This is an author's version published in: https://oatao.univ-toulouse.fr/21206

To cite this version :

Finzi, Anaïs and Mifdaoui, Ahlem and Frances, Fabrice and Lochin, Emmanuel Comparative Analysis of Scheduling Strategies for Heterogeneous Avionics Applications. (In Press: 2018) In: 1st Workshop on Efficient Real-time Data Networks (in conjunction with RTSS), 11 December 2018 - 11 December 2018 (Nashville, United States).

Comparative Analysis of Scheduling Strategies for Heterogeneous Avionics Applications

A. MIFDAOUI, A. FINZI, F. FRANCES and E. LOCHIN
ISAE-Supaéro / Université de Toulouse, France

***Abstract*—A homogeneous avionic communication architecture to interconnect different avionics domains may bring significant advantages, such as easier installation and maintenance in addition to reduced weight and costs. This homogeneous communication architecture needs to support heterogeneous applications, where safety-critical and best effort traffic co-exist. In this paper, we assess the pros and cons of the most relevant scheduling strategies supporting heterogeneous applications versus the main avionics requirements. Furthermore, we conduct a quantitative comparative analysis of the most promising solutions guaranteeing the main avionics requirements through a representative avionics case study. Results show that a recent shaper in Time Sensitive Networks is a promising solution in terms of performance and complexity.**

***Index Terms*—TSN, BLS, AFDX, DRR, NP-SP, avionics, QoS, Schedulers.**

I. INTRODUCTION

Avionics is a field that moved from point-to-point transmissions to high speed networks. However, this field slowly evolves due to the stringent safety requirements and the aircraft long life expectancy, around 25 to 30 years. The comparison of this lifespan against other networking fields is an interesting one. For instance, the last 30 years have seen the development of main stream Internet, from low rate 64Kbit/s to high speed Gigabit fiber connections. Concerning mobile networks, a new generation appears approximatively every 9 years. Hence, between the day in 1990 when an airliner entered into service to its retirement in 2015, a consumer download link was multiplied by 15,000 and 3 mobile network generations were developed. This highlights the stark difference between the closed avionics world, and the Internet and mobile open world.

However, linkages exist between these communities: the newest avionics network, the Avionics Full-DupleX Ethernet (AFDX) [1] is based on a technology developed for the Internet, the Switched Ethernet. The low cost and maturity, after decades of use in industrial markets, are the main advantages of this technology. There are still many technologies from the open world that could be used for avionics networks. In particular in the open world, there is a large number of scheduling strategies to multiplex heterogeneous flows within a network. In this paper, we analyse the most relevant ones to assess their potential use to define an avionics network to support heterogeneous avionics applications.

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 in addition to reduced weight and costs. This homogeneous communication architecture, based on the AFDX technology, needs to support heterogeneous 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.

Hence, we start by presenting the avionics context through the evolution of avionics network and the main avionics requirements in Section II. Afterwards, we assess the pros and cons of the most relevant scheduling strategies supporting heterogeneous applications versus the main avionics requirements in Section III. Finally, we conduct a quantitative comparative analysis of the most promising solutions guaranteeing the main avionics requirements through a representative avionics case study in Section IV.

II. AVIONICS CONTEXT

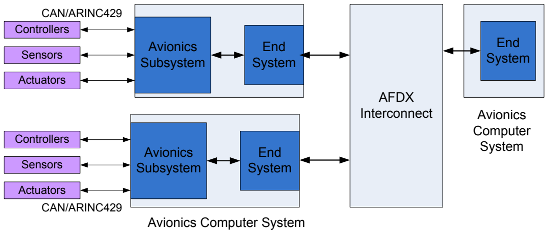


Fig. 1: Current Avionics Network

As shown in Figure 1, the current avionics network, responsible for flight control, cockpit, engines and fuel & landing gears, consists of a high-rate backbone network, the AFDX [1], to interconnect critical subsystems. Moreover, some low-rate data buses, e.g., CAN [12] or ARINC 429 [7], are still used to handle some specific avionics domains, such as the I/O process and the Flight Control Management.

The AFDX [1] network is based on Full Duplex Switched Ethernet protocol at 100Mbit/s, successfully integrated into new generation civil aircraft like the Airbus A380. This technology succeeds to support the important amount of exchanged data due to policing mechanisms added in switches and the

Virtual Link (VL) concept. The latter gives a way to reserve a guaranteed bandwidth to each traffic flow. The VL represents a multicast communication which 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 that can be sent during each BAG. The current AFDX enables the use of both scheduling strategies: First Come First Served (FCFS) and Strict Priority (SP).

The CAN bus [12] is a 1 Mbit/s data bus that operates according to an event-triggered paradigm where messages are transmitted using a priority-based access mechanism. CAN bus works using a producer/consumer communication scheme based on unique identifier per message type. The CAN messages are broadcasted on the bus, then each CAN equipment will filter the consumed data based on the CAN identifier. The collisions on the bus are resolved following a CSMA/CR protocol (Carrier Sense Multiple Access/ Collision Resolution) thanks to the bit arbitration method.

The ARINC429 [7] is a 100 Kbit/s data bus with a point-to-point protocol. It is a mono transmitter multi receivers data bus with unidirectional communication which provides high reliability at the cost of wire weight and limited data rates.

Although this architecture reduces the time to market, it conjointly leads to inherent heterogeneity and new challenges to guarantee the real-time requirements. To enable a homogeneous architecture based on AFDX technology, we identify herein the main avionics requirements and challenges to compare the different scheduling strategies and select the most appropriate one to support heterogeneous flows on the AFDX.

The two main considered avionics requirements are as follows:

- **Predictability**: the impact of a system on an other is known and bounded. The communication architecture must behave in a predictable way, where the extended AFDX has to guarantee bounded latencies respecting the temporal constraints of the heterogeneous traffic.
- **Modularity**: this requirement is related to the flexibility and exchangeability of software and hardware components. An important step towards enhancing the avionics system modularity has been fulfilled with the adoption of the IMA approach [17], i.e., common elementary components can be configured to fit different avionic applications. This feature aims to minimise the (re) configuration and readjustment effort to facilitate system maintenance and its progress over the years. For instance, the event-triggered paradigm of the AFDX is favoring such a requirement.

Moreover, we need to deal with the main challenge of enforcing the Quality of Service (QoS) features, while limiting the impact of the highest priority traffic on the current AFDX traffic and the implementation complexity. These challenges will be denoted by **Fairness**, and **Complexity** along this paper.

III. QUALITATIVE ANALYSIS OF DIFFERENT SCHEDULING STRATEGIES

Various solutions have been proposed in the literature to support heterogeneous applications in embedded systems and particularly in avionics. The first proposed solution is the simplest one, based on Strict Priority like the one specified in the AFDX. Overtime, new solutions with increased complexity were proposed, such as the ones defined in Audio Video Bridging [11] and Time-Sensitive Networking [16].

To quantify the (re)configuration effort needed by an alternative avionics communication architecture in comparison to the current AFDX standard, the considered communication paradigm is of utmost importance since the modularity level of a solution highly depends on such a paradigm. The event-triggered paradigm is known as highly flexible and facilitates the system reconfiguration, but it infers at the same time an indeterminism level and needs further proofs to verify the predictability requirement. On the other hand, the time-triggered paradigm is highly predictable, but presents some limitations in terms of system reconfigurability.

In this section, we will detail the different scheduling strategies and assess their potential ability vs the avionics requirements.The different solutions can be categorized according to the required communication paradigm, i.e., mainly time-triggered or event-triggered.

Non-Preemptive Strict Priority Scheduler The Non-Preemptive Strict Priority (NP-SP) scheduling strategy is the simplest QoS implementation with very limited complexity. Each queue has a defined priority and the scheduler dequeues the first frame of the eligible queue (a queue with enqueued traffic) with the highest priority. This scheduler is defined in the AFDX standard [1]; and due to the leaky bucket shapers in the end-systems and policers in the switches, NP-SP guarantees the predictability requirement.

NP-SP is compliant with an event-triggered paradigm, which allows a high modularity level, but it is a well-known as an unfair scheduler [18].

GPS-like Schedulers–Deficit Round Robin

The Generalized Processor Sharing (GPS) is an idealized scheduling algorithm that achieves perfect fairness: the bandwidth is shared depending on fixed weights. Many algorithms have been developed to come as close as possible to the GPS, such as the Weighted Fair Queuing (WFQ) [4] or Weighted Round Robin (WRR) [19] and Deficit Round Robin (DRR) [9]. Ordinary round-robin servicing of queues can be done in constant time. With WRR, the usual implementation consists in setting a number of frames that can be consecutively sent for each queue. The major problem, however, is the unfairness caused by possibly different packet sizes used by different flows. This flaw can be removed by using a counter to keep track of traffic transmitted as with the Deficit Round Robin (DRR). Nonetheless, these schedulers necessitate a virtual clock, which increases their implementation complexity. In [9], an AFDX network implementing the DRR has been specified and studied.

Results have shown the good performances of the proposal in terms of predictability and fairness, while increasing the implementation complexity. Moreover, like NP-SP, DRR offers a high modularity level due to its compliance with an event-triggered paradigm.

Audio-Video Bridging–Credit Based Shaper

In recent years, there has been a strong interest in the IEEE 802.1 Audio/Video Bridging (AVB) protocol, which provides end-to-end delay guarantees in Ethernet networks. AVB specifies a credit-based shaping (CBS) algorithm for real-time (RT) traffic classes A and B. Each shaped class has a credit-counter, which is replenished at a constant rate (the so-called idle slope) and consumed at the rate allowed by the port (the send slope) when data on the specific class is transferred. When the queue is empty, the credit immediately returns to 0. The different classes are scheduled using a strict priority scheduler, with the CBS preventing the starvation of lower priorities and giving bandwidth guarantees, which are good properties for mixed-criticality applications.

Concerning the predictability of CBS, the different classes are isolated from each other thanks to the counter and their associated blocking effect. However, it has been shown in [2] that the impact of the blocking effect of the AVB on the latency is high, which induces a medium predictability level for this shaper. However, the worst-case latency of unshaped lower priorities is improved due to the shaping of classes A and B, which fulfills the fairness challenge. The main drawback of the CBS is that frames cannot be transmitted if the credit is below 0, no matter the state of the other queues. This fact can cause unnecessary delays if other queues are empty. This issue has been fixed by the TSN [16] task group through different shapers.

Time Sensitive Networking–Time Aware Shaper

TAS[15] uses time-driven scheduling to manage link access between traffic classes, which makes it a good candidate for heterogeneous traffic flows. For each traffic class, the frames are transmitted according to a gate schedule at each output port: it allows frames to pass when opened, and it blocks frames when closed. The different gate schedules are programmed offline, and multiple gates can be opened at the same time. Then, the selected frames are arbitrated according to their priority levels. To prevent frames transmission when the gate is closed, TAS defines guard bands. From the start of a guard band until the gate is opened, no new frames of the corresponding class are allowed to start transmission.

Due to the gate schedule, TAS guarantees a high predictability level, but the modifications are propagated to all flows. This fact limits the TAS modularity, while inferring high implementation complexity. Additionally, when lower classes gates are opened, they are scheduled using a strict priority, which implies a low fairness.

Time Sensitive Networking–Peristaltic Shaper

The Peristaltic Shaper (PS) [14] uses a global time divided in odd and even phases to manage different traffic classes. If a shaped frame arrives in an odd (resp. even) phase, it can not be sent before the start of the next even (resp. odd) phase. The idle time can be used by other priorities. The Peristaltic Shaper has been proposed by the same task group as TAS. Hence, they have often been studied together and similar work has been done.

Similarly TAS, the use of a global time in PS implies a high predictability level but a negative impact on its modularity and implementation complexity: a flow modification can impact the calculation of odd and even phases not only along its path, but also on other flows paths. However, due to the initial waiting time caused by the odd and even phases, lower priority flows may be sent more quickly than under Static Priority scheduler, which makes Peristaltic Shaper an interesting solution in terms of fairness.

Time Sensitive Networking–Urgency-based Scheduler

The main idea of the Urgency-based Scheduler (UBS) [13] is a separation between per flow and per queue. The conceptual separation of per flow queue and state provides per flow shaping at every hop for flow aggregated in the queues. This concept is called $interleaved\ shaping$. This significantly reduces the algorithmic complexity by limiting the number of required queues. Hence the first step when a new frame arrives in the output port is to select the appropriate queue depending on the priority of the flow and its ”urgency” as decided by an interleaving algorithm.

This scheduler is still new, so little research has been done yet. In [13], the scheduler is presented, simulations and timing analysis are performed. The results show high link utilisation and low delays. They also conclude that the implementation complexity is low, in part because they assume the queue selection process is already implemented in the switches thanks to the standardisation of 802.1Qci-Per-Stream Filtering and Policing. But, while implementing it in higher layer is simple, implementing at the hardware level is much more complex.

Time Sensitive Networking–Burst Limiting Shaper

Presented in [8], the BLS is a credit-based shaper that has been characterized in [8] 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 shaped by BLS can vary between a high and a low value. The low value is usually below the lowest priority of unshaped traffic.

BLS is used with a strict priority scheduler, where BLS modifies the priority seen by the SP depending on a credit counter. Hence, depending on the priority value, the shaped frames can be blocked or not by other classes. However, no matter the state of the credit, if a frame is the first of the queue with the highest priority among the eligible queues, then it will be transmitted. Thus, contrary to CBS, the BLS is a non-blocking shaper, which is a large improvement of the predictability guarantees.

The priority change feature enables the BLS to reserve bandwidth for the shaped queue. This fact induces a low implementation complexity; and also improves fairness in comparison to SP, since it limits the bandwidth available to the shaped queue.

A. Discussion

In this section, we assess the pros and cons of the different scheduling strategies vs the four avionics requirements and challenges, to select the most promising ones:

- **predictability:** thanks to the leaky bucket shapers in the AFDX end-systems and the policers in the switches, all the presented solutions can achieve the necessary determinism and isolation. However, AVB/CBS sometimes blocks frames when the transmission link is free, causing unnecessary delays;
- **modularity:** the solutions compliant with event-triggered paradigm, i.e., NP-SP, DRR, CBS, UBS and BLS, better fulfill the modularity criterion, contrary to time-triggered solutions like TAS and PS;
- **fairness:** as aforementioned, there are four solutions fulfilling the fairness constraint: DRR, CBS, PS and BLS;
- **Complexity:** time-triggered solutions like TAS and PS necessitate the implementation of a complex time synchronisation and induce high complexity; Whereas, CBS, BLS and UBS can be used independently from the synchronisation aspect of AVB and TSN. Nevertheless, UBS induces higher complexity.

The considered solutions vs the main avionics requirements and challenges are illustrated in Table I. Hence, the most promising solutions in the avionics context are DRR and TSN/BLS. The quantitative analysis of these scheduling strategies performance will be conducted, with reference to the already specified solution in the AFDX standard NP-SP scheduling strategy.

Solutions	references	Requirements			
NP-SP	[18]	✓✓	✓✓	X	✓✓
GPS/DRR	[9]	✓✓	✓✓	✓✓	✓
AVB/CBS	[2]	✓	✓✓	✓✓	✓✓
TSN/TAS	[15]	✓✓	X	X	X
TSN/PS	[14]	✓✓	X	✓	X
TSN/BLS	[8]	✓✓	✓✓	✓✓	✓✓
TSN/UBS	[13]	✓✓	✓✓	✓✓	X
Avionics requirements and challenges	Predictability Modularity Fairness Complexity				

TABLE I: Existing solutions vs avionics requirements and challenges

IV. QUANTITATIVE ANALYSIS OF DIFFERENT SCHEDULING STRATEGIES

In this section, we conduct performance analysis of the most promising scheduling strategies (BLS and DRR) when incorporated in the AFDX, to evaluate their efficiency to support heterogeneous traffic profiles, in comparison to the current AFDX solution (implementing SP scheduler). First, we describe our representative avionics case study and the testing scenarios. Afterwards, we assess the timing performance and complexity of the selected solutions, 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[1] backbone network, which consists of 4 switches and 64 end-systems as shown in Fig. 2 (a). The different traffic profiles generated by each end-system are described in Tab. II. Each traffic class $j \in \{SCT, RC, BE\}$ is characterized by $(MFS_j, BAG_j, Deadline_j)$. Figure 2 (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. This metric enables the scalability analysis of the extended AFDX with the new scheduling strategies BLS and DRR, 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 implementing SP. 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. The delay bounds are computed based on Network calculus [10], and particularly the proved results in [5] for BLS and [3] for DRR.

(iii) The **computation time** to tune the parameters of each scheduling strategy to improve as much as possible the system performance when using the tuning methods described in [6].

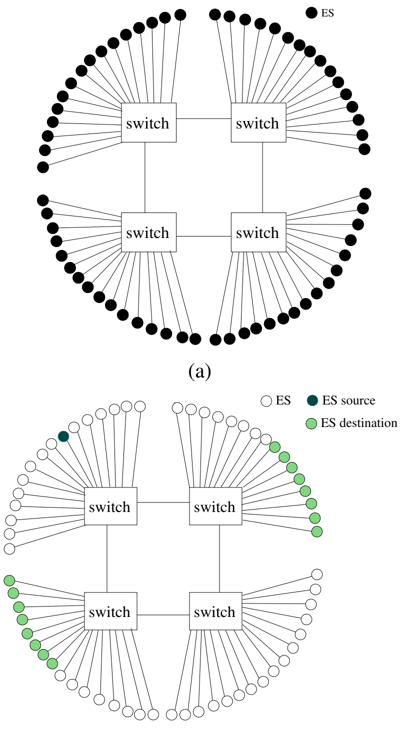
Priority	Traffic Class	MFS (Bytes)	BAG (ms)	Deadline (ms)
High	SCT	64	2	2
Medium	RC	320	2	2
Low	BE	1024	8	none

TABLE II: Avionics flow Characteristics

The testing scenarios are described in Table III. As it can be noticed, the principle of scenario 1 (resp. scenario 2) is to fix the utilisation rate of RC class UR_{RC} (resp. SCT class UR_{SCT}) at 20% and vary the SCT (resp. RC) utilisation rate to assess the impact of increasing network congestion on the timing performance. 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 $j \in \{RC, SCT\}$, $C_j = 16 \cdot n_j^{es} \cdot \frac{MFS_j}{BAG_j}$, and C the transmission capacity of the network (1Gbit/s).

---

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



(a)

(b)

Fig. 2: Representative AFDX network: (a) Architecture; (b) Traffic Communication Patterns

Scenarios	Scenario 1	Scenario 2
$(UR_{RC}; UR_{SCT})(\%)$	(20; [1..80])	([1..80]; 20)
$(n^{es}_{RC}; n^{es}_{SCT})$	(10; [1 : 4 : 110])	([1 : 2 : 39]; 47)

TABLE III: Testing Scenarios 1 and 2

B. Numerical Results

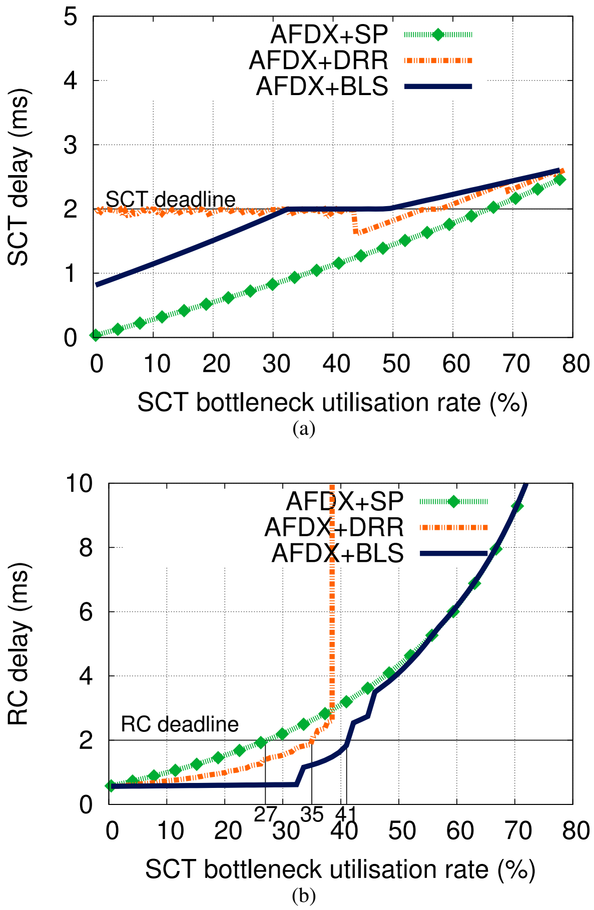
The results of scenarios 1 and 2 are illustrated in Fig.3 and Fig.4, respectively.

First, concerning the maximum bottleneck utilisation rates:

- in Figure 3, we note that the maximum bottleneck SCT utilisation rate is 27% with the current AFDX (AFDX+SP), 35% with DRR-compliant AFDX (AFDX+DRR) and 41% with the extended AFDX incorporating BLS (AFDX+BLS).
- in Figure 4, the maximum bottleneck RC utilisation rate is 33% with SP, 38% with DRR and 41% with BLS.

Hence, incorporating BLS in the AFDX improves the maximum utilisation rate of RC, compared to both SP (up to 24%) and DRR (up to 17%).

Secondly, concerning timing performance and delay bounds, in Figure 3(b), the RC delay bounds with BLS are lower than the delay bounds with either DRR and SP. In particular, the BLS improves the RC delay bound up to 77% compared to the current AFDX with SP, and up to 73% compared to DRR-compliant AFDX (when the SCT and RC deadlines are fulfilled). The same behaviour is visible in Figure 4(b): the BLS improves the RC delay bounds up to 89% compared to SP, and up to 38% compared to DRR (when the deadlines are fulfilled).



(a)

(b)

Fig. 3: Testing Scenario 1 Results: (a) SCT delay bounds; (b) RC delay bounds

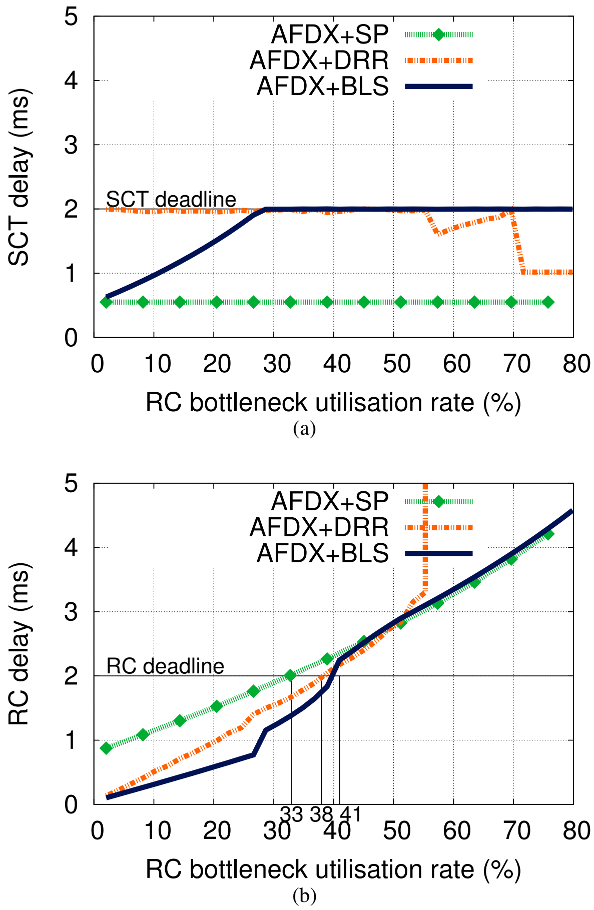
The improvements of the RC delay bounds and schedulability with the BLS and DRR scheduling strategies in AFDX, in reference to the current AFDX (SP), are illustrated in Table IV. We have also computed the computation times to tune the parameters of BLS and DRR to achieve the best performance.

First, we can see that the BLS and DRR improve both the RC delay bounds and the maximum utilisation rates of SCT and RC, compared to SP. We note that the positive impact is much stronger under the BLS than under the DRR. Moreover, we can see that the computation time is multiplied up to 6 times under DRR, in reference to BLS.

Scheduler	improvement compared to SP(%)				computation times (s)	
	maximum RC delay at		maximum			
	$UR^{bn}_{SCT} = 33\%$	$UR^{bn}_{RC} = 28\%$	UR^{bn}_{SCT}	UR^{bn}_{RC}	SCT	RC
BLS	18	22	33	21	57	9
DRR	18	16	26	15	395	58

TABLE IV: Comparing Scheduling Strategies

From these scenarios, we can conclude that with an accurate parameter tuning, the extended AFDX implementing BLS has a large positive impact on both SCT and RC, compared to the current AFDX implementing the SP scheduler or DRR-compliant AFDX, while inducing low complexity.



(a)

(b)

Fig. 4: Testing Scenario 2 Results: (a) SCT delay bounds; (b) RC delay bounds

V. CONCLUSIONS

In this paper, we have assessed the most relevant existing scheduling strategies vs the main avionics requirements, to support heterogeneous applications on the AFDX network. Afterwards, we have conducted a quantitative performance analysis of the most promising solutions, i.e., BLS and DRR, in reference with the current one (SP) through a representative avionics case study. Results show the noticeable performance enhancement of the current AFDX traffic (RC) in presence of the highest priority one (SCT) under BLS, with reference to the current AFDX (SP) and DRR, while keeping a low complexity.

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] Unmesh D Bordoloi, Amir Aminifar, Petru Eles, and Zebo Peng. Schedulability analysis of ethernet avb switches. In *Embedded and Real-Time Computing Systems and Applications (RTCSA), 2014 IEEE 20th International Conference on*, pages 1–10. IEEE, 2014.

[3] Marc Boyer, Giovanni Stea, and William Mangoua Sofack. Deficit Round Robin with network calculus. In *Performance Evaluation Methodologies and Tools (VALUETOOLS)*, 2012.

[4] Alan Demers, Srinivasan Keshav, and Scott Shenker. Analysis and simulation of a fair queueing algorithm. In *ACM SIGCOMM Computer Communication Review*, volume 19. ACM, 1989.

[5] Anaïs Finzi, Ahlem Mifdaoui, Emmanuel Lochin, and Fabrice Frances. Network Calculus-based Timing Analysis of AFDX Networks with Strict Priority and TSN/BLS Shapers. In *SIES*, 2018.

[6] Anaïs Finzi, Ahlem Mifdaoui, Emmanuel Lochin, and Fabrice Frances. Performance Enhancement of Extended AFDX via Bandwidth Reservation for TSN/BLS Shapers. In *RTN workshop in conjunction with ECRTS*, 2018.

[7] Christian M. Fuchs, Advisors Stefan Schneele, and Er Klein. The evolution of avionics networks from arinc 429 to afdx. In *In Proceedings of the Seminars Future Internet (FI), Innovative Internet Technologies and Mobile Communication (IITM) and Aerospace Networks (AN)*, 2012.

[8] Franz-Josef Gotz. Traffic Shaper for Control Data Traffic (CDT)–IEEE 802 AVB Meeting. http://www.ieee802.org/1/files/public/docs2012/new-goetz-CtrDataScheduler-0712-v1.pdf, 2012.

[9] Yu Hua and Xue Liu. Scheduling design and analysis for end-to-end heterogeneous flows in an avionics network. In *INFOCOM, 2011 Proceedings IEEE*, pages 2417–2425. IEEE, 2011.

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

[11] Hyung-Taek Lim, Daniel Herrscher, Martin Johannes Waltl, and Firas Chaari. Performance analysis of the ieee 802.1 ethernet audio/video bridging standard. In *Proceedings of the 5th International ICST Conference on Simulation Tools and Techniques*, 2012.

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

[13] Johannes Specht and Soheil Samii. Urgency-based scheduler for time-sensitive switched ethernet networks. In *ECRTS*, 2016.

[14] Michael Johas Teener. Back to the future:using TAS and pre-emption for deterministic distributed delays–IEEE 802.1 AVB TG Meeting, San Antonio. http://www.ieee802.org/1/files/public/docs2012/new-avb-mjt-back-to-the-future-1112-v01.pdf, 2012.

[15] Daniel Thiele, Rolf Ernst, and Jonas Diemer. Formal worst-case timing analysis of Ethernet TSN's time-aware and peristaltic shapers. In *VNC*. IEEE, 2015.

[16] TSN Task Group. TSN Specifications. http://www.ieee802.org/1/pages/tsn.html.

[17] C. Watkins and R. Walter. Transitioning From Federated Avionics Architectures To integrated Modular Avionics. 26th Digital Avionics Systems Conference (DASC), 2008.

[18] Adam Wierman and Mor Harchol-Balter. Classifying scheduling policies with respect to unfairness in an m/gi/1. In *ACM SIGMETRICS Performance Evaluation Review*, volume 31, pages 238–249. ACM, 2003.

[19] Tianran ZHOU and Xiong Huagang. Design of energy-efficient hierarchical scheduling for integrated modular avionics systems. *Chinese Journal of Aeronautics*, 2012.