Towards Quality of Service Provision with Avionics Full Duplex Switching

Oana Hotescu, Katia Jaffrès-Runser, Jean-Luc Scharbarg and Christian Fraboul
Institut de Recherche en Informatique de Toulouse, Université de Toulouse, INPT,
2 rue Charles Camichel, 31300 Toulouse, France
Email: {oana.hotescu, kjr, jean-luc.scharbarg}-at-enseeiht.fr

***Abstract*—Avionics Full-Duplex Switched Ethernet (AFDX) has been designed to carry exclusively time-critical flows dictated by the severe real-time constraints of avionics. Certification methods ensure end-to-end communication delays are upper bounded at the cost of an over-provision of network bandwidth. This paper investigates the issues related to the efficient use of such wasted communication capacity. The basic idea is to push additional Ethernet flows of lower priority into the network with the goal of offering different quality of service levels to them. This addition should of course be transparent to real-time flows. Different scheduling policies, enforced at the end system and at the switches, can provide such features. This paper questions i) their impact on the real-time flow end-to-end delay and ii) the quality of service non real-time flows can expect in terms of end-to-end delay distribution. The ultimate goal of this study is to decide if such a network configuration offers the opportunity to deploy multimedia applications such as VoIP traffic, live video streaming over an AFDX industrial configuration. A first study investigates the use of static priority queuing and traffic scheduling tables on an industrial A350 AFDX configuration. Traffic scheduling tables are shown to be promising as they can spread non-critical traffic more evenly over time, reducing congestion peaks at switches.**

I. Application domain & Challenge

A. Application domain

Avionics Full DupleX switching (AFDX) networks have become the de-facto technology for commercial avionics embedded networks in the last decade. AFDX follows ARINC664 Part 7 norm [1] and is currently deployed by the biggest aircraft manufacturers. A complex industrial ecosystem has been created around this technology to efficiently design, prototype and certify such networks. Even though alternative solutions are being investigated, this protocol will remain in use for couple of decades in the industry.

AFDX is a switched Ethernet network where network interface cards (the so-called *end systems*) can only emit frames following a pre-defined virtual link (VL) setting. A VL is defined by minimum and maximum frame sizes, together with a BAG duration (Bandwidth Allocation Gap) that expresses the minimum duration that separates two consecutive frames of the VL. A given bandwidth is as such associated to a VL. BAG durations are selected within the discrete power set of {1, 2, 4, 8, 16, 32, 64, 128} milliseconds. AFDX switches are configured statically with VL routes that can be set as multicast. For each egress port, two queues with different priority levels exist.

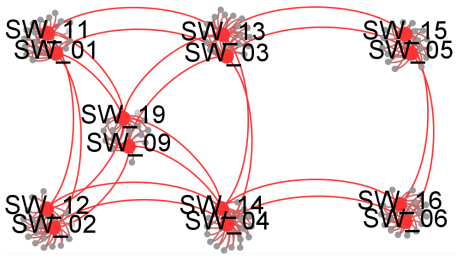

Fig. 1: Example of an A350-like AFDX network topology. Switches are given by red vertices, other nodes representing end systems.

Figure 1 represents an example network similar to the ones deployed on an A350 aircraft. It is composed of 126 end systems interconnected by two redundant AFDX networks composed of 7 switches each. End systems generate 1106 VLs.

B. Challenge

AFDX is a hard real-time network since the end-to-end delay between communicating end systems can be upper bounded. In case a VL is multicast to several end systems, a worst-case bound has to be calculated for each destination (i.e. each VL path). Industrial configurations are certified using network calculus [2] but other methods have been derived to offer tighter bounds [3]. Some of them introduce the concept of offsets to model the local synchronization of all VLs being emitted by the same end system. It is of course utterly important to guaranty the timely transmission of avionics frames but this safety provision comes at the cost of an important over-provisioning of bandwidth.

Our overarching goal is to show that it is possible to take advantage of this wasted bandwidth to push additional flows of lower criticality in this network, for some of which we can still offer soft real-time guarantees. Therefore, it is necessary to decide on a scheduling policy of flows that i) guaranties the timeliness of critical flows and ii) enables the provision of differentiated levels of quality of service for other Ethernet flows.

II. Motivation

The motivation comes from the avionics industry. The targeted additional Ethernet flows would carry live video flows

Link ID	Source node	Destination node	Number of Ethernet	VLs
1	SW_11	CDAU_1	3	88
2	SW_11	SW_19	3	51
3	SW_11	SW_13	3	36
4	SW_13	DU_UP_CTR	6	90
5	SW_13	SW_15	4	77
6	SW_15	SW_16	4	118
7	SW_15	DU_OUT_CAPT	6	88
8	SW_16	DU_OUT_FO	4	89
9	SW_19	CDAU_9	5	99

TABLE I: Congested links shared by VLs and Ethernet flows. All links hold VLs of BAGs {8, 16, 32, 64, 128} ms, only link 1 holds a VL of BAG 1ms as well.

or Voice over IP (VoIP) traffic. Video flows originate from the cameras located at various locations on the airplane while VoIP flows carry messages from the operating crew. Having to deploy a dedicated network for such purposes seem clearly economically inefficient. If such deployment of additional quality of service (QoS) flows succeeds, the AFDX standard can moreover become an even more lasting solution as today.

III. Problem statement

The central idea of this work in progress paper is to compare two different QoS scheduling schemes to extract the best candidate for advanced Ethernet QoS provisioning. Two different QoS policies are investigated in this preliminary study. They are compared to a non-QoS aware FIFO policy where all frames, being time-critical or not, are pushed to a single FIFO queue at the end-systems and at the egress ports of the switches.

Both QoS policies presented hereafter use static priority queuing (SPQ) policies at switch egress ports. High priority is assigned to avionics VLs and low priority to Ethernet flows. The two policies differ by the scheduling algorithm used locally at end systems. End system scheduling policy is only applied to the VLs and flows sharing the same origin end system. Both QoS policies are introduced next.

A. SPQ scheduling.

In this policy, flows and VLs are scheduled at end systems using the same 2-level static priority queuing algorithm as for egress ports of switches.

B. Table scheduling.

This policy has been designed to mitigate the jitter of real-time flows. A scheduling table is defined composed of slots of 31.25μs. The whole table duration is of 128ms leading to 4096 slots. It can be represented by a table of 128 lines of 1ms, each line containing 32 slots. AFDX VLs are allocated to a set of slots according to their BAG duration: a VL gets a slot exactly every BAG ms. For instance, a VL with a BAG of 32ms could get the slots of column 10 at lines 0, 31, 63 and 95. This schedule ensures that frames of a VL leave the end system at dates multiple of its BAG, which completely mitigates the jitter of the VL at emission. As such, a VL frame can only experience additional delay at egress ports of switches due to the interference of other VL frames.

Using this scheduling table, a fixed offset can be set for each VL start. In the previous example, this offset is of $10\times31.25\mu$s. As shown in previous works, these offsets can be accounted for in a worst-case delay analysis [4]. It has even been shown that using offsets, less pessimistic bounds are computed as traffic can be spread such as to reduce concurrent arrivals at switches' egress ports.

No slots are allocated to Ethernet flows. Ethernet frames are stored in a FIFO queue and are sent whenever there is enough time for their emission before the next allocated slot arrives. If there is not enough time to send the Ethernet frame, frame waits for next long enough gap in the table. This feature avoids VLs to be delayed by ongoing Ethernet frames emissions. This table scheduling is reminiscent of scheduling policies implemented in TSN [5], but we recall that such scheduling is not done at the switches where SPQ is being used. As such, network synchronization isn't required, which is in line with avionics safety rules that forbid synchronization in civil aircrafts.

Delay (μs)	Table	SPQ	FIFO
$\geq$300	10000	10000	10000
$\geq$400	5004	5231	5026
$\geq$500	2189	2299	2183
$\geq$600	952	956	922
$\geq$700	404	413	392
$\geq$800	185	190	174
$\geq$900	82	89	85
$\geq$1000	35	39	37
$\geq$1100	11	20	19
$\geq$1200	4	8	7
$\geq$1300	2	4	3
$\geq$1400	0	1	1

TABLE II: Complementary cumulative distribution function of the end-to-end delay of Ethernet frames for table scheduling, SPQ and FIFO policies.

	Table	SPQ	FIFO
Sent frames	10000	10000	10000
Minimum delay (μs)	364	364	364
Received frames with minimum delay	4996	4974	4769
Maximum delay (μs)	1371.8	1444.2	1444.2
Mean delay (μs)	445.52	448.899	445.171
Standard deviation	113.62	115.896	113.771

TABLE III: End-to-end delay distribution for table scheduling, SPQ and FIFO with a Poisson mean rate of 0.000125s

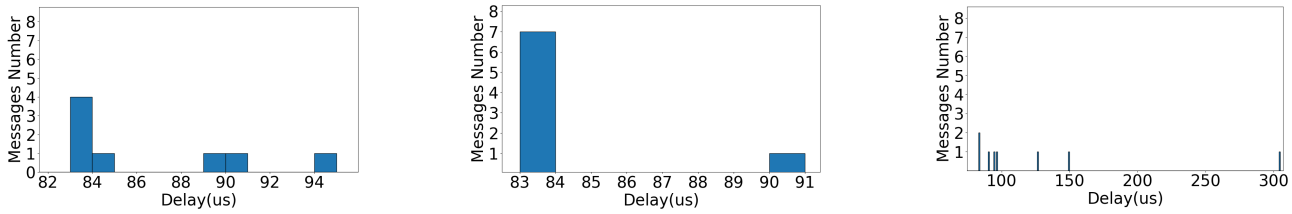

Fig. 2: End-to-end delay of AFDX traffic with a Poisson mean rate of 0.000125s for AFDX table scheduling (left), SPQ (center) and FIFO (right)

IV. Proposed approach and preliminary results

A. A Simulation study

Our main matter is to measure the end-to-end delay distribution for non time-critical Ethernet flows with respect to different scheduling policies. We measure as well this distribution for the real-time VLs. This distribution isn't meant to measure the worst case delay of real-time flows. We are looking for the average time behavior of the VL frames to highlight the impact of our scheduling policies on delays. As mentioned previously, sure upper bounds can be derived for worst case analysis for the tested scheduling policies.

Following results have been obtained with an in-house AFDX simulator developed with the OMNET++ simulation framework. OMNeT++ is a C++-based discrete event simulation environment which enables the modeling and the simulation of network components and communications. Our simulator captures the core parameters of AFDX (routing of virtual links, queuing policies, switching latency, etc.) and is able to simulate 0.5s of the complete A350 configuration in 5 minutes on a regular personal laptop.

B. Ethernet flows

In addition to the avionics flows, we generate Ethernet flows for which we have created specific routes in the switches such as to create congestion on some links. These Ethernet flows aren't constrained by any BAG value. In our simulations, 6 Ethernet flows have been added to the A350 configuration of Figure 1. These flows have been chosen such as to create congestion on links where avionics traffic is relatively heavy. The selected VLs are characterized by a BAG value between 16 ms to 64 ms and a maximum frame length of 350 bytes. In our simulations, VL frames are generated periodically with a frame generation period of at least the BAG. Table I lists the congested links together with the respective number of flows of each kind.

Ethernet frame length is fixed to 1500 bytes, network data rate to 100 $Mbits/s$ and switching delay to 2 μs. For this set of parameters, simulation duration is established to 5 seconds. Ethernet results are presented for the first 10000 Ethernet frames arrived at their destination. Ethernet frames are generated according to a Poisson distribution. Several runs have been launched with different Poisson mean rate values in order to establish how the load of Ethernet flows impacts end-to-end latency and jitter of VLs and Ethernet flows. In the following, we show the results for a total load close to link saturation which corresponds to Ethernet flows sending ~ 8000 frames/s or to a Poisson mean rate of 125ms.

C. Preliminary results

Tables II and III show the statistics of the end-to-end delay of Ethernet frames for all three scheduling policies and Figure 2 the distribution of the end-to-end delay of VLs. VLs have been generated at time 0 and the first frame is send after one BAG has elapsed (i.e. we have an offset of one BAG here). In this specific configuration of the network, the minimum end-to-end duration of a VL frame of 330 bytes is of 84 μs. In this example, the minimum BAG value of the VLs generated by the current end system is 32 ms. Non surprisingly, SPQ provides the lowest delay for VLs while for FIFO, Ethernet flows clearly interfere with VL frames. Table scheduling induces slightly more jitter at the switches than SPQ as frames are spread differently over time. Looking at the Ethernet frame delay statistics, FIFO offers the smallest delays as frames interleave with VL frames. Between SPQ et table scheduling, the latter policy reduces the number of frames with larger delays compared to SPQ.

V. Envisioned solution

From this preliminary study, our intuition shows that table scheduling is a good compromise and could be optimized to offer the best possible QoS setting to Ethernet flows. Indeed, the allocation of VLs to table slots (i.e. the selection of offsets) can be chosen such as to i) mitigate the jitter induced by the network for VLs and ii) reduce the end-to-end delay and jitter of Ethernet flows. Further investigations will as well look at the benefit of frame preemption.

References

[1] Aeronautical Radio Inc. ARINC 664, *Aircraft Data Network, Part 7: Avionics Full Duplex Switched Ethernet (AFDX) Network*, 2005.

[2] Hussein Charara, Jean-Luc Scharbarg, JÃl'rÃt'me Ermont, Christian Fraboul. *Methods for bounding end-to-end delays on an AFDX network.* ECRTS'2006, pages 193-202, Dresde, july 2006.

[3] Henri Bauer, Jean-Luc Scharbarg, Christian Fraboul. *Improving the worst-case delay analysis of an AFDX network using an optimized trajectory approach.* IEEE Transactions on Industrial Informatics, vol.6, no 4, pp 521–533, Nov. 2010.

[4] Xiaoting Li, Jean-Luc Scharbarg, Christian Fraboul. *Worst-case delay analysis on a real-time heterogeneous network.* SIES'2012, Karlsruhe, june 2012.

[5] *IEEE Time-Sensitive Networking Task Group*. [Online]. Available: Http://www.ieee802.org/1/pages/tsn.html