Delay Analysis of AVB traffic in Time-Sensitive Networks (TSN)

Dorin Maxim
LORIA - University of Lorraine
Nancy, France
dorin.maxim@loria.fr

Ye-Qiong Song
LORIA - University of Lorraine
Nancy, France
ye-qiong.song@loria.fr

ABSTRACT

Future autonomous vehicles and ADAS (Advanced Driver Assistance Systems) need real-time audio and video transmission together with control data traffic (CDT). Audio/video stream delay analysis has been largely investigated in AVB (Audio Video Bridging) context, but not yet with the presence of the CDT in the new TSN context. In this paper we present a local delay analysis of AVB frames under hierarchical scheduling of credit-based shaping and time-aware shaping on TSN switches. We present the effects of time aware shaping on AVB traffic, how it changes the relative order of transmission of frames leading to bursts and worst case scenarios for lower priority streams. We also show that these bursts are upper-bounded by the Credit-Bases Shaper, hence the worst-case transmissions delay of a given stream is also upper-bounded. We present the analysis to compute the worst case delay for a frame, as well as the feasibility condition necessary for the analysis to be applied. Our methods (analysis and simulation) are applied to an automotive use case, which is defined within the Eurostars RETINA project, and where both control data traffic and AVB traffic must be guaranteed.

CCS CONCEPTS

•**Computer systems organization** →**Embedded systems;** •**Networks** →Network reliability;

KEYWORDS

Time Sensitive Network (TSN), Audio-Video Bridging (AVB), Ethernet, Delay analysis, Automotive, Real-time

1 INTRODUCTION

IEEE 802.1 TSN (Time-Sensitive Networking) technology [9] is the most important next evolution for in-vehicle networks. It comes from Ethernet AVB by adding several new features such as precise time synchronization through IEEE 802.1AS-2011 [7] (a specific profile of IEEE 1588 PTP) and TAS (Time-Aware Shaper) for supporting both hard real-time and soft real-time constraints.

Enabled by the precise time synchronization, a key new feature of TSN is the definition of the new traffic shaping mechanism TAS, which is capable of accommodating hard real-time streams with deterministic end-to-end delays. TAS uses a pre-defined TDMA-like scheduler that guarantees timely transmission of the most critical CDT (Control Data Traffic). Furthermore, a guard band is added for ensuring that non-CDT traffic such as audio, video and best effort will not interfere with CDT traffic. TAS defines a fixed and periodic schedule cycle of time intervals, which specifies when a traffic queue is enabled (opened) or disabled (closed) for transmission. This schedule is configured offline. Opening/closing is implemented by associating each queue of the output port (up to 8 queues) with a time-aware transmission gate. A TAS cycle represents a complete CDT transmission period, which is $500\mu s$ according to IEEE 802.1 TSN [9], [12].

Although the offline building of the TDMA schedule could be complex, CDT traffic can always be guaranteed by using TAS [6], if there is enough bandwidth. Nevertheless, the use of TAS has an impact on the AVB traffic and as a consequence, the delay analysis of the AVB traffic classes must take into account the underlying delay incurred due to TAS.

In this paper, we focus on the delay analysis of the audio and video traffic classes in the TSN context. This is motivated by the automotive application use case defined within Eurostars RETINA project[1] by extending the first use case of [12], where video streams are main bandwidth demands for autonomous driving and active safety. In our model, after traffic classification according to eight IEEE801.1Q priorities, each audio or video stream is first constrained by a CBS (Credit-Based Shaper), then transmitted during a non-CDT slot which is shared by all non-CDT traffic according to their priorities. To our best knowledge, there is no existing delay analysis for the CBS+TAS scheduling model in the literature.

Delay analysis of AVB without TAS has been extensively studied [3]. We can also find works related to the delay analysis of TSN without CBS for AVB [13], [5]. The closest work to our AVB-TAS model is that of AVB ST [1] with several differences in the model, most notably being that in [1] it is considered CDT time slots are created "on the fly" when a CDT frame needs to be transmitted, as opposed to the TSN approach where the TAS uses a scheduling table to specify when CDT frames are transmitted.

So it is necessary to develop an analysis on CBS+TAS for making practical use of audio and video streaming in autonomous driving.

Contributions: The main contribution of this paper is to give the first analysis of AVB frames using hierarchical CBS+TAS scheduling, knowing that the transmission of these frames is influenced by two shapers (directly by CBS and indirectly by TAS) as well as by their priorities (with respect to other AVB classes) and by FIFO

---

[1] http://retinaproject.eu/

ordering with respect to frames of the same AVB class. In addition, we present a first comprehensive description of the impact of the TAS over AVB streams showing how TAS influences the relative order of transmission of AVB streams, leading to bursts of frames and worst case-transmission delays for lower priority streams. We also apply our analysis framework to an automotive inspired use-case and compare the results of the analysis with simulation results, showing that the analysis is safe for all cases that we have regarded and the tightness varies depending on the load of the switch and the complexity of the TAS scheduling table.

Organization of the paper: in Section 2 we present the related work on which our own contribution is based. Sections 3 presents the model of the network that we are interested in analyzing, the effects of the time aware shaper on the AVB traffic and the formalization of the problem that we solve in the paper. Section 4 presents the main contribution of our work, which is the worst case transmission delay analysis of an AVB frame in a TSN switch and Section 5 presents an automotive case study which we simulate and analyze with our proposed solution. We conclude the paper in Section 6.

2 RELATED WORK

Our work is based on the IEEE 802.1Qbv TSN standard [8, 9] which enhances Ethernet networks to support time sensitive applications, meant to be used in the automotive and industrial control domains. A key new feature of TSN is the definition of the new traffic shaping mechanism TAS (Time Aware Shaper), which is capable of accommodating hard real-time streams with deterministic end-to-end delays. TAS provides latency guarantees by using a pre-defined scheduler that guarantees timely transmission of Control Data Traffic (CDT). Furthermore, the existence of a mechanism is assumed that ensures that non-CDT traffic will not interfere with TAS traffic. TAS defines a fixed and periodic schedule of time intervals, which specifies when a traffic queue is enabled (opened) or disabled (closed) for transmission. This schedule is configured offline for each port of the switch.

As it is a new field of study, there are few research contributions in the domain of Time Sensitive Networks. One of the first papers in the domain is [12], which presents three traffic shapers which were under discussion in the IEEE802.1TSN working group, at the time when the paper was being written (autumn 2014): Time Aware Shaper (TAS), Burst Limiting Shaper (BLS) and Peristaltic Shaper (PS). Analyses are proposed for each of them, to compute the end-to-end latency of a frame over n hops. Evaluations are performed showing that: (1) only TAS is able to schedule the Control Traffic within the maximal delays imposed by the standard; (2) BLS has delays larger by approximately 20% with respect to the limits; (3) PS has delays almost twice as large than those permitted by the standard. An improvement is proposed to the Peristaltic Shaper in the form of a Guard Band which reduces the delays, but not enough to make them less than what is permitted by the standard.

In the same period [13] proposed a formal worst-case analyses for the Time-Aware Shaper (TSN/TAS) and Peristaltic Shaper (TSN/PS). For each shaper two analyses are derived: (a) one analysis for the streams of frames that are controlled by the shaper itself and (b) one analysis for the streams that are not controlled by the shaper (the shaper also has an impact on these streams in the form of

transmission delays). In the case of TSN/TAS it is considered that critical streams have link access (i.e. their gate is opened) during special time intervals that repeat periodically and in which no other stream is allowed to transmit. All non-critical streams transmit outside of these TAS intervals and need to compete for access. This assumptions is quite restrictive on the possible scheduling table that can be used on the network. Also, the credit of non-TAS streams is not considered, i.e. there is always enough credit to transmit.

More recently [5] proposed an analysis to compute fully deterministic schedules (i.e. TAS scheduling tables) for 802.1Qbv-compliant (i.e. TSN) multi-hop switched networks, while identifying functional parameters that affect the communication behaviour. Several methods are presented for computing the static schedules, either by use of Satisfiability Modulo Theories (SMT) or by Optimization Modulo Theories (OMT). Two isolation methods are presented: (a) flow isolation and (b) frame isolation, in order to isolate critical traffic from non-critical (BE) trafic. The flow isolation method makes use of the Time Aware Shaper to close the logical gates through which BE traffic is sent while the gates which transmit critical flows are opened. The frame isolation method allows for interleaving of flows in a deterministic manner such that BE frames do not interfere with critical frames and in this way the egress port can be utilised more efficiently. The hypotheses necessary for the frame isolation method to be applicable are quite stringent: no frame loss, constant payload size, no dropped frames, etc, making the solution quite inapplicable as these situations may easily occur in practice.

In our approach we make use of the *Eligible Interval* analysis proposed in [4] and [3] which we extend to make it applicable to the TSN model. The first contribution [4] presents the *Eligible Interval* analysis for AVB frames, in the case of either lower-priority interference or higher-priority interference upon the frame under analysis – the case of both types of interference is left as future work. An eligible interval is an interval in which a stream has pending load available and also has non-negative credit, i.e. the frames can be sent unless the output port is otherwise occupied. The analysis returns the worst-case response time, which is a tighter upper-bound with respect to the one returned by the busy period analysis technique [2].

Subsequently, [3] extends the eligible intervals analysis presented in [4] to take into account interference from lower- and higher-priority traffic at the same time. The analysis does not rely on any assumptions on interfering priority classes other than those enforced in the Ethernet AVB standard [7], i.e. there is no need to know the arrival patterns or transmission requirements of the interfering traffic. This independence from detailed information of the traffic is achieved by taking advantage of the AVB characteristic, and in particular of the effect of credit based shaper on the interfering traffic, i.e. replenishing and consumption slopes of the credit. The *Eligible Interval* analysis was proved to be tight for AVB networks, and for this reason we choose to extend it to the TSN case.

Also related to the TSN research topic, [1] presents a model very similar to the IEEE 802.1Qbv TSN, called AVB ST (AVB Scheduled Traffic). The main difference between TSN and AVB ST lies in the way protected windows are created for the CDT traffic. While in TSN a scheduling table is used to define which traffic-gates are

closed or opened at each moment in time, AVB ST does not use such a table. Instead AVB ST considers that each CDT frame (called scheduled traffic in this approach) is scheduled in a deterministic manner and the switch has complete knowledge of when each frame arrives and needs to be transmitted. Hence each CDT frame has its own protected window and implicitly a guard band before its arrival. Another important difference between AVB ST and TSN is the fact that in TSN the credit of classes AVB_A and AVB_B does not increase when their transmission gates are closed by the time aware shaper, while in AVB ST their credit still increases (according to the relevant *idleSlope*) when ST frames are being transmitted. This means that the *Eligible Interval* analysis of [4] and [3] can not be applied on AVB ST as the arrival and transmission of an ST message allows for the credits of AVB streams to evolve and may in this way change their transmission order. Hence [1] uses the busy period approach, leading to more pessimistic results than can be obtained with the *Eligible Interval* approach.

3 TSN MODEL AND NOTATIONS

In this section we describe the model of the network we are interested in analyzing as well as the different types of traffic that can be transmitted on the network. We are interested in Time Sensitive Networks (TSN) embedded in the automotive domain. The network is meant to convey Control Data Traffic (CDT), Audio/Video traffic (AVB) and Best Effort traffic (BE). Messages (i.e frames) are sent from end-nodes on the network through one or multiple switches until they reach their destination nodes. Figure 7 presents an example of such a network with three switches and twelve nodes. Table 2 gives details of the ECUs together with the traffic classes they transmit/receive. It extends the model of [12] and has been defined within the Eurostars RETINA project according to the automotive application requirements.

3.1 Network Model

On the egress port of each switch the TSN standard specifies that there are up to eight gates, meaning that the traffic that passes through a switch can be split into at most eight different classes or priorities. At least one of the eight gates is reserved for the control data traffic (CDT). The gates that are not reserved for the CDT are available to be used by the AVB and BE traffic. The difference between the CDT and other traffic is explained in detail below. A graphical depiction of an egress port can be seen in Figure 1. The port also has a Schedule table implemented with the purpose to reserve bandwidth for the CDT and isolate it from the AVB and BE traffic by using a guard band. When a gate is opened (marked as 1 in the schedule table) then the traffic queued up in the gate can be transmitted, else, if the gate is closed (marked as 0 in the table) then transmission is blocked. Except the last ongoing transmission which continues to transmit and it finishes before the end of the guard band. AVB traffic is further restricted using CBS (detailed in Section 3.2.2) for fairly sharing the remaining bandwidth among concurrent streams other than CDT.

In this paper we assume transmission without frame preemption and we only focus on the delay analysis through one egress port. The analysis of the end-to-end delay of a flow crossing several switches is out of the scope of the present paper.
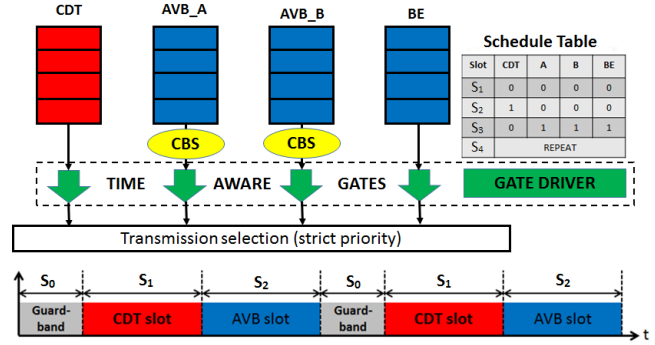

Figure 1: Egress port

3.2 Traffic model

According to the TSN standard, traffic on the network is classified in four classes: CDT (control data traffic), A (Audio), B (video) and BE (best-effort). The CDT class is the novelty that the TSN standard [9] brings with respect to the AVB standard [7] on which it builds upon.

3.2.1 Control Data Traffic. The CDT class is meant to contain flows of traffic of a critical nature, with strict timing constraints, necessary for the functioning of the automotive in which the network is embedded. For this reason the TSN standard specifies shaping mechanisms in order to isolate the CDT class from all other traffic and to guarantee that the imposed delays are respected. The shaping mechanism that is emerging as the best candidate is called Time Aware Shaper (TAS) and it works by dividing the access to the link in a TDMA fashion in order to isolate the CDT traffic in its own time slot(s) such that AVB traffic can not interfere with it.

CDT traffic is considered to be deterministic as it can be controlled on the sending node. Each output port (also called egress port) of each TSN switch has a TAS implemented in the form of a scheduling table which indicates the moments in time when each gate (different classes of traffic pass through different gates) is closed or opened. In order to isolate CDT streams from all other traffic, the TAS scheduling table necessarily contains at least one time slot when only the gate for the CDT streams is opened and all other gates are closed. CDT messages arrive at the switch inside a CDT time-slot (called a protected window in the TSN standard) and are immediately sent without being blocked by AVB traffic. Before each CDT time-slot the scheduling table specifies a guard-band (GB), a time-slot in which AVB frames can not start transmission, but they can finish transmission if they started being sent before the guard-band was activated. This mechanism is necessary just before the CDT time-slot is activated to ensure that no other frame would block the transmission of CDT frames as they would need to wait for the AVB or BE messages to finish transmission. The size of the guard band is equal to the size of the largest frame (over all classes of traffic except CDT) that may pass through the switch. This is in order to ensure that any frame that begins transmission just before that guard band is activated, will finish transmission before that CDT-slot begins and so will not block the transmission of CDT frames. The CDT-slot together with the GB form the Protected Window.
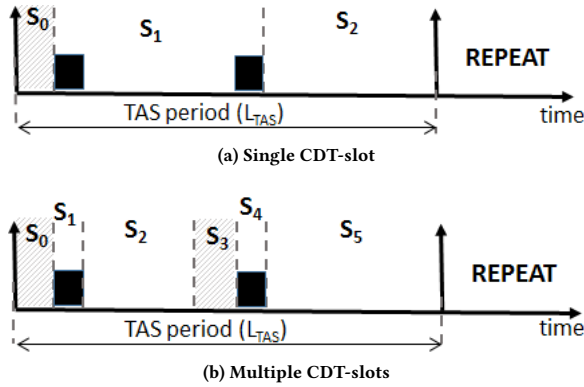
(a) Single CDT-slot

(b) Multiple CDT-slots

Figure 2: Different TAS configurations

Slot	CDT	A	B	BE
S_0	0	0	0	0
S_1	1	0	0	0
S_2	0	1	1	1
REPEAT				

(a) Single CDT-slot

Slot	CDT	A	B	BE
S_0	0	0	0	0
S_1	1	0	0	0
S_2	0	1	1	1
S_3	0	0	0	0
S_4	1	0	0	0
S_5	0	1	1	1
REPEAT				

(b) Multiple CDT-slots

Table 1: Different TAS scheduling tables

Setting up the scheduling tables for each TSN switch is a complex problem and a gross-grain solution for it was proposed in [6]. In this work we consider that the scheduling tables for each switch are given.

A TAS scheduling table is a list of time points, indicating which gate is open at each time point, with the last entry in the list being the *REPEAT* command. The scheduling table has a period L_{TAS} which is split into different time slots in a TDMA fashion and this splitting repeats for the entire life-time of the system. The TSN standard specifies that $L_{TAS} = 500 \ \mu s$ by default.

We distinguish between two types of TAS configurations, depending on the granularity of their scheduling tables:

(1) With a single Protected Window, depicted in Figure 2a,
(2) With multiple Protected Windows, depicted in Figure 2b.

Both TAS schedules of Figure 2 need to transmit two CDT messages, represented by the black boxes, but a different amount of bandwidth is reserved in the two cases. The TAS scheduling tables that give rise to the two configurations are presented in Table 1

In Figure 2a the TAS splits time into just three slots: (1) S_0 is the guard band necessary before each CDT-slot, (2) S_1 is the CDT-slot and (3) S_2 is the bandwidth left for the AVB and BE traffic. This kind of TAS has the advantage that it is easy to derive and implement as all CDT frames are grouped into a single block of time. The potential disadvantage is the fact that the bandwidth between CDT frames is wasted, as nothing transmits in that time interval. The schedule table for this TAS configuration can be seen in Table 1a.

As opposed to the single Protected Window configuration of Figure 2a, the TAS schedule of Figure 2b depicts the more complicated scheduling Table 1b, which splits time in several slots: (a) two guard-bands (S_0 and S_3), (b) two CDT-slots (S_1 and S_4) and (c) two slots left for the transmission of AVB and BE traffic (S_2 and S_5). In this way the TAS only reserves as much bandwidth as necessary for the transmission of CDT frames, freeing up bandwidth to be used by AVB and BE traffic (slot S_2 is gained with respect to the schedule in Figure 2a. Of course this kind of scheduling table can be generalized to an arbitrary number of protected windows not just two. The disadvantage of this kind of configuration is the potentially large complexity of the scheduling table, which increases with the number of CDT flows in the network. Deriving such a scheduling table is not trivial. In addition, each protected window also reserves bandwidth for the guard-band preceding the CDT-slot, which is a potential waste of bandwidth as AVB and BE flows may not always be able to claim it for their transmissions. Also, it may not be worth the effort of making a complex table if the time between CDT frames is small and it is entirely reserved by the guard-band, completely removing the possibility that it is used by AVB or BE flows.

3.2.2 AVB Traffic. The AVB traffic is transmitted outside of the protected windows which reserve time-slots for the CDT class. Besides being blocked by the time aware shaper, AVB traffic is also influenced by several other mechanisms:

- classes AVB_A and AVB_B are shaped by the Credit Based Shaper (CBS) specified by the AVB and TSN standards. This shaping mechanism is necessary in order to avoid bursts of messages of the A and B class arriving all at the same time, i.e. to impose a minimal separation between messages of each stream. As an effect of credit based shaping, bandwidth is left available for best effort (BE) streams.
- as in each gate there are messages of different streams (of the same class), these messages are dispatched in FIFO ordering of their arrival at the egress port.
- when messages of different classes are eligible for transmissions (i.e. have positive credit), then conflicts are resolved according to the fixed priority policy given by the AVB standard which specifies that class A has higher priority than class B which has higher priority than class BE.

Hence the transmission of an AVB flow is influenced by four mechanisms: TAS, CBS, FIFO and FPNS (fixed priority non-preemptive scheduling).

Credit Based Shaping mechanism: Each gate that transmits AVB_A or AVB_B traffic has a CBS mechanism implemented. Streams of class $i \in \{AVB\_A, AVB\_B\}$ queued on the output port are allowed to start transmission when the credit of the class is larger or equal to 0. When class i transmits, its credit decreases at a rate of α_i^- which is called the *sendSlope* of class i and is measured in bits per second. Alternatively, the credit of the class may increase when either its credit is negative (and it is not currently transmitting), or when there are messages of class i queued in the gate but can not transmit because a class of higher priority is transmitting - in this case the credit of class i may increase even if it is already larger than 0. The increase rate of the credit of class i is denoted by α_i^+, called *idleSlope*. A relation between α_i^+ and α_i^- can be established as $\alpha_i^+ + \alpha_i^- \leq BW$, where BW is the transmission rate of the output port (also called the bandwidth of the port). The *idleSlope* (α_i^+) represents the desired

bandwidth reservation for the class and it can not be larger than the total bandwidth BW of the port, i.e. $0 < \alpha_i^+ \leq BW$. The *sendSlope* can then be set as $\alpha_i^- = BW - \alpha_i^+$.

TAS influence on the CBS: The IEEE TSN standard specifies that when a gate which transmits AVB traffic is closed (i.e. during the guard-bands and the CDT-slots) the credit of the traffic in that gate can no longer increase. Also, as AVB streams are allowed to transmit during the guard-band (finish transmitting of frames that had started before the activation of the guard-band), then the credit of that class does decrease during the guard-band. These effects are represented graphically in Figure 3 (best viewed in color on a screen), where two AVB_A flows, one AVB_B flow and two BE flows are being transmitted in a TSN switch (In this case switch SW1 of Figure7). The credit of the AVB_A class is represented by the black line while the credit of the AVB_B class is represented by the red line. Frames of class AVB_A and BE are available for transmission at $t = [0, 125, 250, 500, 625, 750]$, frames of class AVB_B are available for transmission at $t = [0, 250, 500, 750]$, the guard-band is activated at $t = [60, 560]$ and the CDT-slot is activated at $t = [86, 586]$. Green lines mark the moments in time when frames are available to be transmitted. The *idleSlope* and *sendSlope* of the two AVB classes are given in the last two columns of Table 3. All frames have an equal size of 26μs. The guard-band also has a size of 26μs as this is the size of the largest AVB and BE message in our example[2]. The size of the CDT-slot is equal to 150μs.

It can be seen that at $t = 52$, stream AVB_A2 starts transmitting a frame, and it continues transmitting until $t = 78$ with the guard-band being active since $t = 60$. Between $t = 60$ and $t = 78$ the credit of class AVB_A continues to decrease. Once the frame finishes its transmission, then the credit of class AVB_A stays constant until the end of the CDT-slot. As opposed to class AVB_A, class AVB_B has negative credit at $t = 52$, hence its credit is incremented according to its *idleSlope*, until $t = 60$ at which moment its gate is closed due to the activation of the guard-band. The credit of class AVB_B stays constant during the guard band and the CDT-slot, even though it has a negative value. To resume, the credits of AVB classes can decrease during the guard-band but not increase, and it can neither increase nor decrease during the CDT-slot, where it stays at constant values.

Note that at $t = 366$ and at $t = 444$ the credit of class AVB_A drops from a positive value to zero as at those time instants there are no AVB_A frames waiting to be transmitted. This is a characteristic of the IEEE AVB [7] standard regarding credit base shaping and it still applies to TSN switches.

A further effect of the TAS on AVB flows that leads to their worst case transmission delays is presented in detail in Section 4.1 (particularly Figure 5) hence we omit it here and refer the reader to Section 4.1.

To the best of our knowledge this is the first complete description, both textual as well as graphical, of the scheduling of AVB frames in TSN switches and the effects of time aware shaping on the credits and transmission of AVB flows.

---

[2]In general the length of the guard-band corresponds to the size of the largest non-CDT message in the switch. This value is upper-bounded by the time needed to transit one Ethernet MTU frame of size 1534 bytes (e.g. 122,72μs in a 100Mbps switch) without preemption and lower-bounded by the time needed to transmit two Minimal Ethernet frames of 64 bytes (e.g. 10,24 μs on a 100Mbps switch) with preemption

3.3 Problem Description

The problem that we address in this paper can be expressed as follows: given a TSN switch and the traffic passing through it, compute the worst case delay of messages of AVB classes A and B in the switch, knowing that their transmission times are influenced by the two shapers simultaneously (TAS and CBS) as well as by the interference and blocking from other AVB messages of higher and lower priority (FPNS) and by messages of same priority as the message under analysis (FIFO). To the best of our knowledge, up to date there exists no analysis technique to compute the worst-case transmission delay of an AVB message in a TSN network.

4 LOCAL ANALYSIS FOR AVB STREAMS IN TSN SWITCHES

In this section we introduce the main contribution of our work, the formal worst case delay analysis for an AVB message in a TSN switch. For this we extend the eligible interval analysis of [4] and [3] to account for possible blocking created by the TAS. We first start with the case when there is a single protected window (i.e. a single CDT slot and its preceding guard-band) and then generalize the result to the case of multiple protected windows.

Before we proceed with the proposed analysis, we briefly recall here the worst case delay analysis in the case of AVB networks as described in [3]. The analysis computes the worst case delay of a frame τ_i using the following equation (from [3]):

$$R_{AVB}(\tau_i) = R_{FIFO}(\tau_i) + C_L^{max} \times (1 + \frac{\alpha_H^+}{\alpha_H^-}) + C_H^{max} \qquad (1)$$

where $R_{FIFO}(\tau_i)$ is the delay due to the FIFO ordering of messages of the same class on the gate through which the message under analysis is transmitted, computed as follows (from [3]):

$$R_{FIFO}(\tau_i) = C_i + \sum_{\tau_j \in eqp(i), \ j \neq i} C_j \times (1 + \frac{\alpha_H^-}{\alpha_H^+}) \qquad (2)$$

In Equations (1) C_L^{max} represents the maximum transmission time of any message over all flows of lower priority than the flow under analysis τ_i and is used as a blocking time caused by a lower priority message due to the fact that transmission is assumed non-preemptive (note that we are only interested in non-preemptive functioning of TSN in this paper). C_H^{max} represents the maximum transmission time of any message over all flows of priority higher than the flow under analysis. In Equation (2), $eqp(i)$ represents the set of flows of the same priority as the flow under analysis. α_H^+ and α_H^- are the *idleSlope* and respectively *sendSlope* of the higher priority class H.

In order for Equation (1) to be applicable, class i needs to have bounded transmission delays which is ensured if its bandwidth reservation respects the following utilization condition:

$$\sum_{eqp(i)} \frac{C_i}{T_i} \leq \frac{\alpha_i^+}{\alpha_i^+ + \alpha_i^-} \qquad (3)$$

Cao et al. [3] proved that Equation 1 gives a tight bound on the worst-case interference that a message suffers in an AVB switch. A generic worst case scenario is depicted in Figure 6 where the
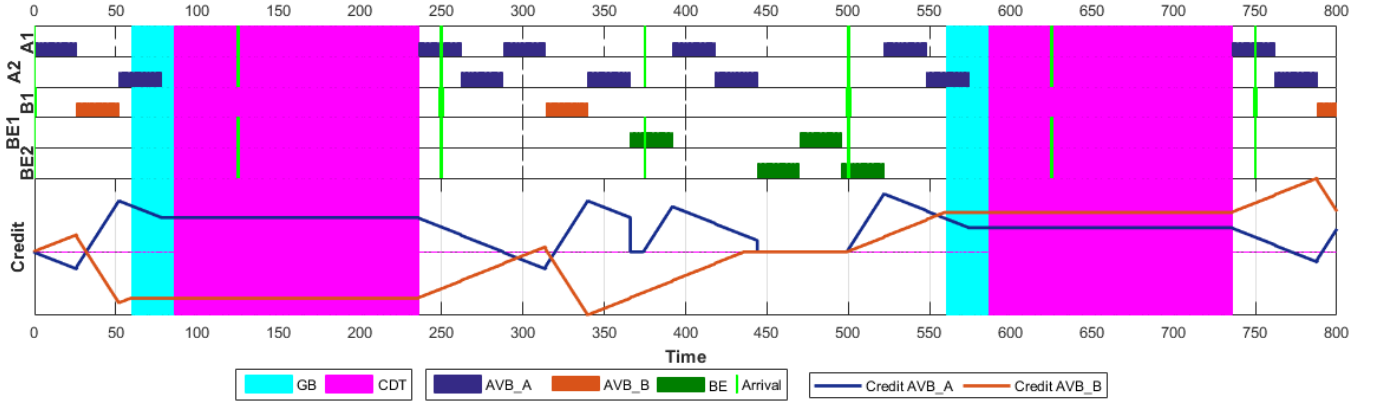
Figure 3: The effect of TAS on credit consumption and replenishment

transmission of message M is delayed by messages of the same priority in the FIFO queue through which it passes, then by a lower priority message L and then by several higher priority messages of class H, leading to Equation 1.

Once the transmission time of an AVB message is computed we can then include the further blocking time that the message can suffer due to the time aware shaper of the TSN switch.

4.1 Analysis in the case of a single Protected Window per TAS cycle

Let us denote by L_{CDT} the length of the CD-slot. That is, the amount of time during which the entire bandwidth of the link is reserved for the transmission of CDT messages.

Similarly, let us denote by L_{GB} the length of the Guard Band preceeding the CDT-slot. That is, the amount of time during which the entire bandwidth of the link is reserved in order to protect the transmission of CDT messages. Note that the size of the Protected window is equal to $L_{CDT} + L_{GB}$.

As the credits of AVB streams do not increment when their gates are closed, i.e. neither during the CDT slot, nor during the guard band slot, then when a protected window occurs it freezes the credits of AVB flows. Subsequently, once the protected window has passed, the transmission of AVB messages can continue from where they left off.

Definition 4.1. We define the bandwidth reservation BR_i of a class $i \in \{\text{AVB\_A, AVB\_B}\}$ in a TSN switch as the proportion of bandwidth (BW) of a link that is given to class i through its credit replenishment and consumption slopes, after we have subtracted the portion of bandwidth reserved for the Protected Window:

$$BR_i = \frac{\alpha_i^+}{\alpha_i^+ + \alpha_i^-} \times (1 - \frac{L_{CDT} + L_{GB} + recover}{L_{TAS}}) \qquad (4)$$

where L_{TAS} is the TAS cycle duration and

$$recover = C_i^{max} \times \frac{\alpha_i^-}{\alpha_i^+} \qquad (5)$$

is the amount of time needed for the credit of class i to recover to zero in the worst case, with C_i^{max} being the largest transmission
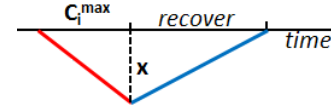

Figure 4: Recover to non-negative credit

time among all messages of class i. We note that C_i^{max} is upper bounded by L_{GB}. The amount of time *recover* comes from the minimal credit value that the class can reach, which depends on the maximal length of any message of the class. A graphical representation of this process is depicted in Figure 4. The minimal credit that the class can reach is denoted in the figure as x. Knowing that $x = C_i^{max} \times \alpha_i^-$ and also that $x = recover \times \alpha_i^+$ we can deduce that $recover = C_i^{max} \times \frac{\alpha_i^-}{\alpha_i^+}$.

Feasibility condition: In order to guarantee that AVB messages have finite worst-case transmission times in a TSN switch, a sufficient condition is that the utilization of the class is less than its bandwidth reservation BR_i:

$$\sum_{eqp(i)} \frac{C_i}{T_i} \leq BR_i \qquad (6)$$

If the feasibility condition is respected, then the worst case delay of an AVB flow τ_i in a TSN switch can be computed using the following equation:

$$R_{TSN}(\tau_i) = R_{AVB}(\tau_i) + L_{CDT} + L_{GB} \qquad (7)$$

where $R_{AVB}(\tau_i)$ is obtained using Equation (1).

This is a conservative upper-bound as it supposes that the message under analysis needs to wait for both the GB and CDT slots to pass before it can be transmitted, in addition to its worst case AVB delay.

Equations (4), (5), (6) and (7) provide a sufficient analysis framework for computing AVB delays in TSN switches and to validate the schedulability of the system.

Equation (4) removes from the total bandwidth of the output port the bandwidth that is blocked by the TAS for the Protected Window and also the bandwidth necessary for the class under analysis to recover to non-negative credit (Equation 5). A fraction of the credit that is left is reserved for class i giving the value of
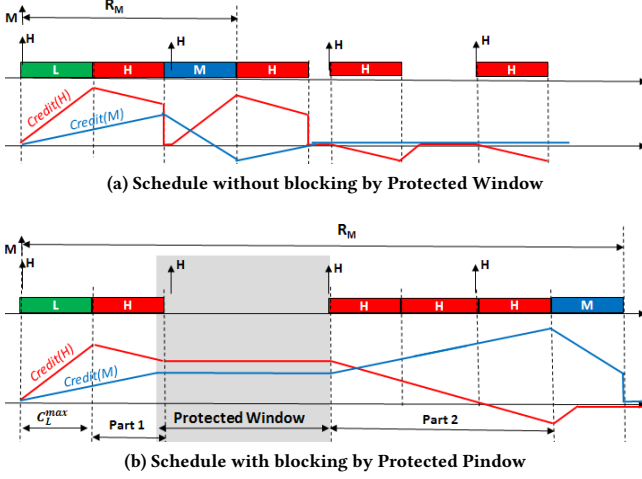
(a) Schedule without blocking by Protected Window

(b) Schedule with blocking by Protected Pindow

Figure 5: The effect of a TAS Protected Window on frame transmissions

BR_i. This reservation is similar to the reservation in AVB switches (Equation 3), where a fraction of the *total* bandwidth is reserved for an AVB class i. By also removing the fraction of bandwidth necessary for the class to recover to non-negative credit and by verifying that the feasibility condition given by Equation (6) is respected, we ensure that there are no frames of class i that are backlogged from one TAS period to the next, which would increase the load in each subsequent TAS period until messages would fail to meet their transmission delays, i.e. their transmission times would be unbounded if Equation 6 is not respected.

Any message of class i can not be blocked by two CDT or GB slots as the utilization condition (6) ensures that enough bandwidth is allocated to the stream so that messages are transmitted inside the same TAS period in which they become available for transmission. In other words, if Equation (6) is respected, the delay of any message of class i in the switch is upper-bounded by L_{TAS}, i.e. $R_{TSN}(\tau_i) \leq L_{TAS}$. This is an upper-bound on the delay of an AVB flow, but we note that Equation 7 is still necessary as the deadline of the flow may be much smaller than L_{TAS} and computing the actual deadline is necessary in order to see if it does indeed respect the deadline.

Worst case transmission delay in the TSN context: An effect of the TAS that leads to the worst case transmission delay of an AVB class can be seen in Figure 5, where the relative order of transmission is changed due to the Protected Window. Let us consider the transmission of the frame denoted by M. In Figure 5a, frame M is transmitted after a lower priority frame L and a higher priority frame H. In Figure 5b, the Protected Window is activated right at the end of the first H frame, blocking in this way the transmission of frame M. Furthermore, several H-frames become available for transmission during the Protected Window, and as they have higher priority than frame M they get transmitted right after the Protected Window, further blocking the transmission of frame M and changing the relative transmission order with respect to Figure 5a.

Nevertheless, the amount of H frames that can block the transmission of frame M is limited by the *idleSlope* and *sendSlope* of stream H, due to the fact that the credit of class H can not increase during the Protected Window. That is, regardless of how many H
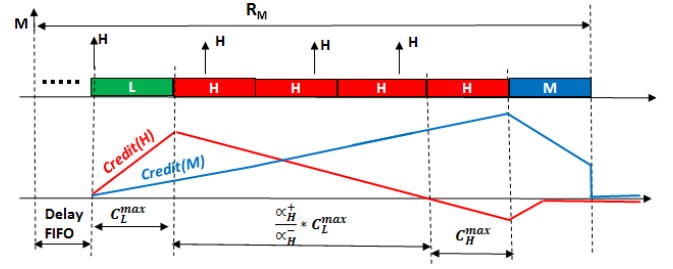

Figure 6: Worst case delay of a frame when there is no Protected Window.

frames are ready for transmission at the end of the protected window, only a subset of these can be transmitted before their credit becomes negative. An intuition of this upper-bound on H frames can be seen in Figure 6, which is the worst case transmission delay for frame M when there is no Protected Window. The credit of stream H can reach a maximal value of $\alpha_H^+ \times C_L^{max}$ and then it decreases at a rate of α_H^- requiring $\frac{\alpha_H^+ \times C_L^{max}}{\alpha_H^-}$ time units before reaching zero, at which point stream H can still send one more frame. This scenario leads to the worst case delay of frame M in an AVB switch, given by Equation 1. As the Credit of class H does not increase during the Protected Window, this upper-bound on the maximal blocking by higher priority streams over stream M continues to hold true, leading to a worst case transmission delay of an AVB frame in a TSN switch given by Equation 7. We note that the worst case scenario is more likely to occur in a TSN switch than it is to occur in an AVB switch as the TAS will lead to bursts of higher priority frames to be transmitted right after the Protected Window, blocking lower priority streams and resulting in a worst case transmission delay where otherwise it wouldn't be the case, as depicted in Figure 5. Also note that in Figure 5b the sum $Part1 + Part2$ is at most equal to $\frac{\alpha_H^+ \times C_L^{max}}{\alpha_H^-} + C_H^{max}$ of Figure 6 which leads to Equation 7 by adding $L_{CDT} + L_{GB}$ to Equation 1.

4.2 Generalization to arbitrary number of Protected Windows per TAS cycle

The case of multiple Protected Windows is more complex to analyze due to the possible inter-leaving of AVB and CDT reservations of the bandwidth.

We denote by L_{CDT}^k the length of the k-th CDT slot, in the set of n CDT slots that my occur during the entire TAS cycle. Each such CDT slot is preceded by a guard band of length L_{GB}, i.e. all guard bands are of equal lengths.

A conservative upper-bound on the worst-case delay of an AVB message in a TSN switch can be obtained by generalizing Equation (7) in the following way:

$$R_{TSN}(\tau_i) = R_{AVB}(\tau_i) + \sum_{k=1}^{n} L_{CDT}^k + n \times L_{GB} \qquad (8)$$

This equation holds under the generalized utilization condition:

$$\sum_{eqp(i)} \frac{C_i}{T_i} \leq BR_i^n \qquad (9)$$

where

$$BR_i^n = \frac{\alpha_i^+}{\alpha_i^+ + \alpha_i^-} \times (1 - \frac{n \times L_{GB} + \sum_{k=1}^{n} L_{CDT}^k + recover}{L_{TAS}}) \quad (10)$$

and *recover* is computed using Equation (5).

In order to compute this upper-bound we need to know the length of each CDT slot in the switch. Still, this upper-bound is pessimistic as it supposes that all protected windows arrive immediately one after the other, which is probably not the case in well designed networks, i.e. there are gaps between protected windows in which AVB messages can be transmitted. The analysis can be refined since not all flows will be blocked by all protected windows. Indeed, some higher priority streams may suffer less blocking from TAS, while lower priority flows will potentially be blocked by all protected windows in a TAS period. Distinguishing the exact amount of TAS blocking that a flow may suffer is a non-trivial problem and we leave it as future work.

The analysis framework given by Equations (8), (9) and (10) is a direct generalization of the case when there is a single Protected Window in the TAS cycle (Section 4.1), and hence it follows the same reasoning of correctness.

5 EXPERIMENTAL RESULTS

In this section we further validate our delay analysis for AVB traffic in TSN switches by applying it on an automotive-inspired TSN system which is simulated and the observed delays are compared with the the delays computed using our theoretical analysis.

The topology of the network is presented in Figure 7. It has three switches and 13 end-nodes. The details of each node and the type of traffic they generate or receive are presented in Table 2. Figure 7 also presents the sources, destinations and paths that each stream takes through the network.

The default parameters of each stream are presented in Table 3. The period (T_i) and end-to-end deadline (D_i^{global}) of each stream are specified in the IEEE TSN standard. The local deadline (D_i^{local}) is computed by dividing the end-to-end deadline to the number
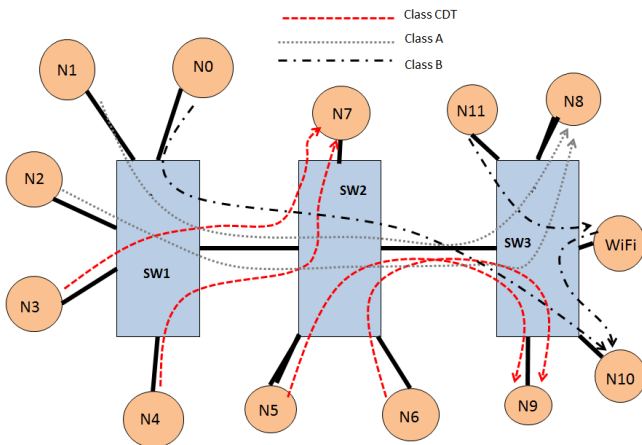

Figure 7: Network topology

Node	Node Type	Stream Type	I/O
N0	Back Camera	Video (Class B)	I
N1	Voice Assistance	Audio (Class A)	I
N2	Media Audio	Audio (Class A)	I
N3	Speed Sensor	CDT	I
N4	Angular Rotation Sensor	CDT	I
N5	Wheel Angle Sensor	CDT	I
N6	Steering Wheel Sensor	CDT	O
N7	Engine Speed Actuator	CDT	I
N8	Sound Output Device	Audio (Class A)	I
N9	Steering Wheel Actuator	CDT	I
N10	Video Output Device	Video (Class B)	I
N11	Front Camera	Video (Class B)	I
WiFi	Wireless module	AVB	I/O

Table 2: Stream details

Class	$Size_i$	C_i	T_i	D_i^{local}	D_i^{global}	α^+	α^-
CDT	175B	$14\mu s$	$500\mu s$	$20\mu s$	$100\mu s$/5hops	na	na
A	325B	$26\mu s$	$125\mu s$	$285\mu s$	2ms/7hops	80	20
B	325B	$26\mu s$	$250\mu s$	$7142\mu s$	50ms/7hops	20	80
BE	325B	$26\mu s$	$125\mu s$	na	na	na	na

Table 3: Traffic classes

of hopes it is specified upon. Other ways for computing the local deadlines exist, for example taking into consideration the load on each switch through which the stream passes, but for our example it suffices to divide the end-to-end deadline to the number of hops.

The size parameter ($Size_i$) of a stream is made up of the payload of a message of the stream plus its header, i.e. $Size_i = Payload_i + Header_i$. The transmission time C_i of a frame of stream i is computed by dividing the size of the stream (converted in *bits*) to the port transmission rate (BW) of the TSN switch:

$$C_i = \frac{Size_i \times 8}{BW} = \frac{(Payload_i + Header_i) \times 8}{BW} \quad (11)$$

For example, in Table 3, a transmission rate of 100 Mbps is considered, resulting in the transmission times C_i presented in the third column of the table. For simplicity, all throughout the section we reason in terms of size, including in this way the payload and header. The *sendSlope* α^- and *idleSlope* α^+ of classes AVB_A and AVB_B (measured in Mbps) are presented in the last two columns of the table.

The two CDT streams are separated by $100\mu s$ on the sending nodes, so that they are never available for transmission to block each-other. That is, if the CDT stream of node N_3 has a frame available for transmission at instant t, then the CDT stream of node N_4 will have a frame available for transmission at instant $t + 100$.

For our investigation we use two TAS configurations, as presented in Table 1. The period of the TAS is $L_{TAS} = 500\mu s$ in both cases.

- The first configuration has a single Protected Window composed by a CDT-slot of size $L_{CDT} = 150\mu s$ (slot S_1 in Table 1a), preceded by a guard-band of size $L_{GB} = 26\mu s$ (slot S_0 in Table 0a). This leaves $500 - 150 - 26 = 324\mu s$ available for the transmission of AVB and BE traffic (slot S_2 in Table 1a).

- The second configuration has two CDT-slots of equal sizes $L_{CDT} = 14\mu s$ (slots S_1 and S_4 in Table 1b), each CDT-slot being preceded by a guard-band of size $26\mu s$ (slots S_0 and S_3 in Table 1b). Due to the separation of $100\mu s$ of the two CDT streams, the two Protected Windows are spaced out so that there are $60\mu s$ between them, available for AVB and BE traffic (slot S_2 in Table 1b). Slot S_5 in the scheduling Table 1b, of size $386\mu s$ is also available for the transmission of AVB and BE traffic.

All AVB and BE streams are first instantiated at $t = 0$ and then subsequently instantiated according to their periods T_i. All throughout our experiments we have varied the starting time of the TAS by giving it an offset (let us denote it by ϕ) that varies between 0 and $L_{TAS} = 500\mu s$ in increments of $1\mu s$. This means that slot T_0 starts at $t = \phi$, then slot S_1 starts after the previous slot has passed, i.e. at $t = S_0 + \phi$, and so on. For each value of the offset, a simulation of $100000\mu s$ was performed. Hence, each experiment contains 500 simulations, one per each value of ϕ and the largest observed delay over all simulations was recorded. Offsetting the TAS with respect to the first arrivals of non-CDT traffic allows for some AVB and BE frames to be transmitted before the first guard-band is activated, modifying the credits accordingly, leading to larger delays for later frames as credits are already negative at the instant when the guard-band is activated. The simulator used is a prototype that we have implemented based on the run-time simulator PAnSim [11] and that we make available for free use to the community[3].

5.1 Default experiment

As a first experiment we simulate the network depicted in Figure 7, at a transmission rate of 100Mbps and with 9 flows crossing the network: four CDT flows passing through the protected windows formed by the TAS, two AVB_A flows, one AVB_B flow and two BE flows. Except for the BE flows, all flows are depicted on the network in Figure 7. The BE flows are not depicted in the figure as they are broadcasted in the entire network, passing through all switches and reaching all end-nodes. As an example of the kind of results that our local delay analysis provides, we present in Table 4 the results for switch SW1. The network was first simulated in an AVB setting, i.e. no CDT traffic, hence no bandwidth blocked by protected windows. The worst case delays for the AVB setting were also calculated with the eligible interval technique presented in [4] and [3]. Then the protected windows were activated to enhance the network to transmit CDT flows in the two TAS configurations discussed in the previous section. The simulated delays are then compared to the delays computed using our proposed analysis in Section 4.1 and Section 4.2 for the two TAS configurations respectively. As it can be seen in Table 4 the delays computed by our analyses are strictly larger than the delays observed during simulation, yet very close even in the case when these are two CDT-slots. This indicates that the analysis is safe and tight, being able to correctly calculate the delay that a frame of a given flow will suffer in the switch. For example, for class AVB_A (streams A1 and A2) in the case of a single Protected Window, our analysis computes a worst case delay on switch SW1 equal to $261\mu s$ while the largest delay we observe during simulation for the stream is $260\mu s$. In the case of

two Protected Windows, for the same class our analysis computes a delay of $165\mu s$ while the largest delay observed during simulation is just $124\mu s$. The difference between the two delays is $41\mu s$ which is about the size of a protected window, i.e. $L_{GB} + L_{CDT} = 26 + 14 = 40\mu s$. This indicates that the stream is not affected by the second Protected Window in the TAS cycle, denoting the pessimism of our generalized analysis. We leave for future work the decrease of this pessimism.

Using our local analysis we can compute a bound for the end-to-end delays of the streams by applying the analysis on each switch that the stream traverses. For example, for stream A2, for the case of a single Protected Window, the analysis computes an end-to-end delay of $574\mu s$ while during simulation the largest delay observed is equal to $324\mu s$. Similarly in the case of multiple Protected Windows the analysis computes an end-to-end delay of $382\mu s$ while the largest simulated delay is equal to $198\mu s$. In both cases we notice an increased amount of pessimism which was to be expected as our analysis is a local one, hence it assumes an arrival pattern that leads to the worst case blocking on each switch through which the stream passes. This assumption is not necessarily true in all switch as frames get serialized from one switch to the other resulting in less blocking and smaller end-to-end delay with respect to what a local analysis can compute. To eliminate this pessimism an end-to-end analysis needs to be proposed, for example a solution based on the trajectory approach [10]. This is one of our future objectives.

5.2 Extended experiment

In order to test the generality of our analysis, for the second experiment we have extended the amount of traffic in the network while keeping the same topology. In order to accommodate more flows we increase the transmission rate of each switch in the network to 1Tbps. The consumption and replenishment slopes are adjusted accordingly: $\alpha_A^+ = 800$Mbps, $\alpha_A^- = 200$Mbps, $\alpha_B^+ = 200$Mbps and $\alpha_B^- = 800$Mbps. For the sake of simplicity we consider that nodes can send multiple flows simultaneously and the routes, source nodes and destination nodes of the flows are the same as presented in Figure 7 and we present here only the results for switch SW1.

For this experiment we consider the size of a CDT frame to be equal to $C_{CDT} = 2\mu s$. There are two CDT streams in the network, generating from nodes N3 and N4. The CDT streams are separated by $100\mu s$, hence we do not investigate the case of a singe Protected Window as the time between the two flows would be too big of a waste relative to the size of the frames in the network. That is, we limit our investigation to the case of two Protected Windows with minimal waste of bandwidth, i.e. $L_{CDT} = 2\mu s$.

As the largest size of any frame on a TSN switch can be 1534 bytes, which means approximately $12\mu s$ on a 1Tb switch, we considered that there are 12 AVB_A streams passing through switch SW1, with transmission times varying from $1\mu s$ to $12\mu s$, in increments of $1\mu s$ such that each stream has a distinct transmission time, i.e. $C_{A_7} = 7\mu s$. The utilization of class AVB_A is $U_A = 0.624$ which is less than its bandwidth reservation $BR_A = 0.7504$, meaning that all streams of the class have bounded delays. Similarly we consider that there are six streams of class AVB_B passing through switch SW1, with transmission times varying from $1\mu s$ to $6\mu s$. The

---

[3]https://members.loria.fr/DMaxim/artifact-rtns2017-tsnavb/

			Single Protected-Window		Multiple Protected Windows	
Stream	AVB Analisys	AVB Simulation	TSN Analysis	TSN Simulation	TSN Analysis	TSN Simulation
A1	85	81	261	260	165	124
A2	85	81	261	260	165	124
B1	182	52	358	317	262	139

Table 4: Worst-case delays on switch SW1 for the default experiment

utilization of class AVB_B is $U_B = 0.084$ which is less than its bandwidth reservation $BR_B = 0.1792$, meaning that all AVB_B streams have bounded delays. The transmission times of AVB streams are presented in the second column of Table 5. All periods T_i and deadlines (D_i^{local} and D_i^{global}) are the same as in Table 3. There are 10 streams of class BE passing through switch SW1 and each of them has a transmission time of $12\mu s$.

The theoretical worst case-delays for the AVB traffic computed with our proposed analysis, as well as the largest delays observed during simulation for each stream are presented in Table 5. Once again we observe that the computed delays correctly and tightly upper-bound the delays observed during simulation. In this experiment, the pessimism induced by the Protected Windows is reduced (compared to the previous experiment) as these are relatively small with respect to the size of the TAS cycle L_{TAS}. For example, in all our simulation stream A1 was transmitted before being blocked by the second Protected Windows of the TAS, denoted by the fact that the difference between the observed delay and the theoretical delay equals $14\mu s$, which is exactly the size of a Protected Window. Stream B6 on the other hand, suffers a high amount of blocking from AVB_A streams, pushing it to also be blocked by the second Protected Window of the TAS, leading to a large transmission delay, which is correctly computed by our analysis.

6 CONCLUDING REMARKS

In this paper we have investigated the effects of time aware shaping on the transmission of AVB flows in TSN switches. We have presented a description of the mechanisms involved and how they interact and influence the scheduling of these flows. We propose an analysis for computing the local delays of AVB flows and an utilization based feasibility condition. Our experiments show that the analysis is safe for all cases that we have regarded and the tightness varies depending on the load of the switch and the complexity of the TAS scheduling table, with more protected windows in the TAS cycle inducing more pessimism in the analysis.

Acknowledgements

REFERENCES

[1] Mohammad Ashjaei, Gaetano Patti, Moris Behnam, Thomas Nolte, Giuliana Alderisi, and Lucia LoBello. 2017. Schedulability analysis of Ethernet Audio Video Bridging networks with scheduled traffic support. *Real-Time Systems* (2017), 1–52. https://doi.org/10.1007/s11241-017-9268-5

[2] U. D. Bordoloi, A. Aminifar, P. Eles, and Z. Peng. 2014. Schedulability analysis of Ethernet AVB switches. In *2014 IEEE 20th International Conference on Embedded and Real-Time Computing Systems and Applications*. 1–10. https://doi.org/10.1109/RTCSA.2014.6910530

[3] Jingyue Cao, Pieter J.L. Cuijpers, Reinder J. Bril, and Johan J. Lukkien. 2016. Independent Yet Tight WCRT Analysis for Individual Priority Classes in Ethernet AVB. In *Proceedings of the 24th International Conference on Real-Time Networks and Systems (RTNS '16)*. ACM, New York, NY, USA, 55–64. https://doi.org/10.1145/2997465.2997493

[4] J. Cao, P. J. L. Cuijpers, R. J. Bril, and J. J. Lukkien. 2016. Tight worst-case response-time analysis for ethernet AVB using eligible intervals. In *2016 IEEE World Conference on Factory Communication Systems (WFCS)*. 1–8. https://doi.org/10.1109/WFCS.2016.7496507

[5] Silviu S. Craciunas, Ramon Serna Oliver, Martin Chmelík, and Wilfried Steiner. 2016. Scheduling Real-Time Communication in IEEE 802.1Qbv Time-Sensitive Networks. In *Proceedings of the 24th International Conference on Real-Time Networks and Systems (RTNS '16)*. ACM, New York, NY, USA, 183–192. https://doi.org/10.1145/2997465.2997470

[6] Frank Dürr and Naresh Ganesh Nayak. 2016. No-wait Packet Scheduling for IEEE Time-sensitive Networks (TSN). In *Proceedings of the 24th International Conference on Real-Time Networks and Systems (RTNS '16)*. ACM, New York, NY, USA, 203–212. https://doi.org/10.1145/2997465.2997494

[7] IEEE. 2009. IEEE 802.1Qav Forwarding and Queuing Enhancements for Time-Sensitive Streams. (2009).

[8] IEEE. 2011. IEEE 802.1AS Standard for local and metropolitan area networks-timing and synchronization for time-sensitive applications in bridged local area networks. (2011).

[9] IEEE. 2015. IEEE 802.1Qbv Standard for local and metropolitan area networks, bridges and bridged networks, amendment 25: enhancement for scheduled traffic. (2015).

[10] Xiaoting Li and Laurent George. 2017. Deterministic delay analysis of AVB switched Ethernet networks using an extended Trajectory Approach. *Real-Time Systems* 53, 1 (01 Jan 2017), 121–186. https://doi.org/10.1007/s11241-016-9260-5

[11] Dorin Maxim and Antoine Bertout. 2017. Analysis and Simulation Tools for Probabilistic Real-Time Systems. In *Proceedings of the 8th International Workshop on Analysis Tools and Methodologies for Embedded and Real-time Systems (WATERS)*.

[12] S. Thangamuthu, N. Concer, P. J. L. Cuijpers, and J. J. Lukkien. 2015. Analysis of Ethernet-switch traffic shapers for in-vehicle networking applications. In *2015 Design, Automation Test in Europe Conference Exhibition (DATE)*. 55–60. https://doi.org/10.7873/DATE.2015.0045

[13] D. Thiele, R. Ernst, and J. Diemer. 2015. Formal worst-case timing analysis of Ethernet TSN's time-aware and peristaltic shapers. In *2015 IEEE Vehicular Networking Conference (VNC)*. 251–258. https://doi.org/10.1109/VNC.2015.7385584

Stream	C_i	TSN Analysis	TSN Simulation
A1	1	138	124
A2	2	137	122
A3	3	137	121
A4	4	137	124
A5	5	137	126
A6	6	136	127
A7	7	136	124
A8	8	136	126
A9	9	136	130
A10	10	135	127
A11	11	135	130
A12	12	135	130
B1	1	201	181
B2	2	197	185
B3	3	193	175
B4	4	198	172
B5	5	185	171
B6	6	181	181

Table 5: Delays on switch SW1 for the extended experiment