps of the end-to-end delay resolution previously presented. First, we identify the initial leaky bucket values of each stream. There are three streams in this network : from *garros/eth1* to *garros/eth0* (stream 1, initial leaky bucket $b_1^0(t)$), from *ferdrupt* to *garros/eth0* (stream 2, $b_2^0(t)$) and from *drec* to *garros/eth0* (stream 3, $b_3^0(t)$). The parameters of the traffic generators give :

$$\begin{aligned} b_1^0(t) &= \sigma_1^0 + \rho_1 t = 72 + 7200t \\ b_2^0(t) &= \sigma_2^0 + \rho_2 t = b_3^0(t) = \sigma_3^0 + \rho_3 t = 1526 + 305200t \end{aligned}$$

As shown on the figure 12, 6 unknowns have to be computed. Also, for each of the basic element of the switch model, we formulate all streams output burstiness. It gives the following equation system :

$$\begin{cases} \frac{C}{\rho_1} \sigma_1^1 = \left(\frac{\rho_1}{C} + 1 \right) \sigma_1^0 + \sigma_2^0 + \frac{\rho_1 + \rho_2}{C - \rho_3} \sigma_3^0 + \left(L_3 - (\rho_1 + \rho_2) \frac{L_3}{C_3} + \rho_2 \frac{L_2}{C_2} \right) & (1.1) \\ \frac{C_{out}(C - (\rho_1 + \rho_2 + \rho_3))}{\rho_1(C - C_{out})} \sigma_1^2 = \left(\frac{\rho_1(C - C_{out})}{C_{out}(C - (\rho_1 + \rho_2 + \rho_3))} + 1 \right) \sigma_1^1 + \sigma_2^1 + \sigma_3^1 & (1.2) \\ \frac{C}{\rho_2} \sigma_2^1 = \sigma_1^0 + \left(\frac{\rho_2}{C} + \frac{\rho_1 + \rho_3}{C_2 - \rho_2} \right) \sigma_2^0 + \sigma_3^0 + \left(\rho_1 \frac{L_1}{C_1} + \rho_3 \frac{L_3}{C_3} \right) & (2.1) \\ \frac{C_{out}(C - (\rho_1 + \rho_2 + \rho_3))}{\rho_2(C - C_{out})} \sigma_2^2 = \sigma_1^1 + \left(\frac{\rho_2(C - C_{out})}{C_{out}(C - (\rho_1 + \rho_2 + \rho_3))} + 1 \right) \sigma_2^1 + \sigma_3^1 & (2.2) \\ \frac{C}{\rho_3} \sigma_3^1 = \sigma_1^0 + \sigma_2^0 + \left(\frac{\rho_3}{C} + \frac{\rho_1 + \rho_2}{C_3 - \rho_3} \right) \sigma_3^0 + \left(\rho_1 \frac{L_1}{C_1} + \rho_2 \frac{L_2}{C_2} \right) & (3.1) \\ \frac{C_{out}(C - (\rho_1 + \rho_2 + \rho_3))}{\rho_3(C - C_{out})} \sigma_3^2 = \sigma_1^1 + \sigma_2^1 + \left(\frac{\rho_3(C - C_{out})}{C_{out}(C - (\rho_1 + \rho_2 + \rho_3))} + 1 \right) \sigma_3^1 & (3.2) \end{cases}$$

These equations are obtained by using the equations (3), (4) and (6). When the burstiness values σ are computed, the next step is to calculate $\overline{D}_1 = \frac{\sigma^2 - \sigma_1^0}{\rho_1}$ in order to determine the maximum end-to-end delay for the stream 1. The result gives : 3025 μs . This reference of 3025 μs will now be compared with the experimental measures. We obtain the results presented on the figure 13.

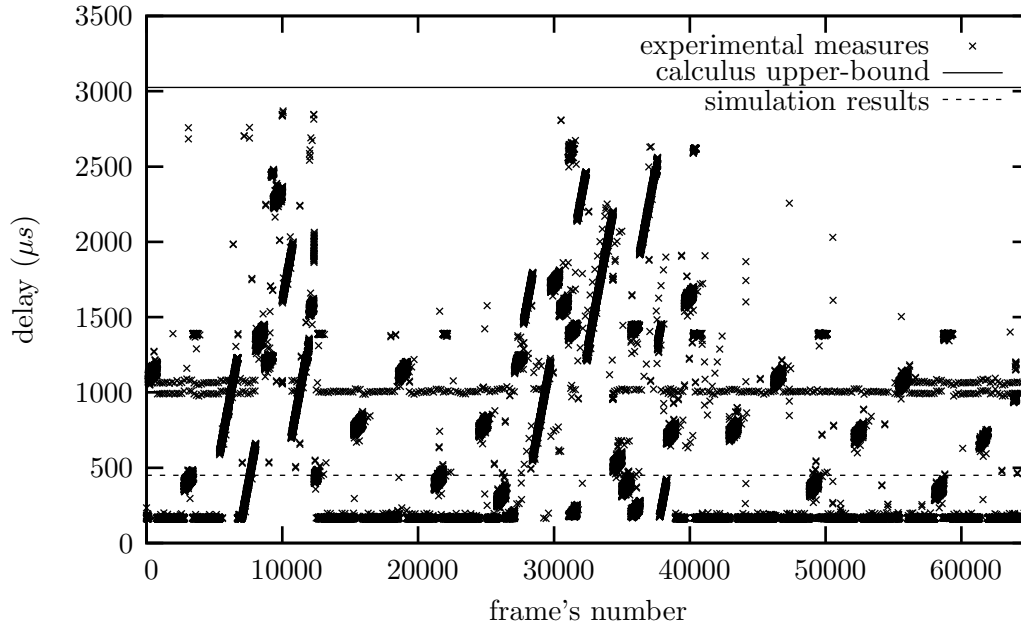


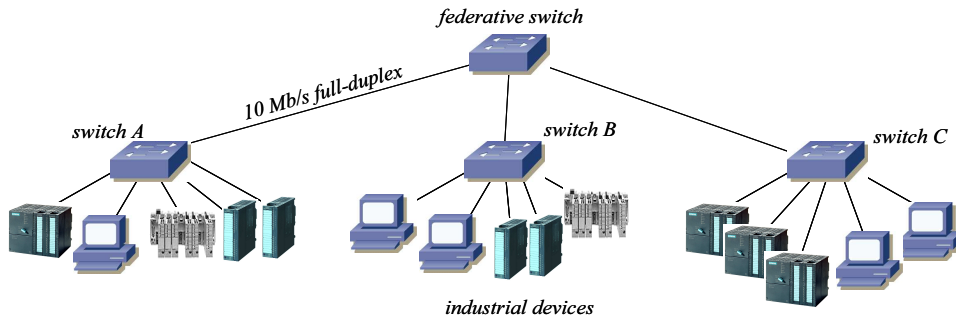
Figure 13. End-to-end delays of each frames sent by garros (stream 1) in μs on the network of the figure 12.

Then, we used the simulation. In Comnet III, a network device such as a switch is modelled by using buffers to represent input and output ports on which are specified both buffer sizes and buffer processing times (CACI Products Company (1998)). It also uses internal buses for moving frames from one port to another one.

On this example, the simulation tool gives a maximum delay of 450 μs . First, if we compare it to the bound provided by the Network Calculus, we can conclude that the over-estimation of the network calculus is very important. But if we compare now this bound to the experimental results presented on the figure 13, we can remark that even if almost all measures are very inferior to this bound and to the simulation value, some measures tend to the calculus bound. Indeed, more than 62 % of the measures are inferior to 450 μs (the average of the measures is 608 μs). Some observed delays grow up to 2869 μs , i.e. nearly 6 times more. It shows that a simple capacity analysis (the load of the link between the switch and the network interface `eth0` of `garros` is less than 50 %) is not sufficient to ensure that an Ethernet network will respect the industrial requirements. Moreover, it shows that calculus bounds are closed to the greatest measures since the over-estimation is only about 5 % of the maximum delay measured.

5 A case study

The first step is to determine the number of switches on which the application has to be distributed. It can be imposed by the designer who can also use graph partitioning techniques (Adoud et al. (2001)).



(a) Network architecture

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1						2									
2	1											3			
3					1						4		1		
4							1	1							
5				1								1			
6	1			2	1					2	4				
7															
8			3												4
9															
10															
11					1			1				1			2
12	3			1											
13										3					
14	1			2											2
15					1				1	1	3				

(b) Exchange matrix

Figure 14. The case study network.

In order to show the benefits provided by the optimization of the segmentation of a switched Ethernet network, the factory communication scenario of the figure 14 is studied. Fifteen industrial devices have to be interconnected on three switches. The exchanges are represented on the figure 14(b). The weights defined on the exchange matrix represent the amount of data sent by a device to another one. For example, device 3 sends to device 11, $4 * (\text{basic length})$ per $\text{time} - \text{cycle}$ bytes per second. If it is supposed that the basic length is 64 bytes (minimum size of an Ethernet frame) and that a device sends a packet each 2 ms (P.L.C. time-cycle). So the rate of arrival of data from device 3 to 11 is 5, 12 kbytes/s.

In order to verify that a hierarchical switched Ethernet architecture could be used in this example, with end-to-end delays always lower than the time-cycle, the Genetic Algorithm with the objective function computed with the network calculus is launched. An arbitrary distribution and an optimized solution are compared. The first one distributes devices $\{1, 2, \dots, 5\}$ on the first switch, $\{6, \dots, 10\}$ on the second one, $\{11, \dots, 15\}$ on the third one. The GA provides the following distribution called 'optimized' architecture and collects the industrial devices $\{1, 4, 6, 7, 14\}$ on the first switch, $\{8, 9, 11, 12, 15\}$ on the second one, $\{2, 3, 5, 10, 13\}$ on the third one. The results are confronted with those provided by *Comnet III* (figure 15).

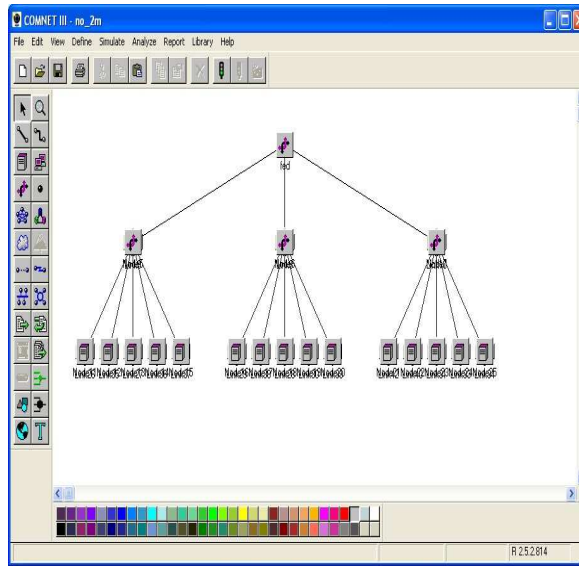


Figure 15. The network simulator *Comnet III*

Moreover, in order to improve the performance evaluation of the design process, experimental measurements have been carried out both for the arbitrary and the optimized architectures. The principle of these measures is the same than for the validation of the method presented in section 4.4. Using the network calculus, the worst communication delays have been identified : exchange from 12 to 1 for the arbitrary topology and exchange from 6 to 11 for the optimized topology. They are given on the table 1. To note that for the arbitrary architecture, the network calculus investigates up to 188 unknowns burtiness values when for the optimized solution, the number of unknowns is only 160. It is due to the fact that paths are shorter in the optimized solution. That could be seen as a first indicator of the interest of our proposition.

The table 1 shows that delays can be reduced if a methodology for distributing the industrial devices on the switches is applied. Indeed, in the optimized solution, the network calculus theory guarantees that maximum delays will be smaller than the time-cycle (2ms), when the non-optimized distribution exceeds it. The simulator results also show that the end-to-end delays are reduced. To note that, the results provided by the simulator are lower than the

Table 1

Performance of the architectures (values are in millisecond)

architecture	network calculus	comnet	experimental
arbitrary	2,55	1,68	2,38
optimized	1,96	1,53	1,94

time-cycle. So the arbitrary solution could be considered as suitable regarding the application constraints. Nevertheless, the experimental measurements confirm the network calculus results. In fact, the values obtained by simulation represent more the more frequent case than the worst one. The simulation does not provide such formal validation due to its probabilistic approach. The set of experimental measures validate the design process presented in this paper. And more generally, this case study shows the pertinence of the method : to provide an optimized segmentation for which we are sure that the application constraints will be respected.

6 Conclusion

In this paper, an algorithm to design hierarchical switched Ethernet networks has been given. The optimization heuristic used is the Genetic Algorithm for which a new objective function has been developed. This function evaluates the maximum end-to-end delays of the messages over the network. Bounds are computed by using the network calculus theory, which considers the worst case. Thus, from a well-known incoming traffic, it is possible to guarantee deterministic performances for the communication architecture. This approach enables to valid the network topology and proposes a distribution of the industrial devices which respects the temporal application constraints, even if this heuristic does not provide the best solution.

In future works, the presented method will be analyzed in order to improve the convergence of the algorithm. Indeed, its running time is important (2 minutes for the case study described in the paragraph 5). It is not a problem for the design of static applications. Nevertheless, it has to be reduced for applications which dynamically evolve in terms of number of interconnected devices, traffic flows and default occurrences.

Supplementary constraints could be taken into account such as the cabling cost, the location of the end equipments which limits the number of possible solutions (and the algorithm execution time, since the search space is reduced). The method could be also recursively processed in order to get a n levels hierarchical topology as it is done with the spectral algorithm (Simon (1991)).

References

- Adoud, H., Rondeau, E., Divoux, T., oct 2001. Configuration of communication networks by analysing co-operation graphs. *Computer Communications* 24 (15–16), 1568–1577, ISSN 0140-3664.
- Alves, M., Tovar, E., Fohler, G., Buttazo, G., jun 2000. CIDER - envisaging a cots communication infrastructure for evolutionnary dependable real-time. In: *Proceedings of the 12th Euromicro Conference on Real-Time Systems*. pp. 19–22, work in progress session.
- CACI Products Company, 1998. Comnet III reference guide. Release 2.0.
- Caponetto, R., Lo Bello, L., Mirabella, O., jun 2002. Fuzzy traffic smoothing : Another step towards statistical real-time communication over Ethernet networks. In: *Proceedings of the 2002 International Workshop on Real-Time LANs in the Internet Age*.
- CENELEC, 1996. Fieldbus; vol.1 p-net, vol.2 profibus, vol. 3 WorlFip. European Standard EN50170.
- Chang, C., 2000. *Performance Guarantees in Communication Networks. Telecommunication Networks and Computer Systems*. Springer Verlag, ISBN 1-85233-226-3.
- Chang, C., Cruz, R., Le Boudec, J.-Y., Thiran, P., oct 2002. A min-plus system theory for constrained traffic regulation and dynamic service guarantees. In: *IEEE/ACM Transactions on Networking*. Vol. 10. pp. 805–817.
- Cruz, R., jan 1991a. A calculus for network delay, part I : Network elements in isolation. *IEEE Transactions on Information Theory* 37, 114–131.
- Cruz, R., jan 1991b. A calculus for network delay, part II : Network analysis. *IEEE Transactions on Information Theory* 37, 132–141.
- Cruz, R., aug 1995. Quality of service guarantees in virtual circuit switched networks. *IEEE Journal on Selected Areas in Communications* 13, 1048–1056.
- Georges, J.-P., Divoux, T., Rondeau, E., sep 2003. Comparison of switched Ethernet architectures models. In: *IEEE Conference on Emerging Technologies and Factory Automation Proceedings*. pp. 375–382.
- Georges, J.-P., Rondeau, E., Divoux, T., aug 2002. Evaluation of switched Ethernet in an industrial context by using the network calculus. In: *4th IEEE International Workshop on Factory Communication Systems*.
- Giaccone, P., Leonardi, E., Prabhakar, B., Shah, D., 2004. Delay bounds for combined input-output switches with low-setup. *Performance Evaluation* 55, 113–128.
- Goldberg, D., 1989. *Genetic Algorithms in Search Optimization*. Addison Wesley.
- Grieu, J., Frances, F., Fraboul, C., sep 2003. Preuve de déterminisme d’un réseau embarqué avionique. In: *Colloque Francophone sur l’Ingénierie Des Protocoles*. Hermes Lavoisier, pp. 287–303.
- Jasperneite, J., Neumann, P., Theis, M., Watson, K., aug 2002. Deterministic real-time communication with switched Ethernet. In: *4th IEEE Interna-*

- tional Workshop on Factory Communication Systems. pp. 11–18.
- Kamen, E. W., Torab, P., Cooper, K., Custodi, G., dec 1999. Design and analysis of packet-switched networks for control systems. Proceedings of the 38th Conference on Decision and Control, 4460–4465.
- Krommenacker, N., 2002. Heuristiques de conception de topologies réseaux : Application aux réseaux industriels. Ph.D. thesis, Université Henri Poincaré Nancy 1.
- Krommenacker, N., Divoux, T., Rondeau, E., aug 2002. Genetic algorithms for industrial Ethernet network design. In: 4th IEEE International Workshop on Factory Communication Systems. pp. 149–155.
- Krommenacker, N., Rondeau, E., Divoux, T., Oct 2001. Study of algorithms to define the cabling plan of switched Ethernet for real-time applications. In: Proceedings of the 8th IEEE International Conference on Emerging Technologies and Factory Communication. Antibes, pp. 223–230.
- Le Boudec, J.-Y., may 1998. Application of network calculus to guaranteed service networks. IEEE Transactions on Information theory 44 (3), 1087–1096.
- Le Boudec, J.-Y., Thiran, P., 2001. Network Calculus, A Theory of Deterministic Queueing Systems for the Internet. Lecture Notes in Computer Science. Springer Verlag, ISBN 3-540-42184-X.
- Phipps, M., 1999. A guide to LAN switch architectures and performance. Packet CISCO Systems users magazine 4, 54–57.
- Rüping, S., Vonnahme, E., Jasperneite, J., sep 1999. Analysis of switched Ethernet networks with different topologies used in automation systems. In: IFAC International Conference on Fieldbus Systems and their Applications. pp. 351–358.
- Seifert, R., 2000. The Switch Book. John Wiley and Sons, New York, ISBN 0-471-34586-5.
- Simon, H., 1991. Partitioning of unstructured problems for parallel processing. Computing Systems in Engineering 2, 135–148.
- Song, Y., nov 2001. Time constrained communication over switched Ethernet. In: IFAC International Conference on Fieldbus Systems and their Applications. pp. 152–169.
- Torab, P., 2000. Performance analysis of packet switched networks with tree topology. Ph.D. thesis, School of Electrical and Computer Engineering, Georgia Institute of Technology.