Network Calculus-based Timing Analysis of AFDX networks with Strict Priority and TSN/BLS Shapers

Anaïs Finzi, Ahlem Mifdaoui, Fabrice Frances, Emmanuel Lochin

Network Calculus-based Timing Analysis of AFDX networks with Strict Priority and TSN/BLS Shapers

A. FINZI, A. MIFDAOUI, F. FRANCES, E. LOCHIN
University of Toulouse-ISAE, France

***Abstract*—A homogeneous avionic communication architecture based on the AFDX supporting mixed-criticality applications will bring significant advantages, i.e., easier maintenance and reduced costs. To cope with this emerging issue, the AFDX may integrate multiple traffic classes: Safety-Critical Traffic (SCT) with hard real-time constraints, Rate-Constrained (RC) traffic requiring bounded latencies and Best Effort (BE) traffic with no delivery constraints. These traffic classes are managed based on a Non-Preemptive Strict Priority (NP-SP) Scheduler, where the highest priority traffic (SCT) is shaped with a Burst Limiting Shaper (BLS). The latter has been defined by the Time Sensitive Networking (TSN) task group to limit the impact of high priority flows on lower priority ones. This paper proposes a Network Calculus-based approach to compute the end-to-end delay bounds of SCT and RC classes. We consider the impact of the BLS and the multi-hop network architecture. We also provide proofs of service curves guaranteed to SCT and RC classes, needed to derive delay bounds with Network Calculus. The proposed approach is evaluated on a realistic AFDX configuration. Results show the efficiency of incorporating the TSN/BLS on top of a NP-SP scheduler in the AFDX to noticeably enhance the RC delay bounds while guaranteeing the SCT deadline, in comparison to an AFDX implementing only a NP-SP scheduler.**

***Index Terms*—TSN, BLS, NP-SP scheduler, Network Calculus, AFDX, mixed-criticality, avionics.**

I. Introduction

The growing number of interconnected end-systems and the expansion of exchanged data in avionics have led to an increase in complexity of the communication architecture. To cope with this trend, a first communication solution based on a high rate backbone network, i.e., the AFDX (Avionics Full Duplex Switched Ethernet) [1], has been implemented by Airbus in the A380, to interconnect critical subsystems. Moreover, some low rate data buses, e.g., CAN [9], are still used to handle some specific avionics domains, such as the I/O process and the Flight Control Management. Although this architecture reduces the time to market, it conjointly leads to inherent heterogeneity and new challenges to guarantee the real-time requirements.

To cope with these emerging issues, with the maturity and reliability progress of the AFDX after a decade of successful use, a homogeneous avionic communication architecture based on such a technology to interconnect different avionics domains may bring significant advantages, such as easier installation and maintenance and reduced weight and costs. This homogeneous communication architecture, based on the AFDX technology, needs to support mixed-criticality applications, where safety-critical and best effort traffic co-exist. Hence, in addition to the current AFDX traffic profile, called Rate Constrained (RC) traffic, at least two extra profiles have to be handled. The first, denoted by Safety-Critical Traffic (SCT), is specified to support flows with hard real-time constraints and the highest criticality, e.g., flight control data; whereas the second is for Best-Effort (BE) flows with no delivery constraint and the lowest criticality, e.g., In-Flight Entertainment traffic.

Various solutions have been proposed in the literature to support mixed-criticality applications in embedded systems and particularly in avionics [13]. However, most of these existing solutions are based on time-triggered communication schemes, which present some limitations compared to the event-triggered AFDX standard in terms of system modularity and reconfigurability.

Therefore, in [3], the assessment of the most relevant existing solutions enabling mixed-criticality on the AFDX vs avionics requirements has been conducted. The Burst-Limiting Shaper (BLS) [4] (defined in the Time Sensitive Networking (TSN) task group [15]) on top of Non-Preemptive Strict-Priority (NP-SP) scheduler has been selected as the most promising solution favoring the main avionics requirements, i.e., predictability, complexity and fairness. Preliminary performance evaluation of such a solution has been provided based on simulations. The first results were encouraging to pursue this line through providing in this paper formal timing analysis to prove certification requirements, a key point in avionics.

There are some interesting approaches in the literature concerning the formal timing analysis of TSN network, and more particularly BLS shaper. The first and seminal one in [10] introduces a first service curve model to induce worst-case delay computation. However, this presentation published by the TSN task group has never been extended in a formal paper. The second one has detailed a more formal worst-case timing analysis in [11]. The proposed model does not take into account the impact of either the same priority flows or the higher ones, which will clearly induce optimistic worst-case delays. The last and more recent one in [14] has proposed a formal analysis of TSN/BLS shaper, based on a Compositional Performance Analysis (CPA) method. This approach has handled the main limitations of the model presented in [11]; and interesting results for an automotive case study have been detailed. The impact of BLS on the highest priority traffic has been showed to deteriorate its timing performance, in comparison with a classic NP-SP scheduler.

In this paper, our main objective is conducting the worst-case timing analysis of our proposal, denoted as extended AFDX, incorporating BLS on top of NP-SP scheduler in

the AFDX, to guarantee the highest priority traffic (SCT) deadline while limiting its impact on the medium one (RC). Our apporach is based on the Network Calculus framework, which has been proved as highly modular and scalable, in comparison with CPA [12], and very effective to prove the certification requirements of avionics applications [5]. Several existing works have used Network Calculus to analyse the timing performance of Switched Ethernet and AFDX [5] [8] [7]. However, to the best of our knowledge, the issue of analyzing the timing performance of TSN/BLS on top of a NP-SP scheduler in avionics domain has not been handled yet in the literature.

Hence, our main contributions in this paper are :

- a Network Calculus-based approach to compute the delay bounds of SCT and RC classes in an extended AFDX network, taking into account the impact of the TSN/BLS and the multi-hop network architecture;
- providing proofs of service curves guaranteed to SCT and RC to compute the delay bounds as defined in Network Calculus;
- performance evaluation of our proposal on a realistic AFDX configuration and comparison of its efficiency with the current AFDX (implementing only NP-SP scheduler).

The rest of the paper is organized as follows. Section II introduces the main features of the extended AFDX incorporating the TSN/BLS on top of NP-SP scheduler to support the multiple traffic classes, i.e., SCT, RC and BE. Section III presents the Network Calculus-based timing analysis methodology, followed to compute the delay bounds of SCT and RC classes. Section IV details the guaranteed service curves offered to SCT and RC classes and their proofs to provide the delay bounds as defined in Network Calculus. Section V evaluates the proposed approach on a realistic avionic configuration. Finally, Section VI concludes the paper and gives some future work.

II. Extended AFDX with TSN/BLS

In this section, we first describe the main features of extended AFDX switch architecture, implementing the TSN/BLS on top of a NP-SP scheduler. More details on this extension can be found in [3]. Then, we detail the BLS behavior and its main parameters.

A. The extended AFDX Switch

The aim of extending the AFDX switch architecture with the TSN/BLS is to handle mixed criticality data, and more specifically three AFDX traffic profiles, as illustrated in Fig.1: (i) the SCT with its priority set by the BLS and the tightest temporal deadline, e.g., Flight-control flows; (ii) the RC with the medium priority and a deadline constraint to guarantee, e.g., current AFDX flows; (iii) the BE with the lowest priority and no time constraint, e.g., In-Flight Entertainment.

The current AFDX standard manages the exchanged data through the Virtual Link (VL) concept. This concept provides a way to reserve a guaranteed bandwidth for each traffic flow. The VL represents a multicast communication, which

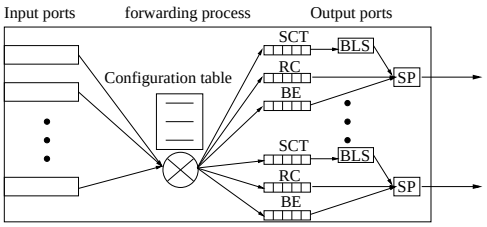

Fig. 1: An extended AFDX switch architecture

originates at a single End System and delivers packets to a fixed set of End Systems. Each VL is characterized by: (i) BAG (Bandwidth Allocation Gap), ranging in powers of 2 from 1 to 128 milliseconds, which represents the minimal inter-arrival time between two consecutive frames; (ii) MFS (Maximal Frame Size), ranging from 64 to 1518 bytes, which represents the size of the largest frame sent during each BAG. All these characteristics still are the same under the extended AFDX.

In Fig.1, we illustrate the architecture of the extended AFDX switch. It consists of: (i) store and forward input ports to verify each frame correctness before sending it to the corresponding output port; (ii) a static configuration table to forward the received frames to the correct output port(s) based on their VL identifier; (iii) the output ports with three priority queues, multiplexed with a NP-SP scheduler, and the highest one is shaped with the BLS.

B. BLS Shaper

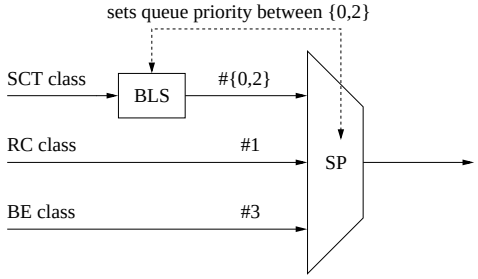

Fig. 2: An extended AFDX switch output port multiplexer architecture

The BLS belongs to the credit-based shapers class and it is generally used on top of Non-Preemptive Static Priority (NP-SP) scheduler as shown in Fig.2. It has been defined in [4] by an upper threshold L_M, a lower threshold L_R, such as $0 \leqslant L_R < L_M$, and a reserved bandwidth BW. Additionally, the priority of a queue q shaped by BLS, denoted $p(q)$, can vary between a high and a low value (with 0 the highest), denoted p_H and p_L. The low value is usually below the lowest priority of the unshaped traffic. In the avionic context, to guarantee the safety isolation level between the different traffic profiles, the low value associated to the SCT is set to be lower than the RC priority level, but higher than the BE priority. Therefore as shown in Fig.2, when considering one class for each traffic type, SCT queue priority oscillates between 0 (the highest) and 2, RC priority is 1 and BE has the priority 3 (the lowest). Thus, when SCT traffic is enqueued, BE traffic can never be sent no matter the state of BLS. In this case, RC is the only traffic that can be sent and this only happens when the SCT

priority is 2. As a consequence, BE traffic is isolated from SCT and RC traffics.

The credit counter varies as follows:
(i) initially, the credit counter starts at 0 and the queue of the burst limited flows is high;
(ii) the main feature of the BLS is the change of priority $p(k)$ of the shaped queue, which occurs in two contexts: 1) if $p(k)$ is high and credit reaches L_M; 2) if $p(k)$ is low and credit reaches L_R;
(iii) when a frame is transmitted, the credit increases (is consumed) with a rate of I_{send}, else the credit decreases (is gained) with a rate of I_{idle};
(iv) when the credit reaches L_M, it stays at this level until the end of the transmission of the current frame (if any);
(v) when the credit reaches 0 it stays at this level until the end of the transmission of the current frame (if any). The credit remains at 0 until a new BLS frame is transmitted.

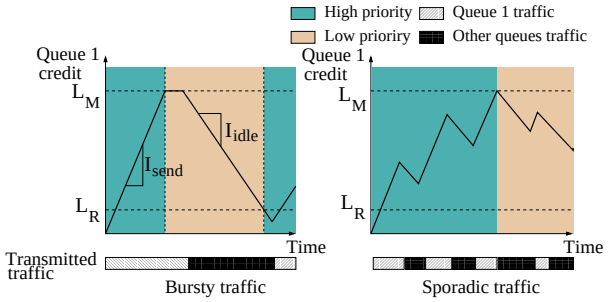

Fig. 3: BLS credit evolution

The behaviour of the BLS is illustrated in Fig. 3. As shown, the credit is always between 0 and L_M. The different parameters of the BLS shaper are defined as follows: (i) the decreasing rate is: $I_{idle} = BW \cdot C$, where C is the link speed and BW is the percentage of bandwidth reserved for BLS frames; (ii) the increasing rate is: $I_{send} = C - I_{idle}$.

It is worth noting that with the BLS, both the priority of the shaped queue and the state of all the queues, i.e., empty or not, define whether the credit is gained or lost. This aspect is depicted in Fig.3 for two arrival scenarios. The first one (left figure) shows the case of a bursty traffic, where the maximum of traffic shaped by the BLS is sent when its priority is the highest. Consequently, the other priorities send as much traffic as possible when the BLS queue priority has the low value. The second one (right figure) is for sporadic traffic, where we can see that when the shaped queue priority is highest but no frame is available, then the credit is regained. However, when the priority is at the low value and the other queues are empty, then shaped queue frames can be transmitted and the credit is consumed.

III. Timing Analysis Methodology

We present in this section the worst-case timing analysis methodology based on Network Calculus (NC), and followed to compute end-to-end delay bounds for SCT and RC classes in the extended AFDX network. We first present the Network Calculus framework and define the considered schedulability condition. Then, we detail the models of traffic flows, end-systems and switches. Finally, we explain the computation of the upper bounds on end-to-end delays. The main notations used in this paper are presented in Table I, where generally upper indices indicate nodes or components and lower indices indicate traffic classes or flows.

C	Link speed
MFS_k	Maximum Frame Size of flow k
BAG_k	Bandwidth Allocation Gap of flow k
J_k, Dl_k	Jitter and deadline of flow k
L_M, L_R	BLS maximum and resume credit levels
I_{idle}, I_{send}	BLS idle and sending slopes
BW	BLS reserved bandwidth
$p(j)$	Priority level of a class j with $p(j) \in \{0,1,2,3\}$
UR_j	The maximum network utilisation rate of a class j
$\beta_{j,k}^{n,i}$	Service curve guaranteed to the flow k of class j in the i^{th} node $n \in \{es, sw\}$ or component $n \in \{bls, sp\}$
β_j^n	Service curve guaranteed for the traffic class j in a node $n \in \{es, sw\}$ or component $n \in \{bls, sp\}$
$\beta^{sp}_{SCT_{p(SCT)}}$	Service curve guaranteed to the SCT when having the priority level p(SCT)
$\alpha_{j,k}^{n,i}$	Input arrival curve of the flow k of class j in the i^{th} node $n \in \{es, sw\}$ or component $n \in \{bls, sp\}$
α_j^n	Input arrival curve of the aggregate traffic of class j in a node $n \in \{es, sw\}$ or component $n \in \{bls, sp\}$
$\alpha_{j,k}^{*,n,i}$	Output arrival curve of the flow k of class j from the i^{th} node $n \in \{es, sw\}$ or component $n \in \{bls, sp\}$
$\alpha_j^{*,n}$	Output arrival curve of the aggregate traffic of class j from a node $n \in \{es, sw\}$ or a component $n \in \{bls, sp\}$
Δ_i^j	The defined BLS windows with $i \in \{send, idle\}$ and $j \in \{max, min\}$

TABLE I: Notations

A. Network Calculus Framework

The timing analysis detailed in this paper is based on Network Calculus theory [6] providing upper bounds on delays and backlogs. Delay bounds depend on the traffic arrival described by the so called *arrival curve* α, and on the availability of the traversed node described by the so called minimum *service curve* β. The definitions of these curves are explained as following.

Definition 1 (Arrival Curve)**.** *[6] A function $\alpha(t)$ is an arrival curve for a data flow with an input cumulative function $R(t)$,i.e., the number of bits received until time t, iff:*

$$\forall t, R(t) \leq R \otimes {}^1\alpha(t)$$

Definition 2 (Strict minimum service curve)**.** *[6] The function β is the minimum* strict *service curve for a data flow with an output cumulative function R^*, if for any backlogged period $]s,t]$[2], $\Delta R^*(t-s) \geq \beta(t-s)$.*

Definition 3 (Maximum service curve)**.** *[6] The function $\gamma(t)$ is the maximum service curve for a data flow with an input cumulative function $R(t)$ and output cumulative function $R^*(t)$ iff:*

$$\forall t, R^*(t) \leq R \otimes \gamma(t)$$

The traffic contracts are generally enforced using a leaky-bucket shaper, i.e., the traffic flow is (r,b)-constrained where r and b are the maximum rate and burst, respectively, and the arrival curve is $\alpha(t) = r \cdot t + b$ for $t > 0$. A common model of service curve is the rate-latency curve $\beta_{R,T}$, defined as

[1] $f \otimes g(t) = \inf_{0 \leq s \leq t}\{f(t-s) + g(s)\}$
[2] $]s,t]$ is called backlogged period if $R(\tau) - R^*(\tau) > 0, \forall \tau \in]s,t]$

$\beta_{R,T}(t) = [R(t-T)]^+$, where R for the transmission capacity, T for the system latency, and $[x]^+$ for the maximum between x and 0.

Then, we need the following results to compute the main performance metrics.

Theorem 1 (Performance Bounds)**.** *[6] Consider a flow F constrained by an arrival curve α crossing a system $\mathcal{S}$ that offers a minimum service curve β and a maximum service curve γ. The performance bounds obtained at any time t are:*
Backlog[3] *: $\forall\, t:\ q(t) \leq v(\alpha, \beta)$*
Delay[4]*: $\forall\, t:\ d(t) \leq h(\alpha, \beta)$*
Output arrival curve[5]*: $\alpha^*(t) = (\alpha \oslash \beta)(t)$*
Tight Output arrival curve: $\alpha^*(t) = ((\gamma \otimes \alpha) \oslash \beta)(t)$

Theorem 2 (Concatenation-Pay Bursts Only Once)**.** *[6] Assume a flow crossing two servers with respective service curves β_1 and β_2. The system composed of the concatenation of the two servers offers a service curve $\beta_1 \otimes \beta_2$.*

Corollary 1. *(Left-over service curve - NP-SP Multiplexing)[2] Consider a system with the strict service curve β and m flows crossing it, $f_1, f_2, .., f_m$. The maximum packet length of f_i is $l_{i,max}$ and f_i is α_i-constrained. The flows are scheduled by the NP-SP policy, where priority of f_i > priority of $f_j \Leftrightarrow i < j$. For each $i \in \{2, .., m\}$, the strict service curve of f_i is given by*[6]*:*

$$(\beta - \sum_{j<i} \alpha_j - \max_{k \geq i} l_{k,max})_{\uparrow}$$

B. Sufficient Schedulability Condition

To infer the real-time guarantees of our proposed solution to SCT and RC classes, we define a sufficient schedulability condition, which consists in verifying that the end-to-end delay bound of each traffic flow is lower than its deadline.

The end-to-end delay expression of a flow k in the class $j \in \{SCT, RC\}$, $EED_{j,k}$, along its path $path_k$ is as follows:

$$EED_{j,k} = d_j^{es} + d_{prop} + \sum_{i \in path_k} d_{j,k}^{sw,i} \qquad (1)$$

With d_j^{es} the delay within the end-system (es) to transmit the aggregate traffic of class j and d_{prop} the propagation delay along the path, which is generally negligible in an avionics network. The last delay $d_{j,k}^{sw,i}$ represents the delay within the i^{th} switch (sw) along the flow path and it consists of several parts: (i) the store and forward delay at the input port, equal to the transmission time of a maximum-sized frame; (ii) the technological latency due to the switching process, upper-bounded by 1μs; (iii) the output port multiplexer delay due to the BLS (bls) and NP-SP (sp) scheduler. To enable the computation of upper bounds on these delays, we need to model the different parts of the network, and more particularly the BLS.

[3] v: maximal vertical distance
[4] h: maximal horizontal distance
[5] $f \oslash g(t) = \sup_{s \geq 0}\{f(t+s) - g(s)\}$
[6] $g_{\uparrow}(t) = \max\{0, \sup_{0 \leq s \leq t} g(s)\}$

C. System Modeling

To compute upper bounds on end-to-end delays of different traffic classes using Network Calculus, we need to model each message flow to compute its maximum arrival curve, and the behavior of end-systems and the crossed switches to compute the minimum service curves.

The characteristics of each traffic flow k of class $j \in \{SCT, RC, BE\}$, generated by an end-system, is characterized by $(BAG_k, MFS_k, Dl_k, J_k)$ for respectively the minimum inter-arrival time, the maximum frame size integrating the protocol overhead, the deadline if any (generally equal to BAG_k unless explicitly specified and infinite for BE) and the jitter.

The arrival curve of each flow k in class j at the input of the i^{th} node $n \in \{es, sw\}$ or a component $n \in \{bls, sp\}$ along its path is a leaky-bucket curve with a burst $b_{j,k}^{n,i}$ and a rate r_k:

$$\alpha_{j,k}^{n,i}(t) = b_{j,k}^{n,i} + r_k \cdot t$$

For instance, the input arrival curve of flow k in class j at the end-system is: $\alpha_{j,k}^{es}(t) = MFS_k + \frac{MFS_k}{BAG_k} \cdot (t + J_k)$.

Therefore, the arrival curve of the aggregate traffic in class j at the input of the i^{th} node $n \in \{es, sw\}$ or a component $n \in \{bls, sp\}$ is: $\alpha_j^{n,i}(t) = \sum_{k \in j} \alpha_{j,k}^{n,i}(t)$. For instance,

$$\alpha_j^{es}(t) = b_j + r_j t \text{ with } \begin{cases} b_j = \sum\limits_{k \in j} MFS_k + \frac{MFS_k}{BAG_k} J_k \\ r_j = \sum\limits_{k \in j} \frac{MFS_k}{BAG_k} \end{cases}$$

For the end-systems, they are implementing a Non-Preemptive Static Priority Scheduler (NP-SP). This scheduler has been already modeled in the literature [2] through Cor. 1, and the defined strict minimum service curve guaranteed to a traffic class j within an end-system es is as follows:

$$\beta_j^{es}(t) = \left[C \cdot t - \sum_{k \in i, p(i) < p(j)} \alpha_{i,k}^{es}(t) - \max_{k \in i, p(i) \geq p(j)} MFS_k \right]_{\uparrow}$$

For the proposed extended AFDX switches, we need to model the impact of the BLS implemented on top of the NP-SP scheduler on the SCT and RC classes.

For SCT class, we distinguish two possible scenarios:
(i) the first one covers the particular case where the priority of SCT remains low (2), i.e., the other queues are empty. The minimum service curve guaranteed within the switch in this case is due to the NP-SP (sp) scheduler and denoted $\beta_{SCT_2}^{sp}$. It is computed via Cor. 1 when considering the impact of RC traffic as the highest priority and the BE as the lowest priority;
(ii) the second one covers the general case where the priority of SCT oscillates between low (2) and high (0), as explained in Section II-B. The minimum service curve guaranteed within the switch in this case is computed via Th. 2. It is the concatenation of the service curves within the bls component β_{SCT}^{bls} (computed in Section IV) and the sp component $\beta_{SCT_0}^{sp}$ (computed via Cor. 1 similarly to β_{SCT}^{es} when SCT has the highest priority 0).

Therefore, for the traffic class SCT, we define the following relation between the service curves guaranteed within the

switch sw (β^{sw}_{SCT}) and the components $\{bls, sp\}$ (respectively β^{bls}_{SCT} and β^{sp}_{SCT}):

$$\beta^{sw}_{SCT}(t) = \max(\beta^{sp}_{SCT_2}, \beta^{sp}_{SCT_0} \otimes \beta^{bls}_{SCT}(t)) \quad (2)$$

For RC class, we need to model the minimum service curve guaranteed within the NP-SP sp scheduler using Cor. 1, when considering the maximum output arrival curve of SCT from the bls component $\alpha^{*,bls}_{SCT}$ (as shown in Fig. 2):

$$\beta^{sw}_{RC}(t) = \left[C \cdot t - \alpha^{*,bls}_{SCT}(t) - \max_{k \in i, p(i) \geq p(RC)} MFS_k \right]_{\uparrow} \quad (3)$$

In Section IV, we will detail the minimum service curve guaranteed within the BLS β^{bls}_{SCT} and the maximum output arrival curve $\alpha^{*,bls}_{SCT}$ (the tight output arrival curve in Th. 1). The latter depends on the respective maximum service curve γ^{bls}_{SCT}, which is also detailed in Section IV.

D. Computing End-to-End Delays

The computation of the end-to-end delay upper bounds for each flow k in class j follows four main steps:

(1) Computing the strict minimum service curve guaranteed to each traffic class j in each node i of type $n \in \{es, sw\}$, $\beta^{n,i}_j$. This will infer the computation of the residual service curve, guaranteed to each individual flow k in class j, $\beta^{n,i}_{j,k}$, using Cor. 1;

(2) Knowing the residual service curve guaranteed to each flow within each crossed node allows the propagation of the arrival curves along the flow path, using Th.1. We can compute the output arrival curve of each node, which will be in its turn the input of the next node;

(3) The computation of the minimum end-to-end service curve of each flow k in class j, based on Th.2, is simply the concatenation of its residual service curve within each crossed node i of type n along the path $path_k$, $\beta^{n,i}_{j,k}$, computed in step (2);

(4) Given the minimum end-to-end service curve of each flow k in class j along its $path_k$ and its maximum arrival curve at the initial source, the end-to-end delay upper bound $EED_{j,k}$ is the maximum horizontal distance between both curves (using Th.1).

Hence, as we can notice, we need to model all the unknown service curves, related to the BLS, to enable the end-to-end delay upper bounds computation. These curves are detailed in the next section. It is worth noting that since the BE class has no deadline, the computation of the respective upper bounds on end-to-end delays are not detailed in this paper.

IV. Service Curves offered by TSN/BLS

To compute the guaranteed service curves by the BLS to SCT and RC classes, we need to detail two types of windows, which are enforced by the BLS behavior. The first one is denoted as *sending window*, during which the SCT has the highest priority and is sent until the consumed credit reaches the maximum threshold, L_M. The second one is called *idle window* where the SCT has the priority just higher than BE and the consumed credit is decreasing until reaching the minimum threshold, L_R. Moreover, due to the non-preemptive message transmission, both windows have minimal and maximal durations, as illustrated in Fig. 4. The various combinations of such durations will induce the different service curves, which are necessary for computing upper bounds on end-to-end delays and detailed in this section.

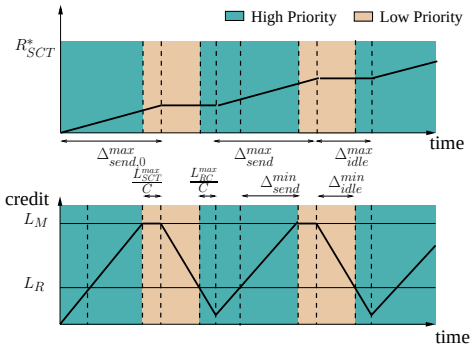

Fig. 4: An example of BLS credit evolution and sending and idle windows

A. Service Curves of SCT

The strict minimum service curve of SCT, β^{bls}_{SCT}, defines a lower bound on the SCT output cumulative traffic from the BLS. This curve represents the most deteriorated behaviour of BLS, in terms of offered service to the SCT, which maximizes its delay within the BLS. Hence, to cover this worst-case behaviour, we combine the maximum *idle window* and the minimum *sending window* durations.

The minimum *sending window* duration, Δ^{min}_{send}, is the time for the consumed credit to go from the lowest to the highest thresholds (from L_R to L_M) with an increasing slope I_{send}:

$$\Delta^{min}_{send} = \frac{L_M - L_R}{I_{send}} \quad (4)$$

The maximum *idle window* duration, Δ^{max}_{idle}, is the time for the consumed credit to go from L_M to L_R with a decreasing slope I_{idle}, in addition to the transmission time of a maximum frame of the RC traffic. The latter is due to the non-preemption feature when a RC frame is starting its transmission just before the consumed credit reaches the lowest threshold (L_R). It is worth noting that the BE class impacts the SCT only within the NP-SP scheduler and not within the BLS since it has a priority (3) lower than the lowest priority of SCT (2):

$$\Delta^{max}_{idle} = \frac{L_M - L_R}{I_{idle}} + \frac{MFS_{RC}}{C} \quad (5)$$

Therefore, the strict minimum service curve guaranteed to the SCT, β^{bls}_{SCT}, is defined in Th. 3 and the proof is detailed in Appendix VII-A.

Theorem 3 (Strict Minimum Service Curve of SCT in BLS)**.** *Consider a SCT crossing a server with a constant rate C, implementing a BLS shaper. The strict minimum service curve guaranteed to the SCT is as follows:*

$$\beta^{bls}_{SCT}(t) = \frac{\Delta^{min}_{send}}{\Delta^{min}_{send} + \Delta^{max}_{idle}} \cdot C \cdot (t - \Delta^{max}_{idle})^+ \quad (6)$$

where $[x]^+$ is the maximum between x and 0 and (Δ^{min}_{send}, Δ^{max}_{idle}) are defined in Eqs. (4) and (5), respectively.

The maximum service curve of SCT, γ^{bls}_{SCT}, represents the best offered service to the SCT, which induces the minimum processing delay within the BLS. As such, in the presence of RC traffic, we combine the minimum *idle window* duration and the maximum *sending window* one to handle this best-case behavior.

The maximum *sending window* duration, Δ^{max}_{send}, is equal to the sum of : (i) the minimum *sending window* duration, Δ^{min}_{send}; (ii) the transmission time of a maximum frame of the SCT due to the non-preemption feature, i.e., one SCT frame may start its transmission just before the consumed credit reaches L_M; (iii) the time to consume the gained credit during the transmission of one additional maximum frame of RC traffic at the end of the *idle window*. The latter parameter is due to the fact that the resume level of BLS, L_R, is the lower threshold on the consumed credit to trigger the priority change of the SCT from lowest to highest, and not an extreme value for the consumed credit itself. Actually, if a frame of RC traffic has been transmitted just at the end of the *idle window*, the consumed credit keeps decreasing until it either reaches 0, or the transmission ends. Therefore, the lowest value the consumed credit can reach due to the non-preemption feature is $\max(0, L_R - \frac{MFS_{RC}}{C}.I_{idle})$. The additional time during which the consumed credit can then increase with a slop I_{send} is $\frac{L_R - \max(0, L_R - \frac{MFS_{RC}}{C}.I_{idle})}{I_{send}}$.

The maximum *sending window* duration is then as follows:

$$\begin{aligned}\Delta^{max}_{send} &= \frac{L_M - L_R}{I_{send}} + \frac{MFS_{SCT}}{C} \\ &+ \min(\frac{MFS_{RC}}{C} \cdot \frac{I_{idle}}{I_{send}}, \frac{L_R}{I_{send}}) \end{aligned} \quad (7)$$

However, it is worth noting that the consumed credit may start at 0, such as at the initialisation phase or after a long period of inactivity. Hence, the maximum *sending window* duration covering such possibility, $\Delta^{max}_{send,0}$, is as follows:

$$\Delta^{max}_{send,0} = \frac{L_M}{I_{send}} + \frac{MFS_{SCT}}{C} \quad (8)$$

The minimum *idle window* duration, Δ^{min}_{idle}, is simply the time it takes for the consumed credit to go from L_M to L_R with a decreasing slope of I_{idle}:

$$\Delta^{min}_{idle} = \frac{L_M - L_R}{I_{idle}} \quad (9)$$

Therefore, the maximum service curve guaranteed to the SCT, γ^{bls}_{SCT} is defined in Th. 4 and the proof is detailed in Appendix VII-B.

Theorem 4 (Maximum Service Curve of SCT in BLS)**.** *Consider a SCT crossing a server with a constant rate C, implementing a BLS shaper. The maximum service curve guaranteed to the SCT is as follows.*

$$\gamma^{bls}_{SCT}(t) = \begin{cases} \textit{if no RC traffic: } C \cdot t \\ \textit{Otherwise:} \\ \frac{\Delta^{max}_{send}}{\Delta^{nom}_{\gamma_{SCT}}} \cdot C \cdot t + \Delta^{max}_{send,0} \cdot C \cdot \frac{\Delta^{min}_{idle}}{\Delta^{nom}_{\gamma_{SCT}}} \end{cases}$$

with $\Delta^{nom}_{\gamma_{SCT}} = \Delta^{max}_{send} + \Delta^{min}_{idle}$.

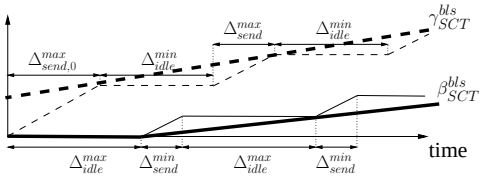

Fig. 5: Minimum and maximum service curves for SCT traffic

The minimum and maximum service curves of SCT defined in Th. 3 and Th. 4, respectively, are illustrated in Fig. 5.

B. Service Curves of RC

To compute the minimum service curve of RC, β^{sw}_{RC} defined in Eq. (3), we need to compute the maximum output arrival curve of SCT from the BLS, $\alpha^{*,bls}_{SCT}$, detailed in the following corollary.

Corollary 2 (Maximum Output Arrival Curve of SCT from BLS)**.** *Consider a SCT with a maximum leaky-bucket arrival curve α^{bls}_{SCT} at the input of a BLS shaper, guaranteeing a minimum rate-latency service curve β^{bls}_{SCT} and a maximum service curve γ^{bls}_{SCT}. The maximum output arrival curve is:*

$$\alpha^{*,bls}_{SCT}(t) = \min(\gamma^{bls}_{SCT}(t), \alpha^{bls}_{SCT} \oslash \beta^{bls}_{SCT}(t)) \quad (10)$$

Proof. To prove Cor. 2, we generalize herein the rule 13 in p. 123 in [6], i.e., $(f \otimes g) \oslash g \leq f \otimes (g \oslash g)$, to the case of three functions f, g and h when $g \oslash h \in \mathcal{F}$, where $\mathcal{F}$ is the set of non negative and wide sense increasing functions:

$$\mathcal{F} = \{f : \mathbb{R}^+ \to \mathbb{R}^+ \mid f(0) = 0, \forall t \geq s : f(t) \geq f(s)\}$$

According to Th. 1, we have $\alpha^*(t) = (\gamma^{bls}_{SCT} \otimes \alpha) \oslash \beta^{bls}_{SCT}$. Moreover, in the particular case of a leaky-bucket arrival curve α and a rate-latency service curve β^{bls}_{SCT}, $\alpha \oslash \beta^{bls}_{SCT}$ is a leaky-bucket curve, which is in $\mathcal{F}$. Hence, we have the necessary condition to prove the following:

$$(\alpha \otimes \gamma) \oslash \beta(t) \leq \gamma \otimes (\alpha \oslash \beta)(t) \leq \min(\gamma(t), \alpha \oslash \beta(t))$$

□

Theorem 5 (Minimum Strict Service Curves of RC)**.** *Consider a SCT with a maximum leaky-bucket arrival curve α at the input of a server with a constant rate C implementing a BLS shaper, guaranteeing a minimum rate-latency service curve β^{bls}_{SCT} (defined in Th. 3) and a maximum service curve γ^{bls}_{SCT} (defined in Th. 4). The minimum strict service curve guaranteed to RC traffic in the NP-SP scheduler, integrating the impact of the BLS, is as follows:*

$$\beta^{sw}_{RC}(t) = [\max(\beta^{sp}_{RC}(t), \beta^{bls}_{RC}(t)) - \max_{k \in i, p(i) \geq p(RC)} MFS_k]_\uparrow$$

where:

$$\begin{aligned} &\beta^{sp}_{RC}(t) = (C \cdot t - \alpha^{bls}_{SCT} \oslash \beta^{bls}_{SCT}(t))^+ \\ &\beta^{bls}_{RC}(t) = \frac{\Delta^{min}_{idle}}{\Delta^{nom}_{\gamma_{SCT}}} \cdot C \cdot \left(t - \Delta^{max}_{send,0}\right)^+ \\ &\Delta^{nom}_{\gamma_{SCT}} = \Delta^{max}_{send} + \Delta^{min}_{idle} \\ &\Delta^{max}_{send} \textit{ and } \Delta^{min}_{idle} \textit{ are defined in Eqs. (7) and (9)} \end{aligned} \quad (11)$$

Proof. According to Cor. 1, the residual minimum strict service curve guaranteed to RC traffic crossing a NP-SP scheduler is as follows:

$$\beta^{sw}_{RC}(t) = \left[C \cdot t - \alpha^{*,bls}_{SCT}(t) - \max_{k \in i, p(i) \geq p(RC)} MFS_k\right]_{\uparrow} \quad (12)$$

Moreover, according to Cor. 2, the maximum output arrival curve of SCT from the BLS, $\alpha^{*,bls}_{SCT}(t)$, is:

$$\alpha^{*,bls}_{SCT}(t) = \min(\gamma^{bls}_{SCT}(t), \alpha^{bls}_{SCT} \oslash \beta^{bls}_{SCT}(t)) \quad (13)$$

Using Eq. (12) and Eq. (13), we can deduce the following:

$$\begin{aligned}
&\beta^{sw}_{RC}(t) = [C \cdot t - \min(\alpha^{bls}_{SCT} \oslash \beta^{bls}_{SCT}(t), \gamma^{bls}_{SCT}(t)) \\
&- \max_{k \in i, p(i) \geq p(RC)} MFS_k]_{\uparrow} \\
&= [\max((C \cdot t - \alpha^{bls}_{SCT} \oslash \beta^{bls}_{SCT}(t))^{+}, (C \cdot t - \gamma^{bls}_{SCT}(t))^{+}) \\
&- \max_{k \in i, p(i) \geq p(RC)} MFS_k]_{\uparrow} \\
&= [\max((C \cdot t - \alpha^{bls}_{SCT} \oslash \beta^{bls}_{SCT}(t))^{+}, \\
&\frac{\Delta^{min}_{idle}}{\Delta^{nom}_{\gamma_{SCT}}} \cdot C \cdot \left(t - \Delta^{max}_{send,0}\right)^{+}) - \max_{k \in i, p(i) \geq p(RC)} MFS_k]_{\uparrow}
\end{aligned}$$

□

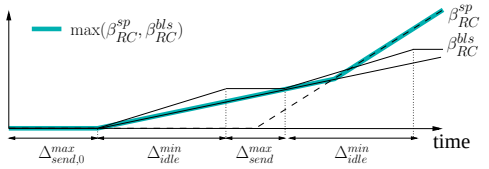

Fig. 6: RC traffic minimum service curve

The minimum service curve of RC defined in Th. 5 is illustrated in Fig. 6.

V. Performance Analysis

In this section, we conduct performance analysis of the extended AFDX (implementing TSN/BLS on top of 3-priority NP-SP scheduler) to evaluate its efficiency to support multiple traffic profiles, in comparison to the current AFDX solution (implementing 3-priority NP-SP scheduler). This evaluation is based on the worst-case timing analysis methodology and the various service curves detailed in Sections III and IV, respectively. First, we describe our realistic avionics case study. Afterwards, we assess the scalability of the extended AFDX to handle the current avionics traffic configuration, in comparison to the current AFDX. Finally, we analyse the impact of such an extended solution on SCT and RC delay bounds when considering future avionics traffic configurations, in comparison to the current AFDX.

A. Avionics Case Study

Our case study is a representative avionics communication architecture of the A380, based on a 1-Gigabit AFDX[7] backbone network, which consists of 4 switches and 64 end-systems as shown in Fig. 7 (a). The maximum utilisation rate on the 100Mbps current AFDX on board the A380 is 30%. Thus, on the 1Gigabit version, the maximum utilisation rate will be only of 3 %. However, there is only standard AFDX traffic (RC) circulating on this current communication architecture. Hence, to enable the performance analysis of the extended AFDX, we have extended this current traffic configuration, denoted herein as *reference configuration*, to support different traffic profiles generated by each end-system, which are described in Tab. II.

Each flow in each traffic class $j \in \{SCT, RC, BE\}$ is characterized by the same $(MFS_j, BAG_j, Dl_j, J_j)$ detailed in Section III-C. The SCT has a deadline of 2ms, and because of the BLS behavior it admits the highest priority 0 during the sending windows and the priority below RC (2) during the idle windows. Figure 7 (b) shows the traffic communication patterns between the source and the final destinations of a given flow. Each circulating traffic flow on the backbone network is a multicast flow with 16 destinations, and crosses two successive switches before reaching its final destinations. The first switch in the path receives traffic from 16 end-systems to forward it in a multicast way to its two neighboring switches. Afterwards, the second switch in the path, which receives traffic from the two predecessor switches, forwards the traffic in its turn to the final end-system.

The main considered performance metrics are:
(i) the maximum utilisation rate of each traffic class, that can be sent on the extended AFDX architecture while respecting the schedulability condition, described in Section III-B. This metric enables the scalability analysis of the extended AFDX, in comparison with the current one;
(ii) the delay bounds of SCT and RC classes to prove the predictability of the extended AFDX and analyse its impact on the system timing performance, in comparison with the current AFDX. It is worth noting that since the BE does not have a deadline, and its largest impact on the other priorities is the transmission time of a maximum sized frame, then the timing performance of this class is not detailed herein.

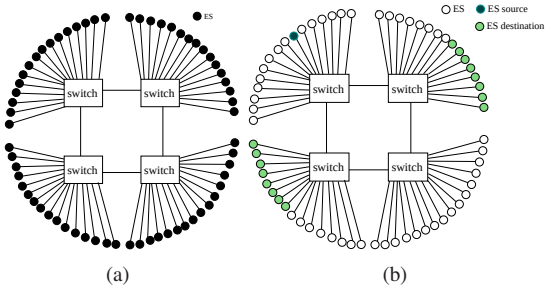

Fig. 7: Representative AFDX network: (a) Architecture; (b) Traffic Communication Patterns

Priority	Traffic Class	MFS (Bytes)	BAG (ms)	Deadline (ms)	Jitter (ms)
0/2	SCT	64	2	2	0
1	RC	320	2	2	0
3	BE	1024	8	none	0.5

TABLE II: Avionics flow Characteristics

B. Testing Scenarios

To compute both performance metrics, we consider four scenarios: scenarios 1 and 2 are for the scalability analysis;

[7]The 1-Gigabit version of the AFDX is under specification.

whereas scenarios 3 and 4 are for timing analysis.

To enable scenarios 1 and 2, we have started from the *reference configuration*, and then computed the maximum utilisation rate of SCT class that can be transmitted on the current AFDX, while respecting the schedulability condition in the presence of 3% of RC traffic. This computation has shown that the current AFDX can support up to a maximum utilisation rate for SCT of $UR_{SCT} = 28.7\%$. Hence, scenario 1 (resp. scenario 2) consists in starting from a traffic configuration characterized by ($UR_{SCT} = 28.7\%; UR_{RC} = 3\%$) circulating on the extended AFDX, then increasing the UR_{SCT} (resp. UR_{RC}) until finding the maximum value which still respects the schedulability condition. The aim of this scenario is to compare the scalability of the extended and current AFDX solutions, when increasing the congestion due to SCT (resp. RC) traffic. The results of both scenarios (1 and 2) are detailed in Section V-C.

Afterwards, to have an idea about the timing performance of future avionics configurations based on the 1 Gigabit AFDX technology, which may very probably support higher utilisation rate of SCT and RC traffic than the *reference configuration*, we consider scenarios 3 and 4 described in Table III. As it can be noticed, the principle of scenario 3 (resp. scenario 4) is to fix the utilisation rate of RC class (resp. SCT class) at 20% and vary the SCT (resp. RC) utilisation rate to assess the impact of increasing network congestion on the timing performance. The considered BLS parameters are the same for both scenarios: $BW = 0.46$ to support a maximum utilisation rate of SCT of $UR_{SCT} = 45\%$ (this is an upper bound for the estimated future needs in terms of SCT traffic); $L_R = 0$ and $L_M(bits) = 22,077$ to enable the transmission of a maximum SCT burst within the BLS of 80 frames during a minimum sending window, i.e., a generated burst of 5 SCT flows per End-System. Moreover, as it is illustrated in Table III, the variation of the utilisation rate of a class j is obtained through increasing the number of generated traffic flows within each end-system, n_j^{es}. Thus, the maximum utilisation rate is equal to $UR_j(\%) = \frac{C_j}{C}$ with C_j the capacity used in the bottleneck by the aggregate traffic of class $j \in \{RC, SCT\}$ and $C_j = 16 \cdot n_j^{es} \cdot \frac{MFS_j}{BAG_j}$. The BE traffic is only used to achieve full utilisation rate. The results of both scenarios (3 and 4) are detailed in Section V-D.

Scenarios	Scenario 3	Scenario 4
$(UR_{RC}; UR_{SCT})(\%)$	$(20; [1..45])$	$([1..80]; 20)$
$(n_{RC}^{es}; n_{SCT}^{es})$	$(10; [1:4:110])$	$([1:2:39]; 47)$
$(BW; L_M; L_R)$	$(0.46; 22077; 0)$	$(0.46; 22077; 0)$

TABLE III: Considered Test Scenarios 3 and 4

C. Scalability of the current avionics configuration

Config.	Ref.	Sc. 1	Sc. 2
$(UR_{SCT}; UR_{RC})$	$(28.7; 3)$	$(43; 3)$	$(28.7; 13)$
$(BW; L_M; L_R)$	N.A	$(0.90; 10240; 0)$	$(0.65; 35840; 0)$

TABLE IV: Results of Scenarios 1 and 2

The aim of this section is to analyse the scalability of the extended AFDX, in comparison with the current one. Hence, starting from the *reference configuration* characterized by ($UR_{RC} = 3\%$, $UR_{SCT} = 28.7\%$), we have tested for scenario 1 (resp. scenario 2) various BLS parameters to increase as much as possible the maximum SCT (resp. RC) utilisation rate. As shown in Table IV, there exists a BLS configuration for scenario 1 (resp. scenario 2) allowing to achieve a maximum utilisation rate of SCT (resp. RC) of $UR_{SCT} = 43\%$ (resp. of $UR_{RC} = 13\%$) under extended AFDX, instead of only $UR_{SCT} = 28.7\%$ ($UR_{RC} = 3\%$) under current AFDX.

These results show an enhancement of scalability (maximum utilisation rate) with the extended AFDX of 50% and 333% for the SCT and RC classes, respectively, in comparison with the current AFDX.

D. Timing performance of future avionics configurations

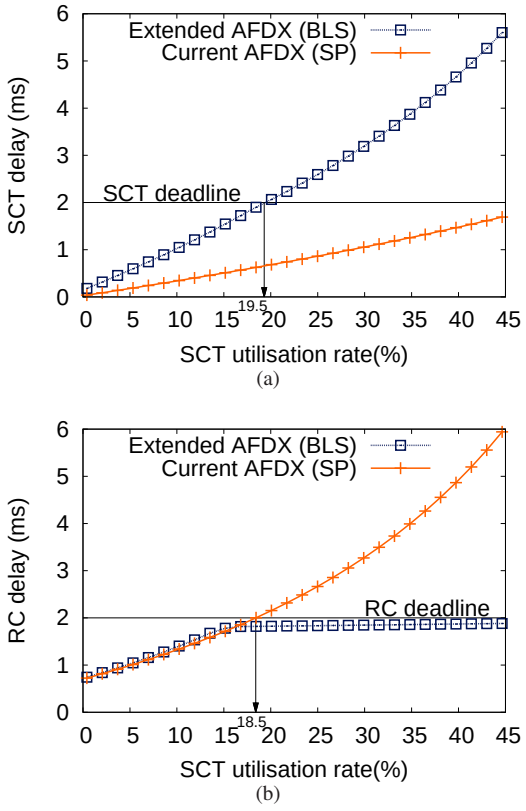

Fig. 8: Scenario 3: Impact of SCT max. utilisation rate on: (a) SCT delays; (b) RC delays

SCT timing performance

To analyse the timing performance of SCT when using the extended AFDX instead of the current one, we focus on Figures 8(a) and 9(a) showing the SCT delay bounds evolution, regarding the SCT and RC utilisation rate variation, respectively.

As shown in Fig. 8 (a), when increasing the utilisation rate of the SCT, the delay upper bounds are obviously increasing under both solutions, but are globally higher under the extended AFDX. This fact is due to the BLS behavior on top of the NP-SP scheduler implemented within the extended AFDX, which infers dividing the SCT burst to be sent within many *sending windows*; whereas the regular NP-SP scheduler implemented within the current AFDX is sending the SCT burst all at once.

On the other hand, as it can be noticed in Fig. 9 (a), when increasing the utilisation rate of the RC, the SCT delay bounds are constant under the current AFDX since SCT has the highest priority level and is at most delayed by a maximum sized frame of lower priorities, i.e., RC and BE; whereas they are increasing under the extended AFDX for a RC utilisation rate up to 20% and become equal to the SCT deadline (2ms) for a RC utilisation rate higher than 20%. The increase is due to the fact that the RC rate is not large enough to use all the bandwidth guaranteed by the BLS; thus the guaranteed SCT service within the BLS is limited by the left part of the Eq.(2). This shows the good isolation level, enforced by the BLS, between RC and SCT classes.

These results show the impact of the extended AFDX network on SCT when increasing the network congestion. The main interesting feature to highlight is its efficiency to guarantee a high isolation level between the SCT and RC traffic, which is one of the key requirements for avionics applications.

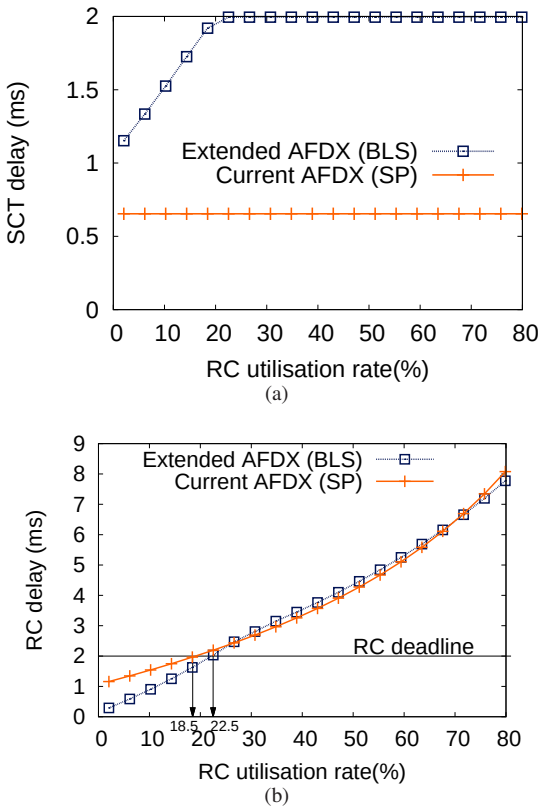

Fig. 9: Scenario 4: Impact of RC max. utilisation rate on: (a) SCT delays; (b) RC delays

RC timing performance

We detail herein the main interesting results concerning the impact of the extended AFDX on the RC timing performance, based on Figures 8(b) and 9(b).

Fig. 8 (b) illustrates the variation of the RC delay upper bounds in terms of the SCT utilisation rate. We can easily distinguish two phases on this figure. The first one is observed for a utilisation rate below 14%, where the delay upper bounds under both solutions are very similar. The second phase (when the utilisation rate is higher than 14%) shows that the delay upper bounds increase inherently under current AFDX, whereas they are constant under the extended AFDX.

These results are coherent with the guaranteed service to the RC traffic in Th. 5, which is the maximum between β_{RC}^{sp} and β_{RC}^{bls}. Hence, during the first phase, the service corresponds to β_{RC}^{sp}, which is impacted by the maximum arrival curve of the SCT; thus its maximum utilisation rate. This fact explains the delay bounds increase. Afterwards, the service becomes related to β_{RC}^{bls} during the second phase, which enforces a maximum constraint on the arrival curve of the SCT under the extended AFDX due to the BLS, γ_{SCT}^{bls}. This maximum constraint implies a constant delay under the extended AFDX. On the other hand, the service guaranteed under current AFDX is deeply related to the arrival curve of SCT, which explains the inherent delay bound increase.

Fig. 9 (b) shows the impact of the RC utilisation rate variation on the RC delay bounds. As it can be noticed, the RC delay bounds are increasing under both solutions, but still are better under extended AFDX. For instance, for a RC utilisation rate of 10%, we observe a delay bound of 1.5ms and 0.9ms under the current and extended AFDX, respectively; thus the enhancement of the delay bound is about 40% at $UR_{RC} = 10\%$, and it goes up to 74% at $UR_{RC} = 2\%$.

These results show the valuable impact of the extended AFDX on RC traffic, in comparison with the current AFDX solution. We can distinguish two interesting features: (i) the first one concerns the noticeable RC delay bounds decrease, where they become constant after a given SCT utilisation rate; (ii) the second one is the enhancement of the RC delay bound under the extended AFDX when varying the RC utilisation rate, which is up to 74%.

VI. Conclusions

In this paper, we have proposed a worst-case timing analysis using Network Calculus of an extended AFDX (implementing a BLS shaper on top of NP-SP scheduler). The conducted performance analysis on a realistic avionics case study highlights the benefit of using the extended AFDX, to isolate the highest priority traffic SCT and mitigate its impact on the RC one. Numerical results have shown noticeable enhancements in terms of delay upper bounds of the RC traffic (up to 74%) and maximum utilisation rate (up to 333% for RC and up to 50% for SCT), in comparison with the current AFDX network. As a next step, we will introduce a tuning method to find the best BLS parameters, which respect the highest priority traffic deadline, while decreasing as much as possible the RC delay bounds.

VII. Appendices

A. Proof of Theorem 3

Proof. Consider $R_{SCT}^{*}(t)$ the output cumulative function of the SCT at the output of the server implementing a BLS, and $\Delta R_{SCT}^{*}(\delta)$ the variation of the output cumulative function during δ. To prove that β_{SCT}^{bls} in Eq. (6) is a strict minimum service curve, we need to prove Def. 2 for any backlogged

period δ, i.e., the SCT flows are continuously backlogged during δ.

During a backlogged period δ, the SCT has at least p opportunities of full service constrained by $\beta(t) = C \cdot t$ during the minimum sending window Δ^{min}_{send}, then:

$$\Delta R^*_{SCT}(\delta) \geq p \cdot C \cdot \Delta^{min}_{send} \tag{14}$$

The main idea is to find a lower bound of p to define the service curve guaranteed to SCT, β^{bls}_{SCT}. On the other hand, if SCT has p opportunities to be transmitted, then the RC traffic (since it is the only traffic class with a priority higher than the lowest priority of SCT during the idle window) has at most $(p+1)$ opportunities to be transmitted during at the worst-case the maximum idle window, Δ^{max}_{idle}, then:

$$\Delta R^*_{RC}(\delta) \leq (p+1) \cdot C \cdot \Delta^{max}_{idle} \tag{15}$$

Giving the strict service curve property of $C \cdot t$ since we have a constant rate server and using Eq. (15), we have:

$$\begin{aligned} C \cdot \delta &\leq \Delta R^*_{SCT}(\delta) + \Delta R^*_{RC}(\delta) \\ &\leq \Delta R^*_{SCT}(\delta) + (p+1) \cdot C \cdot \Delta^{max}_{idle} \end{aligned}$$

Consequently, the lower bound of p is as follows:

$$p \geq \frac{C \cdot \delta - \Delta R^*_{SCT}(\delta)}{C \cdot \Delta^{max}_{idle}} - 1 \tag{16}$$

When injecting Eq.(16) in Eq. (14), we obtain:

$$\begin{aligned} \Delta R^*_{SCT}(\delta) &\geq \left(\frac{C \cdot \delta - \Delta R^*_{SCT}(\delta)}{C \cdot \Delta^{max}_{idle}} - 1 \right) \cdot C \cdot \Delta^{min}_{send} \\ \Delta R^*_{SCT}(\delta) . \left(1 + \frac{C \cdot \Delta^{min}_{send}}{C \cdot \Delta^{max}_{idle}} \right) &\geq \left(\frac{C \cdot \delta}{C \cdot \Delta^{max}_{idle}} - 1 \right) . C \Delta^{min}_{send} \\ \Delta R^*_{SCT}(\delta) &\geq \frac{\frac{\delta}{\Delta^{max}_{idle}} - 1}{\frac{\Delta^{min}_{send}}{\Delta^{max}_{idle}} + 1} \cdot C \cdot \Delta^{min}_{send} \end{aligned}$$

Giving that $\Delta R^*_{SCT}(\delta) \geq 0$, then:

$$\Delta R^*_{SCT}(\delta) \geq \frac{\Delta^{min}_{send}}{\Delta^{min}_{send} + \Delta^{max}_{idle}} \cdot C \cdot (\delta - \Delta^{max}_{idle})^+$$

□

B. Proof of Theorem 4

Proof. First, it is obvious that in the absence of RC traffic, SCT can use the maximum service $\gamma(t) = C.t$. Then, for the more general case, consider $R^*_{SCT}(t)$ the output cumulative function of the SCT at the output of the server implementing a BLS, and $\Delta R^*_{SCT}(\delta)$ the variation of the output cumulative function during δ.

During a backlogged period δ for SCT and RC traffic, the SCT has at most $p+1$ opportunities with p times a full service constrained by γ during the maximum sending window Δ^{max}_{send}, in addition to once during $\Delta^{max}_{send,0}$, then:

$$\Delta R^*_{SCT}(\delta) \leq p \cdot C \cdot \Delta^{max}_{send} + C \cdot \Delta^{max}_{send,0} \tag{17}$$

The main idea is to find an upper bound of p to define the maximum service curve guaranteed to SCT, γ^{bls}_{SCT}. On the other hand, if SCT has at most $p+1$ opportunities to be transmitted, then the RC traffic has at least p opportunities to be transmitted during the minimum idle window, Δ^{min}_{idle}, then:

$$\Delta R^*_{RC}(\delta) \geq p \cdot C \cdot \Delta^{min}_{idle} \tag{18}$$

Giving the maximum service curve property of γ and using Eq. (18), we have:

$$\begin{aligned} C \cdot \delta &\geq \Delta R^*_{SCT}(\delta) + \Delta R^*_{RC}(\delta) \\ &\geq \Delta R^*_{SCT}(\delta) + p \cdot C \cdot \Delta^{min}_{idle} \end{aligned}$$

Consequently, the upper bound of p is as follows:

$$p \leq \frac{C \cdot \delta - \Delta R^*_{SCT}(\delta)}{C \cdot \Delta^{min}_{idle}} \tag{19}$$

When injecting Eq.(19) in Eq.(17), we obtain:

$$\begin{aligned} \Delta R^*_{SCT}(\delta) &\leq \frac{C \cdot \delta - \Delta R^*_{SCT}(\delta)}{C \cdot \Delta^{min}_{idle}} \cdot C \cdot \Delta^{max}_{send} + C \cdot \Delta^{max}_{send,0} \\ \Delta R^*_{SCT}(\delta) \cdot \left(1 + \frac{\Delta^{max}_{send}}{\Delta^{min}_{idle}} \right) &\leq \frac{\delta}{\Delta^{min}_{idle}} \cdot C \cdot \Delta^{max}_{send} + C \cdot \Delta^{max}_{send,0} \\ \Delta R^*_{SCT}(\delta) &\leq \frac{\frac{\delta}{\Delta^{min}_{idle}} \cdot C \cdot \Delta^{max}_{send} + C \cdot \Delta^{max}_{send,0}}{1 + \frac{\Delta^{max}_{send}}{\Delta^{min}_{idle}}} \\ \Delta R^*_{SCT}(\delta) &\leq \frac{\Delta^{max}_{send}}{\Delta^{nom}_{\gamma SCT}} \cdot C \cdot \delta + \Delta^{max}_{send,0} \cdot C \cdot \frac{\Delta^{min}_{idle}}{\Delta^{nom}_{\gamma SCT}} \end{aligned} \tag{20}$$

Where $\Delta^{nom}_{\gamma SCT} = \Delta^{max}_{send} + \Delta^{min}_{idle}$.

□

REFERENCES

[1] Airlines Electronic Engineering Committee. Aircraft Data Network Part 7, Avionics Full Duplex Switched Ethernet (AFDX) Network, ARINC Specification 664. Aeronautical Radio, 2002.

[2] Anne Bouillard, Laurent Jouhet, and Eric Thierry. Service curves in Network Calculus: dos and don'ts. Research report, INRIA, 2009.

[3] Anaïs Finzi, Ahlem Mifdaoui, Emmanuel Lochin, and Fabrice Frances. Incorporating TNS/BLS in the AFDX for Mixed-Criticality Avionics Applications: Specification. In *ERTS2*, 2018.

[4] Franz-Josef Gotz. Traffic Shaper for Control Data Traffic (CDT). *IEEE 802 AVB Meeting*.

[5] Jérôme Grieu. *Analyse et évaluation de techniques de commutation Ethernet pour l'interconnexion des systèmes avioniques*. PhD thesis, INPT, 2004.

[6] J.Y. Le Boudec and P. Thiran. *Network calculus: a theory of deterministic queuing systems for the internet*. Springer-Verlag, 2001.

[7] Jork Loeser and Hermann Haertig. Low-latency hard real-time communication over switched ethernet. In *ECRTS*, 2004.

[8] Zhao Luxi, Pop Paul, Li Qiao, Chen Junyan, and Xiong Huagang. Timing analysis of rate-constrained traffic in TTEthernet using network calculus. *Real-Time Systems*, 2017.

[9] R. Bosch GmbH. CAN specification Version 2,0. Technical report, 1991.

[10] F.-J. Gotz S. Kerschbaum and F. Chen. Towards the Calculation of Performance Guarantees for BLS in Time-Sensitive Networks. *IEEE 802.1 TSN Meeting*, 2013.

[11] N. Concer S. Thangamuthu, P. Cuijpers JL, and J. Lukkien. Analysis of ethernet-switch traffic shapers for in-vehicle networking applications. In *DATE*, 2015.

[12] Perathoner Simon, Wandeler Ernesto, Thiele Lothar, and et al. Influence of different abstractions on the performance analysis of distributed hard real-time systems. *Design Automation for Embedded Systems*, 2009.

[13] W. Steiner, P. Heise, and S. Schneele. Recent ieee 802 developments and their relevance for the avionics industry. In *2014 IEEE/AIAA 33rd DASC*.

[14] D. Thiele and R. Ernst. Formal worst-case timing analysis of Ethernet TSN's burst-limiting shaper. In *DATE*, 2016.

[15] TSN Task Group. TSN Specifications.