

Worst-case end-to-end delay analysis of an avionics AFDX network

Henri Bauer^{1,2}, Jean-Luc Scharbarg², Christian Fraboul²

¹ Airbus France - 316 Route de Bayonne 31300 Toulouse, France

² Université de Toulouse - IRIT/ENSEEIH/INPT - 2, rue Camichel 31000 Toulouse, France
{Henri.Bauer, Jean-Luc.Scharbarg, Christian.Fraboul}@enseeiht.fr

Abstract—AFDX (Avionics Full Duplex Switched Ethernet) standardized as ARINC 664 is a major upgrade for avionics systems. But network delay analysis is required to evaluate end-to-end delay's upper bounds. The Network Calculus approach, that has been used to evaluate such end-to-end delay upper bounds for certification purposes, is shortly described. The Trajectory approach is an alternative method that can be applied to an AFDX avionics network. We show on an industrial configuration, in which cases the Trajectory approach outperforms the existing end-to-end delays upper bounds and how the combination of the two methods can lead to an improvement of the existing analysis.

I. INTRODUCTION

Designing and manufacturing new civilian Aircraft has lead to an increase of the number of embedded systems and functions. In order to preserve performance and maintainability, the concept of common resource sharing has been introduced at platform level with the IMA (Integrated Modular Avionics) concept [1] and at network level with communication multiplexing.

The traditional mono-emitter ARINC 429 bus [2] with limited bandwidth cannot cope with new communication needs in terms of weight and complexity because of the number of needed buses. The AFDX [3] brings an answer by multiplexing huge amounts of communication flows over a full duplex switched Ethernet network. It has become the reference communication technology in the context of avionics and provides a backbone network for the avionics platform.

The main problem is that full duplex switched Ethernet eliminates the inherent indeterminism of vintage (CSMA-CD) Ethernet but it shifts in fact the indeterminism problem to the switch level where various flows can enter in competition for sharing resources of a given switch (FIFO buffering of output ports).

In fact, an AFDX network includes specific mechanisms in order to guarantee the determinism of the avionics communications. In order to prove this determinism for certification purposes, the problem is to demonstrate that end-to-end communication delays can be upper bounded.

Main AFDX specific assumptions deal with the static definition of avionics flows that can be described as multicast links. All the flows are asynchronous, but have to respect a bandwidth envelope (burst and rate) at network ingress point. Each flow is statically mapped on the network of interconnected AFDX switches.

For a given flow, the end-to-end communication delay of a frame can be described as the sum of transmission delays on links and latencies in switches. As the links are full duplex there is no packet collision on links [4]. The transmission delay depends only on the transmission rate (typically 100Mb/s) and on the frame length (less than or equal to the 1518bytes Ethernet maximal frame size). On the other hand, the latency in switches is highly variable because of the confluence of asynchronous flows, which compete on each switch output port (according to a servicing policy). Therefore, it is necessary to analyze precisely the latency in every switch output port in order to determine upper bounds on end-to-end delay and jitter of each flow.

The upper bounds on avionics networks are computed with the Network Calculus method [5]. Though, [6] proposes an alternative method to determine such upper bound with the Trajectory approach, applied to AFDX networks. The Trajectory approach outperforms the Network Calculus by providing tighter bounds on average, but there are some scenarios in which, the Network Calculus keeps the lead over the Trajectory approach. The goal of this paper is analyze the performances of both methods in order to determine the parameters that favor one method over the other.

The paper is organized as follows. Section II presents the AFDX end-to-end delay analysis problem, both network calculus and trajectory approaches which are considered in this paper and results obtained on an industrial configuration with these approaches. Section III analyzes these results and exhibits trends on the performance of the two approaches. Section IV concludes this study and gives directions for future research.

II. AFDX END-TO-END DELAY ANALYSIS

A. The AFDX network context

The AFDX is a switched Ethernet network taking into account avionic constraints. An illustrative example is depicted in Figure 1. It is composed of five interconnected switches $S1$ to $S5$. Each switch has no buffers on input ports and one FIFO buffer for each output port. The inputs and outputs of the network are called *End Systems* ($e1$ to $e10$ in Figure 1). Each end system is connected to exactly one switch port and each switch port is connected to at most one end system. Links between switches are all full duplex.

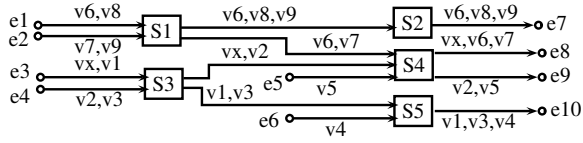


Fig. 1. An illustrative AFDX configuration

The end-to-end traffic characterization is made by the definition of *Virtual Links*. As standardized by ARINC-664, Virtual Link (VL) is a concept of virtual communication channel. Thus it is possible to statically define all the flows (VL) which enter the network [3].

End systems exchange Ethernet frames through VLs. Switching a frame from a transmitting to a receiving end system is based on VL. The Virtual Link defines a logical unidirectional connection from one source end system to one or more destination end systems. Coming back to the example in Figure 1, vx is an unicast VL with path $\{e3-S3-S4-e8\}$, while $v6$ is a multicast VL with paths $\{e1-S1-S2-e7\}$ and $\{e1-S1-S4-e8\}$.

The routing is statically defined. Only one end system within the avionics network can be the source of one Virtual Link, (*ie.* mono transmitter assumption). A VL definition also includes the Bandwidth Allocation Gap (BAG), the minimum and the maximum frame length (s_{min} and s_{max}). BAG is the minimum delay between two consecutive frames of the associated VL (which actually defines a VL as a sporadic flow).

Each VL parameters (BAG , s_{max}) are assured by a shaping unit at end system level and a traffic policing unit at each switch entry port (specificity of AFDX switches, compared to standard Ethernet switches).

All these strong constraints that the AFDX model adds to a classical full duplex switched Ethernet enables a precise analysis of the network, using either worst case Network Calculus or schedulability analysis.

B. Worst-case end-to-end delay analysis

Different approaches have been proposed in order to analyze flow's end-to-end delay and jitter. Among them, the deterministic Worst Case Network Calculus [7], [8] has been used in a first step to compute upper bounds, mainly for avionics network certification purposes [9], [10], [11]. This worst case communication delays analysis gives also intermediate information on latency time in switch output ports, which permits to scale the switch memory buffers and avoid buffer overflows (*ie.* frame losses). But this worst case delay analysis is obviously pessimistic. The Network Calculus is a *holistic* approach [12] and the worst case scenario is considered on each node visited by a flow, taking into account maximum possible jitter introduced by previously visited nodes. This approach can indeed lead to impossible scenarios. There are also other pessimism causes, intrinsic to the Network Calculus theory as envelopes are used instead of the exact arrival curve and service curves.

The network calculus approach has been improved in the AFDX context, thanks to the grouping technique. It has brought a significant improvement to end-to-end delay upper bounds (25 % in average on an industrial configuration). The grouping technique is a refinement of the network modeling in order to take into account the serialization of flows which already share a common link. Informally, the worst-case incoming traffic in a switch output port is divided and grouped by flows coming from the same source (*ie.* transmission link). Each group is shaped by a leaky bucket with a burst equal to the largest frame size and a rate equal to the rate of the source.

More recently, the trajectory approach has been proposed for response time analysis [13]. This approach has been applied to the AFDX context in [6]. It is based on the analysis of the worst-case scenario experienced by a packet on its trajectory. It considers the longest busy period in each crossed node. A busy period is a time interval during which there is always at least a packet to process. As an example, let's consider the small configuration in figure 2. All the VLs in this configuration

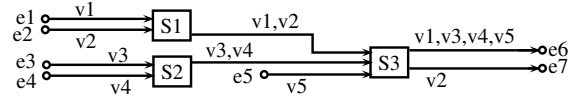


Fig. 2. Illustrative AFDX configuration

have identical characteristics : $BAG = 4000\mu s$ and $s_{max} = 4000bits$. The entire network works at $R = 100Mb/s$ and the technological latency in an output port is $L = 16\mu s$. There are five end systems ($e1$ to $e5$) working in emission and two end systems in reception ($e6$ and $e7$). Each sending end system emits one VL. They all arrive at end system $e6$, except for $v2$ which ends at end-system $e7$. End systems are interconnected thanks to three AFDX switches ($S1$ to $S3$).

The trajectory approach considers the scenario depicted in figure 3 in order to calculate the worst-case delay for $v1$. The

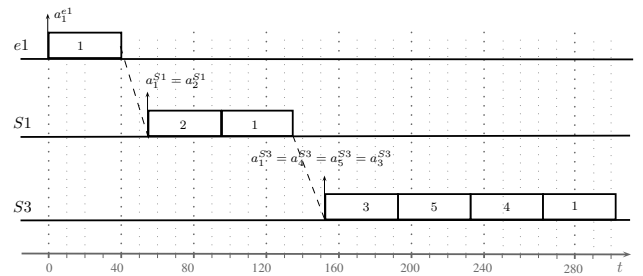


Fig. 3. Trajectory approach worst-case for $v1$

obtained delay is $312 \mu s$. This scenario is impossible, since $v3$ and $v4$ reach $S3$ from the same input link, *ie.* $S2-S3$. Consequently, frames from $v3$ and $v4$ cannot arrive in $S3$ at the same time. The scenario in figure 3 considers that they arrive at the same time.

This problem can be handled by the grouping technique previously introduced in the network calculus approach. This

grouping technique has been successfully introduced in the trajectory approach [6], with similar improvements on end-to-end delays upper bounds. Then, the scenario considered by this enhanced trajectory approach, in order to calculate the worst-case delay of $v1$, is depicted in figure 4.

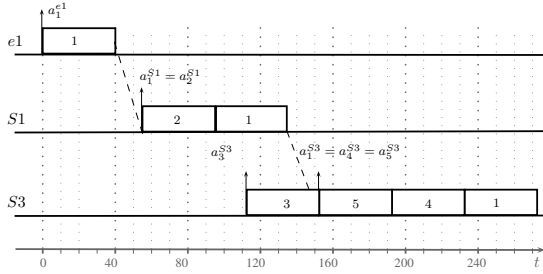


Fig. 4. Enhanced worst-case trajectory approach for $v1$

C. Existing results on an industrial configuration

The tool which has been developed implements both network calculus and trajectory approaches in the AFDX context. Then, the two approaches have been applied to an industrial AFDX configuration including nearby 1000 virtual links corresponding to more than 6000 paths. The configuration is more precisely described in [5]. It interconnects Aircraft functions in the avionics domain. It includes more than one hundred end systems and two redundant AFDX sub-networks, each composed of eight switches. The obtained results are presented in [6]. Table I shows, first the benefit obtained by the trajectory approach over the network calculus one (first column), second the benefit obtained by a combined approach over the network calculus one. The combined approach keeps for each VL path the best obtained by either trajectory or network calculus approach. The column of tab The mean benefit of Trajectories

Benefit	Trajectory/WCNC	Best/WCNC
Mean	10.46%	10.71%
Maximum	34.02%	34.02%
Minimum	-18.39%	0%

TABLE I

END-TO-END DELAY BOUND COMPARISON ON AN INDUSTRIAL NETWORK

over network calculus is over 10%. Trajectories are better for roughly 92 % of VL paths. In the best case (*ie.* for a given VL path), this improvement raises up to 34%. For the remaining 8 % of VL paths, the bound provided by the trajectory approach is more pessimistic than the one obtained with network calculus. In the worst case, this bound is 18% more pessimistic.

The benefit obtained with the combined approach is very close to the one obtained with trajectories, but this time, the bound is of course never worse than the one obtained by the network calculus approach.

The next section analyzes these results. The goal is to exhibit some global trends that favor one or the other approach.

III. COMPARATIVE ANALYSIS OF THE TWO APPROACHES

This comparative analysis proceeds in two steps. First, the results obtained on the industrial configuration are considered. The goal is to identify the parameters which impact the performances of the two approaches. Second, the influence of these parameters is more precisely studied on simpler configurations.

A. Industrial configuration analysis

In AFDX networks, the flow admission control relies on two parameters: BAG and s_{max} . These parameters are sufficient to define a maximal rate of flow, corresponding to a leaky bucket which size is s_{max} and the leak rate is $\frac{s_{max}}{BAG}$.

Therefore, we consider the BAG and s_{max} as potential key parameters for performance evaluation between Trajectories and network calculus.

1) *The impact of the BAG:* First, we calculate the average benefit of the Trajectory approach over the Network Calculus one for each possible value of the BAG. Results are depicted on figure 5. On this industrial application, BAG values are harmonic between 2 ms and 128 ms. For each BAG value,

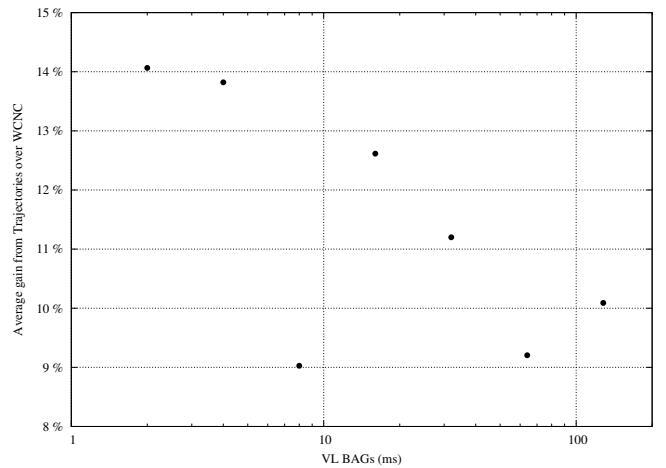


Fig. 5. Mean benefit of Trajectories over network calculus for every possible BAG values

the Trajectory approach outperforms the network calculus in average and the benefit globally increases when the value of the BAG decreases. Since VLs with short BAG s cause potentially higher network loads, This could mean that the trajectory approach is less impacted than the network calculus one by the increase of the network load.

2) *The impact of s_{max} :* Second, we calculate for each possible value of s_{max} the ratio of VL paths for which the trajectory approach does not improve the network calculus result. Figure 6 shows the obtained results. The only limitation for s_{max} is the Ethernet frame size (between 64B and 1518B). The trajectory approach always gives the best results when $s_{max} \geq 900B$. The ratio of VL paths for which the network calculus approach gives the tightest bounds globally increases when s_{max} decreases. Once again, higher s_{max} values correspond to VLs that cause higher network loads. Thus, the

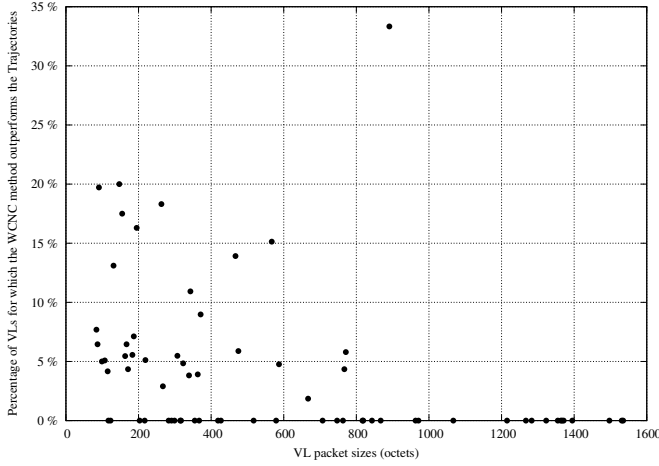


Fig. 6. Percentage of VL paths for a given s_{max} for which Trajectory bound is more pessimistic than the network calculus bound

Trajectory approach seems to outperform the network calculus for highly loaded networks.

3) *Summary*: From this small empirical analysis, it seems that BAG and s_{max} impact the relative performances of trajectory and network calculus approaches. In the next section, a more dedicated analysis is conducted on a sample configuration in order to evaluate more precisely the influence of both BAG and s_{max} .

B. Parameter influence analysis on a sample configuration

Let us come back to the sample AFDX configuration depicted in Figure 2. All the VLs of this configuration have a BAG of 4 ms and a s_{max} of 500B. The idea is to vary the s_{max} and BAG parameters of VL v_1 and to observe the evolution of the end-to-end delay bounds provided by the network calculus and the trajectory approaches for the different values of these parameters.

1) *Influence of s_{max}* : Figure 7 depicts the worst-case end-to-end delays of v_1 obtained by both trajectory and network calculus approaches for s_{max} of v_1 ranging between 100B and 1500B. We observe that the Trajectory bound is slightly tighter than the network calculus bound as long as the packet size of VL v_1 stays greater or equal to the size of the packets of all other VLs (i.e. $4000b = 500B$). The two slopes intersect around 500B. Then, the distance between the Trajectory bound and the network calculus one increases when s_{max} decreases.

This confirms partly what was felt in section III-A2, i.e. the trajectory approach performs better for greater s_{max} values. However, this has nothing to do with the global load of the configuration. In fact, this is due to an assumption in the Trajectory approach theory. The Trajectory approach counts the packets in the busy periods of v_1 in every crossed node (end-system or switch) of the path. But, the last packet of a busy period corresponds under the Trajectory approach assumptions to the first packet of the busy period in the following node. In our case (and most of the time for simple configurations), this packet belongs to the flow under study

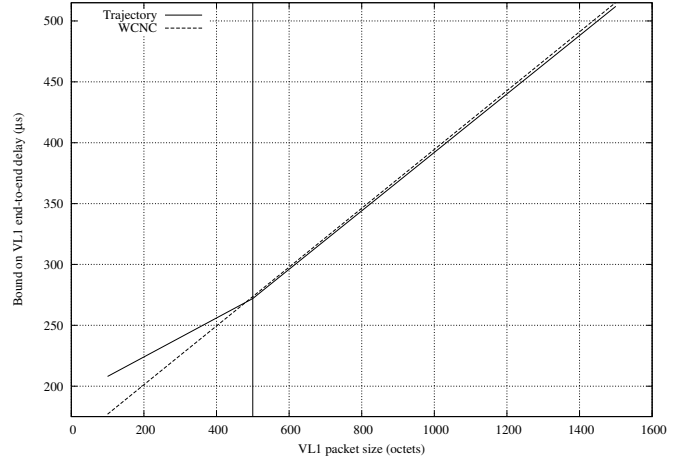


Fig. 7. Effect of s_{max} variation for VL v_1 on end-to-end delay bounds

(packet 1 from v_1). However, it can belong to another flow for more complex configurations. Consequently, the size of this particular packet, that has to be counted twice, is upper bounded by the size of the biggest packet of a VL meeting v_1 in that node (see inequation (2) of [13]).

Then, the pessimism of the trajectory approach increases with the difference between the size of the packet under study and the size of the biggest packet it meets. Higher differences are more likely to occur with smaller values of s_{max} .

2) *Influence of BAG* : We still consider the sample configuration depicted in Figure 2. Figure 8 depicts the worst-case end-to-end delays of v_1 obtained by both trajectory and network calculus approaches for BAG of v_1 ranging between 1 ms and 128 ms. Clearly, on this example, the value of the

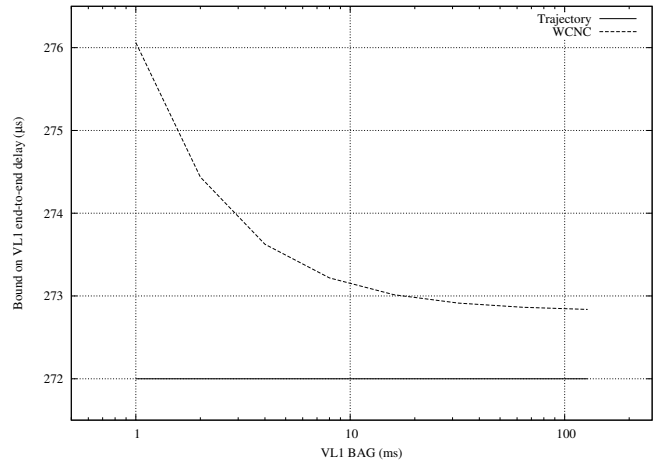


Fig. 8. Effect of BAG variation for VL v_1 on end-to-end delay bounds

BAG of v_1 has no influence on the bound obtained by the trajectory approach. Conversely, the network calculus end-to-end delay bound increases with smaller values of BAG for v_1 . this confirms the conclusion of section III-A1. With smaller BAG values, the load of the network is in fact increasing. This is because when a service curve of a flow is propagated

through a network, this curve is increased not only by the burst of the other packets, but also by the constant long-term rate $\frac{s_{max}}{BAG}$ (see [7]).

3) *Combined influence of BAG and s_{max}* : Figure 9 gives a more complete view of the influence of BAG and s_{max} variations on end-to-end delay bounds for VL $v1$. It depicts the difference in μs between the network calculus upper bound and the Trajectory upper bound. Positive values mean that the Trajectory bound is the tightest and negative values that the network calculus bound is the tightest.

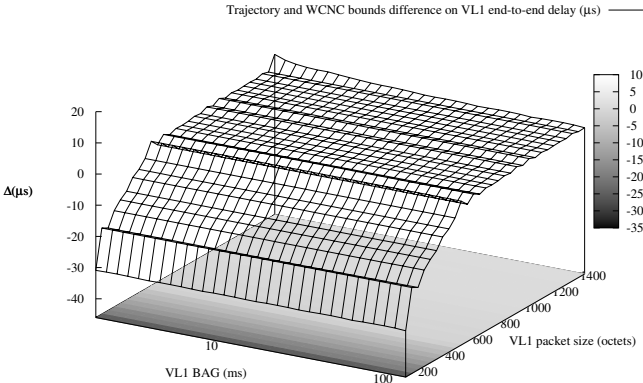


Fig. 9. Effect of BAG and s_{max} variation for VL $v1$ on end-to-end delay bounds

4) *Summary*: On this small sample configuration, we observe the same trends as on the complete industrial configuration described in Section III-A. The Trajectory approach outperforms the network calculus in most cases. Trajectories obtain their best results when the network load increases (with larger packet sizes and lower BAG values).

The drawback is for VLs with little packet size for which the pessimism of the Trajectory approach steps up, due to an overestimation in the worst-case response-time calculation.

IV. CONCLUSION

In this paper, we compare two approaches for bounding end-to-end delays on an industrial AFDX network. In this context, we show that, in most cases, the trajectory approach gives tighter results than the network calculus one. Moreover, we analyze, first on the industrial configuration, second on smaller configurations, the cases where the trajectory approach is outperformed by network calculus. This analysis emphasizes the fact that the minimum interframe delay and the frame length directly influence the performance of the two approaches.

A good solution is to combine the two approaches, keeping for each VL path the tightest computed upper bound.

The next step will be to try to adapt the trajectory approach, in order to improve the results in the cases where the bounds are worst than those obtained with the network calculus approach.

REFERENCES

- [1] ARINC 653, *Aeronautical Radio Inc. ARINC specification 653. Avionics application Software Standard Interface.*, 1997.
- [2] ARINC 429, *Aeronautical Radio Inc. ARINC specification 429. Digital Information Transfer System (DITS) parts 1,2,3.*, 2001.
- [3] "ARINC Specification 664: Aircraft Data Network, Parts 1,2,7," Aeronautical Radio Inc., Tech. Rep., 2002-2005.
- [4] J. Jasperneite, P. Neumann, M. Theis, and K. Watson, "Deterministic Real-Time Communication with Switched Ethernet," in *Proceedings of the 4th IEEE International Workshop on Factory Communication Systems*. Västerås, Sweden: IEEE Press, August 2002, pp. 11–18.
- [5] H. Charara, J.-L. Scharbag, J. Ermont, and C. Fraboul, "Methods for bounding end-to-end delays on an AFDX network," in *Proceedings of the 18th ECRTS*, Dresden, Germany, July 2006.
- [6] H. Bauer, J.-L. Scharbag, and C. Fraboul, "Applying and optimizing trajectory approach for performance evaluation of AFDX avionics network," in *Proc. of the 14th International Conference on Emerging Technologies and Factory Automation*. Mallorca, Spain: IEEE, September 2009.
- [7] J.-Y. Le Boudec and P. Thiran, *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*, ser. Lecture Notes in Computer Science. Springer-Verlag, 2001, vol. 2050, ISBN: 3-540-42184-X.
- [8] C.-S. Chang, *Performance Guarantees in Communication Networks*. London, UK: Springer-Verlag, 2000, ISBN = {1852332263}.
- [9] F. Frances, C. Fraboul, and J. Grieu, "Using network calculus to optimize the AFDX network," in *Proceedings of ERTS*, Toulouse, France, 2006.
- [10] J. Grieu, F. Frances, and C. Fraboul, "Preuve de déterminisme d'un réseau embarqué avionique," in *Actes du Colloque Francophone sur l'Ingénierie des Protocoles*, Paris, 7-10 Octobre 2003.
- [11] J. Grieu, "Analyse et évaluation de techniques de commutation Ethernet pour l'interconnexion des systèmes avioniques," Ph.D. dissertation, INP-ENSEEIH, France, September 2004.
- [12] K. Tindell and J. Clark, "Holistic schedulability analysis for distributed hard real-time systems," *Microprocess. Microprogram.*, vol. 40, no. 2-3, pp. 117–134, 1994.
- [13] S. Martin and P. Minet, "Schedulability analysis of flows scheduled with fifo: application to the expedited forwarding class," *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, pp. 8 pp.–, April 2006.