Timing Analysis of AVB Traffic in TSN Networks using Network Calculus

Luxi Zhao, Paul Pop, Zhong Zheng, Qiao Li

***Abstract*—Time-Sensitive Networking (TSN) is a collection of standards that extend Ethernet to support safety-critical and real-time applications. TSN integrates multiple traffic types, i.e., Time-Triggered (TT) traffic scheduled based on Gate-Control Lists (GCLs), Audio-Video-Bridging (AVB) traffic that requires bounded latencies, and Best-Effort (BE) traffic, for which no guarantees are provided. This paper proposes a Network Calculus-based approach to determine the worst-case end-to-end delays of AVB traffic in a TSN network with both non-preemption and preemption modes. We consider the effects of TT traffic due to GCLs, "guard bands", i.e., time windows that block other traffic from transmitting, and preemption overhead on the service for AVB traffic. We provide a proof of non-overflow condition for AVB credit, which is used to control the AVB traffic transmission. The analysis method is evaluated on realistic test cases, and compared to related work.**

***Index Terms*—Time-Sensitive Networking; TSN; Audio-Video-Bridging; AVB; timing analysis; Network Calculus**

I. INTRODUCTION

Ethernet [1] is a well-established network protocol that has excellent bandwidth, scalability, compatibility and cost properties. However, it is not suitable for real-time and safety critical applications. Therefore, several extensions to Ethernet protocol have been proposed, such as, ARINC 664 Specification Part 7 (AFDX) [2], TTEthernet [3], and EtherCAT [4]. However, they are incompatible with each other, and as a result, they cannot operate on the same physical links in a network without losing real-time guarantees [5]. Consequently, the IEEE 802.1 Time-Sensitive Networking (TSN) task group [6] has been working since 2012 on standardizing real-time and safety-critical enhancements for Ethernet. In this paper we consider that TSN is supporting AVB (802.1BA) with the currently finished sub-standards IEEE 802.1Qbv [7] Enhancements for Scheduled Traffic by adding a Time-Aware Shaper (TAS) and with 802.1Qbu [8] for preemption.

TSN classifies flows into the three traffic-types based on the criticality, Time-Triggered (TT) traffic, Audio-Video Bridging (AVB) traffic [9] and Best-Effort (BE) traffic. TT traffic supports hard real-time applications that require very low latency and jitter. TT traffic has the highest priority and it is transmitted based on schedule tables called Gate-Control Lists (GCLs) that rely on a global synchronized clock (802.1ASrev [10]). AVB traffic is intended for applications that require bounded end-to-end latencies, but has a lower priority than TT traffic. AVB (802.1 BA) introduces two new shaped traffic classes (AVB Class A and B) with bounded worst-case end-to-end delays (WCDs). AVB uses the Credit-Based Shaper (CBS) [9] to prevent the starvation of lower priority flows. BE traffic has the lowest priority and is used for applications that do not require any timing guarantees. The choice of traffic type for messages depends on the particularities of applications.

The schedulability of the scheduled TT traffic can be guaranteed during design phase, by synthesizing the GCLs [13]. However, an AVB flow is schedulable only if its worst-case end-to-end delay (WCD) is smaller than its deadline. Although latency analysis methods have been successfully applied to AVB traffic in AVB networks [14], [15], [16], [17], they do not consider the effect that TT traffic has on the latency of AVB traffic in TSN. Moreover, in order to fit for more general case, [15] relaxes constraints of parameters (idle slope and send slope) of CBS in 802.1Qav protocol [11]. However, this may cause credit overflow, which should not be allowed [18]. A Network Calculus-based analysis to compute the WCDs of Rate-Constrained (RC) traffic with the consideration of the static scheduled TT frames in TTEthernet has been proposed [19], [20], [21], but the technique is not applicable for TSN. RC traffic differs from AVB, and TSN schedules TT traffic differently from TTEthernet: TTEthernet schedules each individual TT flows, whereas in TSN the GCLs do not refer to individual TT flows but to the TT queues in the output ports of network switches. [22] gave simulation results from a conceptual implementation of Ethernet AVB with additional TT traffic. However, CBS behavior with TT traffic in [22] is different with this paper, which may cause the credit overflow. And simulation gives the experimental upper bound delays based on scenarios, and cannot be used for certifications because of the missed rare events. Recently, the AVB Latency Math equation [9] has been extended to consider the TT traffic in TSN [23] based on the same CBS behavior with TT traffic. However, depending on the scenario the analysis is both unsafe, i.e. the delays calculated are smaller than the exact WCDs and overly pessimistic, i.e., the WCDs determined are very large (see Sect. VI for a comparison between our proposed analysis and [23]). In addition, it can only be used to determine the WCDs of Class A AVB traffic.

In this paper we are interested to propose a timing analysis for AVB traffic in a TSN network. The main contributions are:

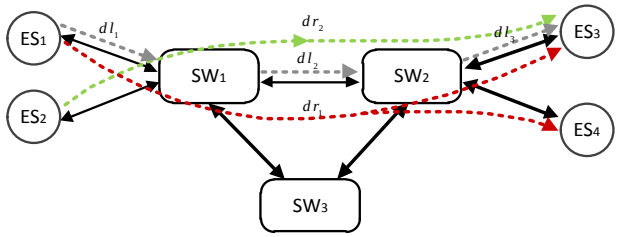

Fig. 1. TSN network topology example

- We first give a proof of non-overflow condition for AVB credit based on general parameters of CBS. This is a pre-condition of timing analysis for AVB traffic.
- We propose a Network Calculus-based method to determine the WCDs of AVB flows in a TSN network, considering the effects of TT traffic controlled by GCLs, guard bands for the non-preemption mode and preemption overheads for the preemption mode.
- We derive the remaining service curve for AVB Class A and Class B respectively considering the non-preemption and preemption integration modes.
- We evaluate the proposed approach on realistic test cases, including the Orion Crew Exploration Vehicle (CEV), and compare with related work.

II. System Model

A. Architecture Model

A TSN network is composed of a set of end systems (ES) and switches (SW) also called nodes, connected via physical links. The links are full duplex, allowing thus communication in both directions, and the networks can be multi-hop. The output port of a SW is connected to one ES or an input port of another SW. An example is presented in Fig. 1, where we have 4 ESes, ES_1 to ES_4, and 3 SWs, SW_1 to SW_3.

The topology of TSN network is modeled as an undirected graph $G(\mathbf{E},\mathbf{V})$, where $\mathbf{V}=\mathbf{ES}\cup\mathbf{SW}$ is the set of end systems ($\mathbf{ES}$) and switches ($\mathbf{SW}$), and $\mathbf{E}$ is the set of physical links. For Fig. 1, $\mathbf{V}=\mathbf{ES}\cup\mathbf{SW}=\{ES_1,ES_2,ES_3,ES_4\}\cup\{SW_1,SW_2,SW_3\}$, and the physical links $\mathbf{E}$ are depicted with black, double arrows.

A dataflow link $dl_i=[v_a,v_b]\in\mathbf{L}$, where $\mathbf{L}$ is the set of dataflow links in the network, is a directed edge from v_a to v_b, where v_a and $v_b\in\mathbf{V}$ can be ESes or SWs. The physical link rate is denoted as $dl_i.C$. In this paper, we assumed that all physical links have the same rate C. As there is only one output port for each dataflow link, dl_i can also refer to the output port h in v_a associated with the link to v_b. A dataflow routing $dr_i\in\mathbf{R}$ is an ordered sequence of dataflow links connecting a single source ES to one or more destination ESes. For example in Fig. 1, dr_1 connects the source end system ES_1 to the destination end systems ES_3 and ES_4, while dr_2 connects ES_2 to ES_3.

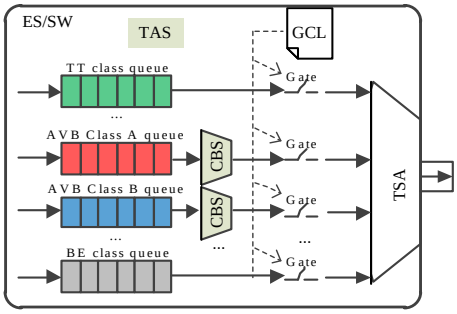

Fig. 2. A TAS for an output port in ES/SW

B. Application Model

The tasks of applications running in ESes communicate via flows, which have a single source and may have multiple destinations. As mentioned, TSN supports three traffic classes: TT, AVB and BE. We assume that the traffic class for each application has been decided by the designer. We define the sets τ_{TT}, τ_{AVB_M}, τ_{BE} with $\tau=\tau_{TT}\bigcup\tau_{AVB\_M}\bigcup\tau_{BE}$ is the set of all the flows in the TSN network, where the subscript $M\in\{A,B\}$ for AVB denotes the Class A or B. The routing $\tau_{TC_i}.dr$ of each flow τ_{TC_i} ($TC\in\{TT,AVB\_M,BE\}$) is known.

For each TT flow $\tau_{TT_i}\in\tau_{TT}$, we know the frame size l_{TT_i} and the period p_{TT_i}. As TT transmission in TSN is based on the Gate-Control Lists (GCLs) related to TT queues but not to individual TT flows, the periodicity of each TT flow τ_{TT_i} may not be maintained along its dataflow routing.

AVB traffic is compatible with the flow model in Ethernet AVB [24]. Together with strict priority scheduling, Ethernet AVB adds a Credit-Based Shaper (CBS) for each traffic class. The flows assigned the same priority belong to the same traffic class (Class A or Class B), and frames within each traffic class are in FIFO order. For an AVB flow $\tau_{AVB\_M_i}\in\tau_{AVB\_M}$, we know its frame size $l_{AVB\_M_i}$, period $p_{AVB\_M_i}$ in the source ES and the traffic class it belongs to ($M\in$ Class A or B).

III. TSN Protocol

In this section, we present how TT and AVB flows are transmitted in TSN. IEEE 802.1ASrev provides a clock synchronization protocol to a global time base for TT transmission. Taking advantage of the global synchronized clock, IEEE 802.1Qbv defines a Time-Aware Shaper (TAS) to achieve low latency for TT traffic by establishing completely independent time windows. Fig. 2 gives an illustration of TAS for an output port of a node. Each TAS has eight queues for storing frames that wait to be forwarded on the corresponding link, one or more for TT queues, two for AVB (Class A and B respectively) and remaining for queues are used for BE. When frames from critical flows arrive at input ports, they are filtered into queues based on their stream identification using the per-stream filtering and policing functionality defined in IEEE

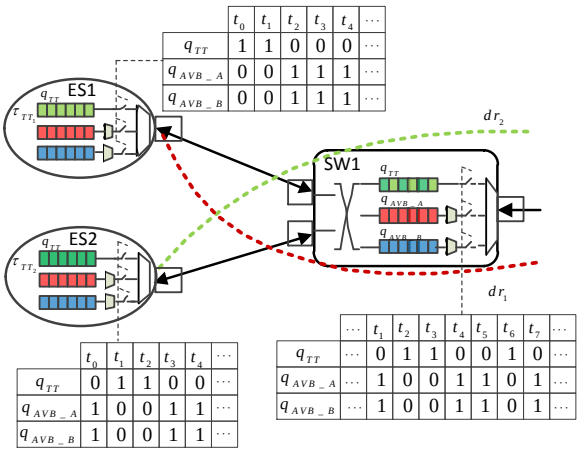

Fig. 3. An example of GCLs for output ports in ES/SW

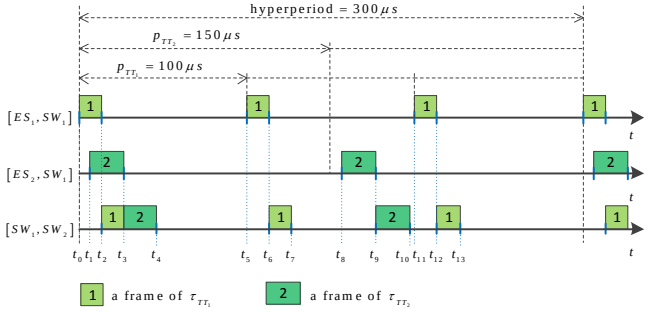

Fig. 4. Schedules of TT frames on dataflow links

802.1Qci [12]. Every queue has a gate with two states, open and closed. Frames waiting in the queue are eligible to be forwarded only if the associated gate is open.

TAS controls the gates for each queue according to a Gate-Control List (GCL), which is designed offline and contains the times when the associated gates are open and closed. The GCL is defined for each output port of an ES or SW, see the example in Fig. 3. In the figure, GCLs are given by tables below the respective queues. The open and closed states are respectively represented by 1 and 0. For example, the gate for TT queue (light green) in ES_1 is open from time t_0 to t_2, and closed from t_2 to t_5. Using the GCLs to schedule forwarding of frames in a route from source ES to destination ES, enables TT traffic suitable for hard real-time communication. The length of GCL is limited and it is repeated after a hyperperiod p^h_{GCL}, which is the Least Common Multiple (LCM) of periods of the intersecting TT flows sharing the output port h.

Fig. 4 shows using a Gantt chart how TT frames are transmitted using the GCLs from Fig. 3. The x-axis represents time dimension, while y-axis is related to output ports of nodes. Moreover, the rectangles represent TT frames transmission. The left side of rectangle means the start time of the frame transmitted, and its width represents the transmission duration which is related to the frame size and the physical link rate. Let us assume that there are two TT flows τ_{TT_1} and τ_{TT_2}

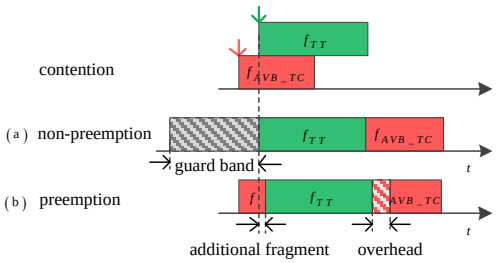

Fig. 5. Two integration modes: non-preemption and preemption

respectively sent from ES_1 and ES_2 with period $100\mu s$ and $150\mu s$. Then next such two TT flows will be multiplex and share the queue of output port in SW_1. There is an equivalence relationship between the set of GCLs in Fig. 3 and the schedules of TT frames on dataflow links for the associated output ports shown in Fig. 4. For example, at time t_0, the gate for TT traffic in ES_1 is open, therefore the TT frame of τ_{TT_1} is initiated to be transmitted from $[ES_1, SW_1]$.

Researchers have proposed methods to synthesize the GCLs [13], [25], and have outlined the constraints that have to be satisfied for schedule feasibility. For example, when associated gate for TT traffic is open, the remaining gates for other traffic are closed, and vice versa. In Fig. 3, the red and blue queues are respectively dedicated for Class A and B of AVB traffic. At time t_2 and t_3 in SW_1, the gate for the TT queue is open, while the gates for both AVB Class A and Class B queues are closed. During this period of time, frames from AVB traffic are forbidden to transmit until the TT gate is closed and associated AVB gates are open from time t_4. Therefore, AVB and BE traffic are prevented from initiating transmission in the time windows reserved for TT frames.

However, if an AVB or a BE frame is already in transmission at the beginning of time window for TT (see Fig. 5), TT traffic may be delayed. Hence, TSN uses two integration modes. One is *non-preemption* "guard band" before each time window of TT frame [7], as shown in Fig. 5a. The guard band has the length of a maximum-sized frame that may interfere with the respective TT traffic, which in the worst-case is the Ethernet Maximum Transmission Unit (MTU) of 1500 bytes. During the guard band, the gates associated with AVB and BE traffic are closed in advance to make sure that the link is idle when a TT queue is open for transmission. The non-preemption integration mode will lead to wasted bandwidth due to the guard band, but it ensures no delays for TT traffic. The other integration mode is the *preemption* mode defined by IEEE 802.1Qbu [8], as shown in Fig. 5b. With preemption, an AVB frame will be interrupted by a TT frame and its transmission is resumed from the stopping point once the transmission of the TT frame completes. When a gate for TT queue is ready for open, the associated gate for the preemptable AVB frame which is already in the process of being transmitted is open for the duration to finish transmitting a fragment before opening the TT queue [8]. Then when resuming transmission, the remaining AVB frame will include an overhead used to separate and

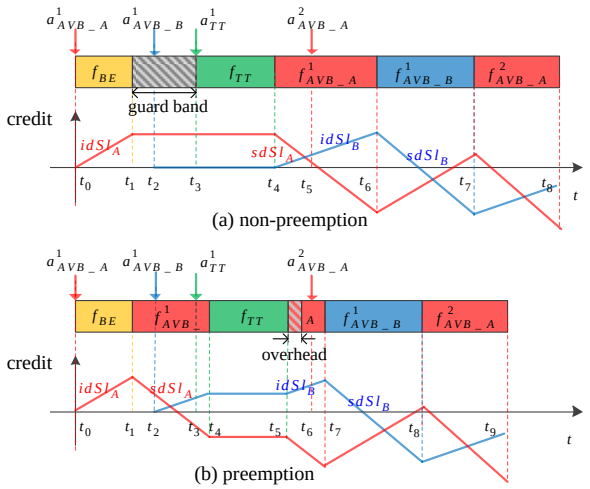

Fig. 6. CBS example

reassemble at the destination ES. Compared to the guard band, overhead can be negligible and therefore the use of preemption will decrease the latency of AVB traffic and increase bandwidth. However, it will introduce jitter for TT traffic.

The availability of an AVB queue is also determined by a Credit-Based Shaper (CBS) and the purpose of CBS is to prevent the starvation of lower priority flows. Hence, an enqueued AVB frame is allowed to be transmitted if (i) the queue gate is open, (ii) the CBS allows it and (iii) there are no other higher priority AVB frames being transmitted.

In the following, let us first explain how CBS works in TSN. Each AVB class has an associated credit value. The transmission of an AVB frame cannot be initiated when this credit is negative. And the credit is initialized to zero. In AVB network [9], if the AVB queue not empty, credit can be decreased with a send slope $sdSl_M$ ($M \in \{A,B\}$) during the transmission of an frame of AVB Class M and increased with an idle slope $idSl_M$ ($M \in \{A,B\}$) when Class M frames are waiting to be transmitted. Moreover, if the AVB queue is empty, and its credit is positive, the credit is set to zero; otherwise, it is increased with idle slope until zero. This is also the same with the situation in TSN network if the gate for associated AVB traffic queue is open. Besides, in TSN network, there needs additional consideration about AVB credit when the time gate is closed, which is still an open question [27]. To avoid the credit overflow, we are interested to the frozen form in this paper, i.e., the credit is frozen when the associated AVB gate is closed.

We show an example in Fig. 6 how CBS works interfered with TT and BE frames respectively with two integration modes. Rectangles on the first timeline represent the transmission of frames and down arrows on top give the arrival times of frames. Polylines on the second timeline shows the variation of credit for respective AVB class, where AVB Class A and B are respectively shown with red and blue. Fig. 6a is shown with the non-preemption mode. An AVB Class A frame $f^1_{AVB\_A}$ arrives at t_0, meanwhile a BE frame is on transmission. Due to non-preemption of BE frames, $f^1_{AVB\_A}$ has to wait until f_{BE} finishing its transmission, and credit A is increased with the idle slope $idSl_A$. At time t_1, the transmission of f_{BE} is done. However, the AVB gates are closed due to the reservation for TT traffic and insufficient idle interval (caused guard band) for the whole frame $f^1_{AVB\_A}$ transmission. Therefore, credit A is frozen during $[t_1,t_4]$ when AVB gates are closed. Even if an AVB Class B frame $f^1_{AVB\_B}$ arrvies at t_2, its credit should also be frozen. From time t_4, since the gate for TT queue is closed and Class A has higher priority than Class B, $f^1_{AVB\_A}$ is allowed to be transmitted. Credit A and B are respectively decreased and increased with the send slope $sdSl_A$ and idle slope $idSl_B$. During the transmission of $f^1_{AVB\_A}$, another frame $f^2_{AVB\_A}$ is enqueued in the Class A queue at time t_5. Then at t_6 when frame $f^1_{AVB\_A}$ finishes, there are two frames $f^2_{AVB\_A}$ and $f^1_{AVB\_B}$ waiting to be transmitted. But credit A at this time is negtive, therefore $f^2_{AVB\_A}$ is not allowed to transmit and $f^1_{AVB\_B}$ obtains the transmitting permission. At the end of $f^1_{AVB\_B}$ transmission, $f^2_{AVB\_A}$ starts transmission as its credit has increased to greater than 0. In Fig. 6b, the arrival times are all the same compared in Fig. 6a. However, due to the preemption integration mode, the TT frame f_{TT} is delayed from t_3 to t_4, and the remaining frame $f^1_{AVB\_A}$ is added with an overhead. In addition, we can find that the transmission of AVB frames in Fig. 6b are finished earlier than in Fig. 6a.

IV. Network Calculus Background

Network Calculus [26] is a mature theory proposed for deterministic performance analysis. It is used to construct arrival and service curve models for the investigated flows and network nodes. The arrival and service curves are defined by means of the min-plus convolution.

An arrival curve $\alpha(t)$ is a model constraining the arrival process $R(t)$ of a flow, in which $R(t)$ represents the input cumulative function counting the total data bits of the flow that has arrived in the network node up to time t. We say that $R(t)$ is constrained by $\alpha(t)$ if

$$R(t) \le \inf_{0 \le u \le t} \{R(u) + \alpha(t-u)\} = (R \otimes \alpha)(t), \qquad (1)$$

where *inf* means infimum (greatest lower bound) and $\otimes$ is the notation of min-plus convolution. A typical example of an arrival curve is the "leaky bucket" model given by

$$\alpha_{\sigma,\rho}(t) = \begin{cases} \rho t + \sigma, & t \ge 0 \\ 0, & t < 0 \end{cases},$$

where σ represents the maximum burst tolerance of the flow and ρ is the long-term average rate of the flow.

A service curve $\beta(t)$ models the processing capability of the available resource. Assume that $R^*(t)$ is the departure process, which is the output cumulative function that counts the total data bits of the flow departure from the network node up to time t. We say that the network node offers the service curve $\beta(t)$ for the flow if

$$R^*(t) \geq \inf_{0 \leq u \leq t} \{R(u) + \beta(t-u)\} = (R \otimes \beta)(t). \quad (2)$$

A typical example of the service curve is the rate-latency service curve given by

$$\beta_{R,T}(t) = R[t-T]^+,$$

where R represents the service rate, T represents the service latency and the notation $[x]^+$ is equal to x if $x \geq 0$ and 0 otherwise.

If a flow $R(t)$ of arrival curve $\alpha(t)$ across a server with the service curve $\beta(t)$, then the output flow $R^*(t)$ can be bounded by the arrival curve $\alpha'(t)$,

$$\alpha'(t) = \alpha \oslash \beta(t) = \sup_{u \geq 0} (\alpha(t+u) - \beta(u)). \quad (3)$$

where *sup* means supremum (least upper bound).

Let us assume that the flow constrained by the arrival curve $\alpha(t)$ traverses the network node offering the service curve $\beta(t)$. Then the latency experienced by the flow in the network node is bounded by the maximum horizontal deviation between $\alpha(t)$ and $\beta(t)$

$$h(\alpha, \beta) = \sup_{s \geq 0} \{inf \{\tau \geq 0 | \alpha(s) \leq \beta(s+\tau)\}\}. \quad (4)$$

V. Worst-case Analysis for AVB Traffic

A. *Non-overflow condition for AVB traffic*

We first give the non-overflow condition for AVB credit in this section, which is the pre-condition to bound the credit of AVB Class M and thus to the subsequent service analysis for the AVB traffic.

Let us recall from Sect. III how AVB is transmitted. In TSN network, the transmission of AVB traffic is not only related to the gate states, but also to CBS. Although TT transmissions in both preemption and non-preemption modes delays AVB traffic, the credits for both classes are frozen during these periods. Therefore, we can say that AVB credits will not be affected by TT traffic. In fact, the credit value is related to the transmission and backlog of AVB frames during respective AVB gate open, and settings of idle slope $idSl_M$ and send slope $sdSl_M$ for each traffic class, which are configuration parameters given by designer. 802.1Qbv [7] gives the constraints between $idSl_M$ and $sdSl_M$, i.e., $sdSl_M = idSl_M - C$, and [15] considers the more general situation, assuming any $idSl_M$ and $sdSl_M$.

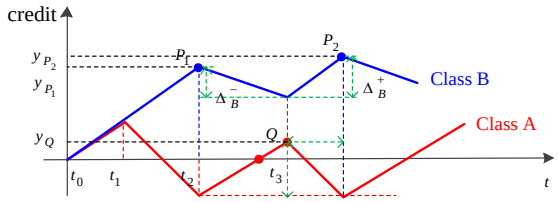

Fig. 7. Example of credit of Class B infinitely increasing

However, with any kind of parameters for AVB traffic A and B, the overloaded AVB traffic may cause the credit overflow, which is a problem that may cause the failure of the anti-starvation function of CBS and should not be allowed. Thus, we need to constrain $idSl_M$ and $sdSl_M$ to make sure the credit of Class M is bounded in more general situation. This has not been discussed in the literature so far.

As Class A has higher priority, its credit should be decreased with the associated frame transmission as soon as the credit A is larger than 0 at the end of the transmission of another lower priority class frame, and be increased when the credit A is smaller than 0 at the end of the transmission of a Class A frame. Therefore, for any $idSl_A$ and $sdSl_A$, credit A can still be bounded by [15],

$$l_{AVB\_A}^{\max} \cdot \frac{sdSl_A}{C} \leq credit_A \leq \max\{l_{AVB\_B}^{\max}, l_{BE}^{\max}\} \cdot \frac{idSl_A}{C}. \quad (5)$$

But, due to the lower priority of Class B, its credit can be increased if frames of Class A allows to be transmitted no matter whether the credit B is smaller than 0. Therefore, credit B may overflow if a greedy Class A happens, i.e., always sending data, for example in Fig. 7. As the credit is frozen during the TT transmission and guard bands with both non-preemption and preemption modes, Fig. 7 ignores the frozen parts when discussing this problem.

In this section, we are interested to give the constraints condition that credit B does not overflow in more general situation, as given in the following Theorem 1.

Theorem 1 The non-overflow condition of credit B is

$$sdSl_A \cdot sdSl_B \geq idSl_A \cdot idSl_B. \quad (6)$$

Proof:

The credit B is increased only when Class B frames waiting in the queue. Then in the first rising period in Fig. 7, the upper bounds of credit B can be reached at the end of the transmission of a maximum BE frame and the maximum number of Class A frames with the maximum size. Such as the duration $[t_0, t_2]$ in Fig. 7, credit A and B are increased due to the maximum BE frame transmission, and then credit B increases to the upper bound P_1 when credit A starts at the maximum value and finishes at the minimum value as given in formula (4).

However, due to the arbitrariness of $idSl_M$ and $sdSl_M$, P_1 in the first rising of credit B may not be the real upper bound. As shown in Fig. 7, y_{P_2} is larger than y_{P_1}. Then at time t_2, even if there are frames in queue A, they cannot be forwarded because of the negative credit A. Therefore, credit A must be increased next, and Class B traffic obtains the service and its credit is decreased. As long as credit A is larger than 0, Class A traffic with higher priority will regain the service for the worst-case duration to finish the current Class B frame transmission, regardless of whether credit B is greater than 0. Assume that Class A regains service at point Q in Fig. 7, and at this time credit B is decreased from point P_1 by

$$\Delta_B^- = -\left[\left(-credit_A^{\min} + y_Q\right)/idSl_A\right]\cdot sdSl_B, \tag{7}$$

where y_Q is the ordinate value of point Q. Then credit A is decreased again from the point Q and credit B goes into the second rising period. The maximum service time for Class A in the second service period is

$$t_A^- = -\left(y_Q - credit_A^{\min}\right)/sdSl_A. \tag{8}$$

At this time, credit B reaches the second highest point P_2 and is increased by

$$\Delta_B^+ = t_A^- \cdot idSl_B. \tag{9}$$

The above discussion also applies to the subsequent rising period of credit B. Then to make credit B increase finitely, $y_{P_{i+1}} \le y_{P_i}$ ($i \ge 1$) should be always true, where P_i is the ordinate value of the highest point in the i th rising period of credit B. Therefore, the non-overflow condition of upper bound of credit B is,

$$y_{P_{i+1}} - y_{P_i} = \Delta_B^+ - \Delta_B^- = \left(y_Q - credit_A^{\min}\right)\left(\frac{sdSl_B}{idSl_A} - \frac{idSl_B}{sdSl_A}\right) \le 0$$

i.e.,

$$sdSl_B \cdot sdSl_A \ge idSl_B \cdot idSl_A.$$

■

With the non-overflow condition in more general case, the bounds of credit B satisfy [15],

$$l_{AVB\_B}^{\max}\cdot\frac{sdSl_B}{C} \le credit_B \le \left(\frac{l_{BE}^{\max} + l_{AVB\_A}^{\max}}{C} - l_n^{\max}\cdot\frac{idSl_A}{sdSl_A\cdot C}\right)\cdot idSl_B \tag{10}$$

where $l_n^{\max} = \max\left\{l_{AVB\_B}^{\max}, l_{BE}^{\max}\right\}$.

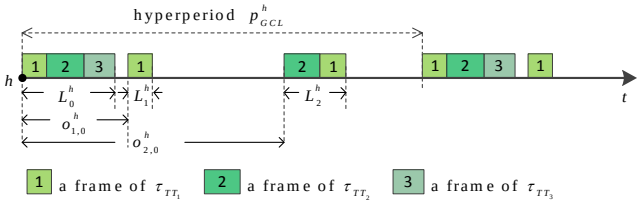

Fig. 8. GCL for TT traffic in an output port h

B. Service Curve for AVB traffic with non-preemption and preemption modes

In this section, we focus on the service curve analysis for AVB Class M ($M \in \{A, B\}$) available in an output port h by considering the presence of TT traffic with the non-preemption and the preemption modes, respectively.

Let us start by discussing the aggregate arrival curve considering the impact of TT traffic in the output port h, as the remaining service for AVB traffic depends on it. In TTEthernet network, TT aggregate arrival curve is obtained by summing the arrival curve of each single intersecting periodic TT flows shifting with relative offsets [21]. However in TSN, GCLs control the gate states for TT queue and not the TT frames. If end systems are non-scheduled [13], TT flows may lose the nature of the periodicity along the transmission path. In this paper, we establish the aggregate arrival curve with the impact of TT traffic based on the TT traffic window. In addition, the constraint of gurad band effect for the non-preemption mode is merged into the TT aggregate arrival curve, and the constraint curve of overheads for the preemption mode is also built.

As mentioned in Sect. III, the GCL for an output port h is repeated after the hyperperiod p_{GCL}^h. Therefore, any TT traffic window takes p_{GCL}^h as a cycle. For a given GCL in an output port, it can be known that the finite number of TT traffic windows in the hyperperiod p_{GCL}^h, which is assumed with N^h. For example N^h equals to 3 in Fig. 8. In addition, we can also know when the TT gate opens and how long it lasts. It is assumed that the length of the i th TT traffic window in the output port h is written as L_i^h ($i \in \left[0, N^h - 1\right]$), as shown in Fig. 8. Moreover, the relative offset $o_{j,i}^h$ ($j \in \left[i+1, i+N^h-1\right]$) between the starting time of the i th and j th TT windows is known by taking the i th TT window as the reference. For example in Fig. 8, $o_{1,0}^h$ and $o_{2,0}^h$ respectively represent the offsets by taking the 0th TT window with length L_0^h as the reference. Note that $o_{j,i}^h$ equals to 0 if $j = i$. Then a possible TT aggregate arrival curve can be given by,

$$\alpha_{TT,i}^h(t) = \sum_{j=i}^{i+N^h-1} L_j^h \cdot C \cdot \left\lceil \frac{t - o_{j,i}^h}{p_{GCL}^h} \right\rceil. \tag{11}$$

where $L_j^h \cdot C$ represents the maximum number of bits that

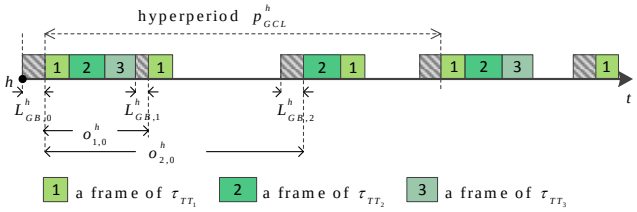

Fig. 9. Guard bands before TT traffic windows

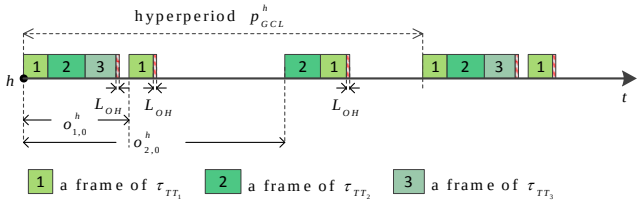

Fig. 10. Overheads after TT traffic windows

could be transmitted during the TT traffic window of length L^h_j; each staircase function represents the upper bound of TT transmission in the periodic TT traffic windows of length L^h_j; the relative offsets give the relationships between different TT traffic windows in the hyperperiod. The proof of formula (10) is similar to that in [21]. Then, by selecting different TT traffic windows in the hyperperiod as the reference, we will obtain a set of possible aggregate arrival curves $\alpha^h_{TT,i}(t)$ ($i \in \left[0, N^h - 1\right]$) from (10). In the worst-case, the TT aggregate arrival curve is the upper envelope of all these possible curves, as given by.

Lemma 1 The aggregate arrival curve for intersecting TT flows in an output port h is

$$\alpha^h_{TT}(t) = \max_{0 \le i \le N^h - 1} \left\{ \alpha^h_{TT,i}(t) \right\}. \tag{12}$$

However, for the non-preemption mode, the guard band is established before each time that the TT traffic transmission starts. Then the AVB transmission is not permitted between the start of the guard band and the start of the TT traffic window. In the worst-case, the guard band $L^h_{GB,i}$ before the i th ($i \in \left[0, N^h - 1\right]$) TT traffic window is as long as the minimum of the transmission time of maximum size of AVB frames competing the output port h and idle time interval between two consecutive TT traffic windows, i.e., $\left(i - 1 + N^h\right)\%N^h$ th and i th windows. For example in Fig. 9, $L^h_{GB,0}$ equals to the transmission time of the maximum AVB frame, since there is a much larger idle time between the 2nd and the 0th TT traffic windows. But $L^h_{GB,1}$ equals to the idle time between the 0th and the 1st windows, which is smaller than the transmission time of the maximum AVB frame. For the constraint of guard bands, [21] constructs a separate overall guard band (also called timely block in TTEthernet) curve $\gamma^h_{GB}(t)$. But in this paper, we will merge guard bands effects into the construction of TT aggregate arrival curve due to the fact that guard bands may only appear immediately before each TT traffic window and the credit for the respective AVB traffic is frozen during guard bands and TT traffic windows. Here it is assumed as a new window called "GB+TT" window with length $L^h_i + L^h_{GB,i}$.

Lemma 2 With the non-preemption mode, the aggregate arrival curve for intersecting TT flows and guard bands in an output port h is given by

$$\alpha^h_{GB+TT}(t) = \max_{0 \le i \le N^h - 1} \left\{ \sum_{j=i}^{i+N^h-1} \left(L^h_j + L^h_{GB,j}\right) \cdot C \cdot \left\lceil \frac{t - o^h_{j,i} + L^h_{GB,j} - L^h_{GB,i}}{p^h_{GCL}} \right\rceil \right\} \tag{13}$$

which is similar to the curves in (10) and (11). Here we have $\left(L^h_j + L^h_{GB,j}\right) \cdot C$ just by adding the worst-case number of bits caused by the j th guard band to the maximum bits of TT frames transmitted in the j th TT traffic window, both of which postpone the service for AVB traffic. Moreover, due to the possible of different length of guard band, the distance between new "GB+TT" windows may change, which is reflected by $L^h_{GB,j} - L^h_{GB,i}$ in (12).

For the preemption mode, if an AVB frame is preempted, an overhead is added with the remaining AVB frame. In the worst-case, each TT traffic window preempts a frame of AVB Class M, as shown in Fig. 10. It is assumed that the length of the overhead is $L_{OH} \cdot C$. Overheads can be taken as the separate part causing the latency of AVB traffic. Since the overhead will only appear immediately after each TT traffic window, the constraint curve of overheads can be given by the following Lemma 3.

Lemma 3 With the preemption mode, the overhead arrival curve in an output port h is given by

$$\alpha^h_{OH}(t) = \max_{0 \le i \le N^h - 1} \left\{ \sum_{j=i}^{i+N^h-1} L_{OH} \cdot C \cdot \left\lceil \frac{t - o^h_{j,i} - L^h_j}{p^h_{GCL}} \right\rceil \right\}, \tag{14}$$

where L^h_j is the length of j th TT traffic window.

In the following, we focus on the analysis of service curve for AVB Class M in the output port h. Regardless of modes, any time interval Δt can be decomposed by,

$$\Delta t = \Delta t^+ + \Delta t^- + \Delta t^0. \tag{15}$$

where $\Delta t^+ = \sum_i \Delta t^+_i$ represents the rising time of credit M, $\Delta t^- = \sum_j \Delta t^-_j$ represents the descent time of credit M and $\Delta t^0 = \sum_k \Delta t^0_k$ is the frozen time of credit M, as shown with credit A for example in Fig. 11. The service could only be supplied for AVB traffic during the descent time Δt^- of credit. Then the remaining service for AVB traffic with respective integration modes is given in the following theorems.

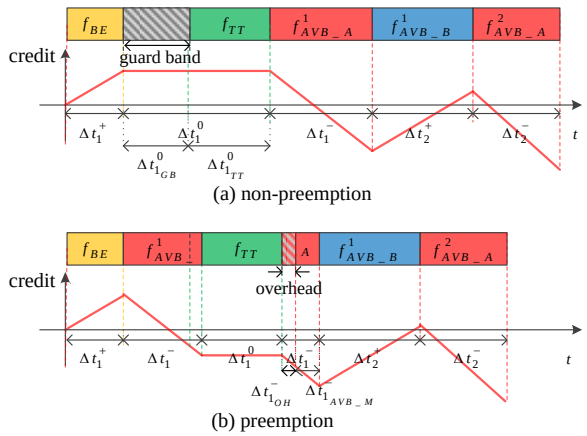

Fig. 11. Decomposed interval with different integration modes

Theorem 2 The service curve for AVB Class M ($M \in \{A, B\}$) with non-preemption mode in an output port h is given by

$$\beta_{AVB\_M}^{h[npr]}(t)=\frac{C\cdot idSl_M}{idSl_M-sdSl_M}\left[\sup_{0\le u\le t}\left\{u-\frac{\alpha_{GB+TT}^{h}(u)}{C}-\frac{credit_M^{\max}}{idSl_M}\right\}\right]^{+} \tag{16}$$

where $[npr]$ represents the non-preemption integration mode, $\alpha_{GB+TT}^{h}(s)$ is given by Lemma 2 if for non-preemption mode, and $credit_M^{\max}$ is the upper bound of credit M given by (4) and (9).

Proof: Assume that $R_M^h(t)$ (resp. $R_{TT}^h(t)$, $R_{GB}^h(t)$) and $R_M^{h^*}(t)$ (resp. $R_{TT}^{h^*}(t)$, $R_{GB}^{h^*}(t)$) are the arrival and departure processes of AVB flows of Class M (resp. TT flows, guard bands) crossing through the output port h. Let s be the beginning of the latest server busy period. At time s, the backlogs for all flows are empty, i.e., $R_M^{h^*}(s)=R_M^h(s)$, $R_{TT}^{h^*}(s)=R_{TT}^h(s)$, $R_{GB}^{h^*}(s)=R_{GB}^h(s)$ and $credit_M(s)=0$.

For some arbitrary time $t \ge s$, the interval $\Delta t = t-s$ can be decomposed by,

$$\Delta t = \Delta t^{+} + \Delta t^{-} + \Delta t^{0}.$$

For the non-preemption mode, Δt^0 is caused by the guard bands and TT traffic windows, i.e.,

$$\Delta t^{0} = \Delta t_{TT}^{0} + \Delta t_{GB}^{0},$$

and Δt^- represents the duration of frame transmission of AVB Class M. Therefore we have

$$\Delta t^{-} = \Delta t_{AVB\_M}^{-}.$$

Then, the variation of credit during the time interval Δt satisfies

$$\begin{aligned} creidt_M(t)-creidt_M(s) &= creidt_M(t) \\ &= \Delta t^{+}\cdot idSl_M + \Delta t^{-}\cdot sdSl_M \\ &= \left(\Delta t-\left(\Delta t_{TT}^{0}+\Delta t_{GB}^{0}\right)\right)\cdot idSl_M - \Delta t_{AVB\_M}^{-}\cdot\left(idSl_M-sdSl_M\right) \end{aligned}$$

Therefore, with non-preemption mode, we obtain the relationship of service times for AVB Class M, TT traffic and guard bands in any interval Δt,

$$\Delta t_{AVB\_M}^{-}=\frac{\left(\Delta t-\left(\Delta t_{TT}^{0}+\Delta t_{GB}^{0}\right)\right)\cdot idSl_M - creidt_M(t)}{idSl_M-sdSl_M} \tag{17}$$

Moreover, for the worst-case, the output frames of TT traffic during Δt can be given by

$$R_{TT}^{h^*}(t)-R_{TT}^{h^*}(s)=R_{TT}^{h^*}(t)-R_{TT}^{h}(s)=C\cdot\Delta t_{TT}^{0},$$

and similarly the wasted service during Δt due to guard bands is

$$R_{GB}^{h^*}(t)-R_{GB}^{h^*}(s)=R_{GB}^{h^*}(t)-R_{GB}^{h}(s)=C\cdot\Delta t_{GB}^{0}.$$

Thus, $\Delta t^0 = \Delta t_{TT}^0 + \Delta t_{GB}^0$ can be limited by,

$$\begin{aligned} \Delta t^{0} &\le \left(\left(R_{TT}^{h}(t)+R_{GB}^{h}(t)\right)-\left(R_{TT}^{h}(s)+R_{GB}^{h}(s)\right)\right)/C \\ &\le \alpha_{GB+TT}^{h}(\Delta t)/C \end{aligned} \tag{18}$$

Then, considering (17) and (18), the output frames of AVB Class M over the interval Δt is bounded by,

$$\begin{aligned} R_M^{h^*}(t)-R_M^{h^*}(s) &= C\cdot\Delta t_{AVB\_M}^{-} \\ &\ge C\cdot\frac{\left(\Delta t-\alpha_{GB+TT}^{h}(\Delta t)/C\right)\cdot idSl_M - credit_M^{\max}}{idSl_M-sdSl_M}\cdot \end{aligned}$$

Since the departure cumulative function $R_M^{h^*}(t)$ is a wide-sense increasing function, we have

$$R_M^{h^*}(t)-R_M^{h^*}(s)\ge 0$$

and

$$\begin{aligned} &R_M^{h^*}(t)-R_M^{h^*}(s) \\ &\ge \sup_{0\le u\le \Delta t=t-s}\left\{C\cdot\frac{\left(u-\alpha_{GB+TT}^{h}(u)/C\right)\cdot idSl_M - credit_M^{\max}}{idSl_M-sdSl_M}\right\}\cdot \end{aligned}$$

Therefore,

$$R_M^{h*}(t)-R_M^h(s) \geq \left[\sup_{0\le u\le \Delta t}\left\{\frac{C\cdot idSl_M}{idSl_M-sdSl_M}\cdot\left(u-\frac{\alpha_{GB+TT}^h(u)}{C}-\frac{credit_M^{\max}}{idSl_M}\right)\right\}\right]^+ . = \beta_{AVB\_M}^{h[npr]}(t-s)$$

Then for $\forall 0\le s\le t$,

$$R_M^{h*}(t)\ge \inf_{0\le s\le t}\left\{R_M^h(s)+\beta_{AVB\_M}^{h[npr]}(t-s)\right\}=\left(R_M^h\otimes\beta_{AVB\_M}^{h[npr]}\right)(t).$$

Thus, with non-preemption integration mode $\beta_{AVB\_M}^{h[npr]}(t)$ is the service curve for AVB Class M.

■

Theorem 3 The service curve for AVB Class M ($M\in\{A,B\}$) with preemption mode in an output port h is given by

$$\beta_{AVB\_M}^{h[pr]}(t)=\frac{C\cdot idSl_M}{idSl_M-sdSl_M}\left[\sup_{0\le u\le t}\left\{u-\frac{\alpha_{TT}^h(u)}{C}-\frac{\alpha_{OH}^h(u)}{C}\cdot\frac{idSl_M-sdSl_M}{idSl_M}-\frac{credit_M^{\max}}{idSl_M}\right\}\right]^+ \quad (19)$$

where $[pr]$ represents the preemption integration mode, $\alpha_{TT}^h(t)$ and $\alpha_{OH}^h(t)$ are respectively given by Lemma 1 and Lemma 3.

Proof: Assume that $R_M^h(t)$ (resp. $R_{TT}^h(t)$, $R_{OH}^h(t)$) and $R_M^{h*}(t)$ (resp. $R_{TT}^{h*}(t)$, $R_{OH}^{h*}(t)$) are the arrival and departure processes of AVB flows of Class M (resp. TT flows, overheads) crossing through the output port h. Let s be the beginning of the latest server busy period. At time s, the backlogs for all flows are empty, i.e., $R_M^{h*}(s)=R_M^h(s)$, $R_{TT}^{h*}(s)=R_{TT}^h(s)$, $R_{OH}^{h*}(s)=R_{OH}^h(s)$ and $credit_M(s)=0$.

For the preemption mode, Δt^0 represents the length of TT traffic windows during the arbitrary interval $\Delta t=t-s$,

$$\Delta t^0=\Delta t_{TT}^0$$

and Δt^- can be broken down into the durations of frame transmission of AVB Class M and overheads duration due to preemption, i.e.,

$$\Delta t^-=\Delta t_{OH}^- +\Delta t_{AVB\_M}^- .$$

Then, the variation of credit in Δt satisfies

$$credit_M(t)-credit_M(s)=credit_M(t) = \Delta t^+\cdot idSl_M+\Delta t^-\cdot sdSl_M = \left(\Delta t-\Delta t_{TT}^0\right)\cdot idSl_M-\left(\Delta t_{OH}^-+\Delta t_{AVB\_M}^-\right)\cdot\left(idSl_M-sdSl_M\right) .$$

Thus, with preemption mode, we obtain the relationship of service times for AVB Class M, additional service time for AVB overheads and TT windows duration in any interval Δt,

$$\Delta t_{AVB\_M}^- = \frac{\left(\Delta t-\Delta t_{TT}^0\right)\cdot idSl_M-\Delta t_{OH}^-\cdot\left(idSl_M-sdSl_M\right)-credit_M(t)}{idSl_M-sdSl_M} \quad (20)$$

Since for the worst-case, the output frames of TT traffic during Δt is

$$R_{TT}^{h*}(t)-R_{TT}^{h*}(s)=R_{TT}^{h*}(t)-R_{TT}^h(s)=C\cdot\Delta t_{TT}^0 ,$$

and the additional service for AVB overheads during Δt can be given by

$$R_{OH}^{h*}(t)-R_{OH}^{h*}(s)=R_{OH}^{h*}(t)-R_{OH}^h(s)=C\cdot\Delta t_{OH}^- .$$

Thus, Δt_{TT}^0 can be limited by,

$$\Delta t_{TT}^0\le\left(R_{TT}^{h*}(t)-R_{TT}^h(s)\right)/C\le\alpha_{TT}^h(\Delta t)/C , \quad (21)$$

and similarly Δt_{OH}^- satisfies,

$$\Delta t_{OH}^-\le\alpha_{OH}^h(\Delta t)/C . \quad (22)$$

Then, considering (20), (21) and (22), the output frames of Class M over the interval Δt is bounded by,

$$R_M^{h*}(t)-R_M^{h*}(s)=C\cdot\Delta t_{AVB\_M}^-\ge\frac{C\cdot idSl_M}{idSl_M-sdSl_M}\cdot\left(\Delta t-\frac{\alpha_{TT}^h(\Delta t)}{C}-\frac{\alpha_{OH}^h(\Delta t)}{C}\cdot\frac{idSl_M-sdSl_M}{idSl_M}-\frac{credit_M^{\max}}{idSl_M}\right)$$

Similarly, we can get the service curve $\beta_{AVB\_M}^{h[pr]}(t)$ for AVB Class M with the preemption integration mode.

■

According to the network calculus theory, the upper bound latency of an Class M flow $\tau_{AVB\_M_i}$ in the output port h is given by the maximum horizontal deviation between the arrival curve $\alpha_{AVB\_M}^h(t)$ of intersecting flows of AVB Class M and the service curve $\beta_{AVB\_M}^h(t)$ for AVB Class M in the output port h,

$$D_{AVB\_M_i}^h=h\left(\alpha_{AVB\_M}^h(t),\beta_{AVB\_M}^h(t)\right), \quad (23)$$

where the service curve is from Theorem 2 or Theorem 3 depending on which integration modes to choose. The arrival curve in the output port h of the node can be given by

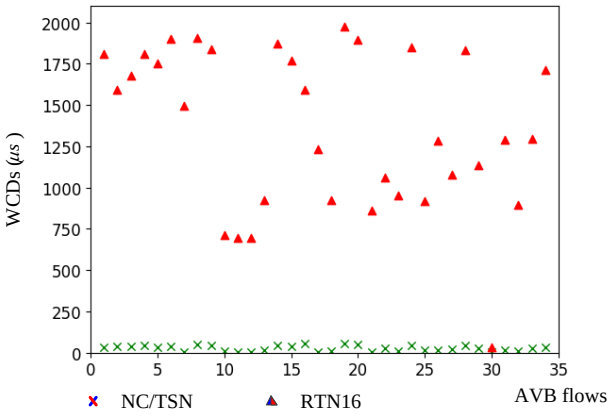

Fig. 12. Comparison of WCDs by NC/TSN and RTN16

$$\alpha^{h}_{AVB\_M}(t) = \sum_{\tau_{AVB\_M_i} \in h} \sigma^{h}_{AVB\_M_i} + \sum_{\tau_{AVB\_M_i} \in h} \rho^{h}_{AVB\_M_i} \cdot t ,$$

where $\sigma^{h}_{AVB\_M_i}$ is the burst of the flow $\tau_{AVB\_M_i}$ in h and $\rho^{h}_{AVB\_M_i}$ is the long-term rate of $\tau_{AVB\_M_i}$ in h. In the output port h_0 of source end system, $\sigma^{h_0}_{AVB\_M_i} = l_{AVB\_M_i}$ and $\rho^{h_0}_{AVB\_M_i} = l_{AVB\_M_i} / p_{AVB\_M_i}$. In addition, the output arrival curve $\alpha^{h_0}_{AVB\_M_i}{}'(t)$ for the output port h_0 can be taken as the input arrival curve $\alpha^{h}_{AVB\_M_i}(t)$ for the next output port h of the node along the dataflow routing of $\tau_{AVB\_M_i}$, and can be given by [26],

$$\begin{aligned}\alpha^{h}_{AVB\_M_i}(t) &= \alpha^{h_0}_{AVB\_M_i}\left(t + D^{h_0}_{AVB\_M_i}\right) \\ &= \sigma^{h_0}_{AVB\_M_i} + \rho^{h_0}_{AVB\_M_i} \cdot \left(t + D^{h_0}_{AVB\_M_i}\right)\end{aligned}.$$

By disseminating the computation of latency bounds along the routing $\tau_{AVB\_M_i}.dr$, the WCD of the flow $\tau_{AVB\_M_i}$ is obtained by the sum of delays from its source ES to its destination ES,

$$D_{AVB\_M_i} = \sum_{h \in \tau_{AVB\_M_i}.dr} D^{h}_{AVB\_M_i} + (h-1) \cdot d_{tech} . \qquad (24)$$

where d_{tech} is the constant technical latency in a SW.

VI. Experimental Results

For the evaluation of our approach, we have used four test cases, based on the Orion Crew Exploration Vehicle (CEV) [28] adapted to use TSN, respectively named here TC1 to TC4. CEV has a topology of 31 ESes, 15 SWs and 39 routes connected by dataflow links transmitting at 1 Gbps. Our proposed analysis is implemented in C++ using the RTC toolbox [29], running on a computer with Intel Core i7-3520M CPU at 2.90 GHz and 4 GB of RAM.

To compare our method using network calculus (called NC/TSN) with the existing approach in [23] (called RTN16),

TABLE I Parameters of Traffic in TC1

(a) Frame sizes and periods of TT traffic

Flow	Size (B)	Period (ms)	Flow	Size (B)	Period (ms)
TT1	145	62.5	TT9	186	37.5
TT2	508	125	TT10	1420	37.5
TT3	1268	37.5	TT11	197	25
TT4	888	37.5	TT12	246	12.5
TT5	170	25	TT13	103	12.5
TT6	1527	37.5	TT14	1053	37.5
TT7	578	62.5	TT15	913	75
TT8	908	75			

(b) Frame sizes, periods and traffic class of AVB traffic

Flow	Size (B)	Period (ms)	type	Flow	Size (B)	Period (ms)	type
RC1	63	125	A	RC18	47	125	B
RC2	57	125	A	RC19	116	125	A
RC3	70	125	B	RC20	8	125	A
RC4	25	125	A	RC21	38	125	A
RC5	4	125	B	RC22	8	125	A
RC6	19	125	B	RC23	48	125	B
RC7	56	125	A	RC24	18	125	A
RC8	12	125	A	RC25	49	125	A
RC9	10	125	B	RC26	16	125	B
RC10	55	125	A	RC27	10	125	A
RC11	26	125	A	RC28	124	125	A
RC12	4	125	B	RC29	4	125	A
RC13	40	125	A	RC30	9	125	A
RC14	4	125	A	RC31	68	125	A
RC15	14	125	B	RC32	57	125	A
RC16	187	125	A	RC33	16	125	B
RC17	4	125	A	RC34	17	125	A

we use a test case TC1 directly from [23], including 5 TT flows and 34 AVB flows of Class A. RTN16 can be only used to calculate the WCDs for AVB Class A considering the preemption mode. We show the WCD results in Fig. 12. As we can see from Fig. 12, RTN16 obtains very pessimistic WCDs, since it is taken as an approximation depending on the scenario. Our proposed analysis reduces the pessimism on average by 98.3% and 99.7% maximum.

In TC2, we are interested to verify how our method handles the two integration modes, i.e., non-preemption and preemption. TC2 is a more general case, running 15 TT flows, 20 AVB flows of Class A and 14 AVB flows of Class B (including multicast flows). The details of the TT and AVB flows are presented in TABLE I, and the GCLs have been generated using [25]. In addition, the idle slopes of Class A and B are respectively 60% and 15% of the total bandwidth, and we assume that parameters of CBS satisfies $sdSl_M = idSl_M - C$ in this experiment. The compared results are shown in Fig. 13. The WCDs for the preemption and non-preemption modes are respectively marked with "x" and "▲" symbols. WCDs are also distinguished with Class A and B by red and blue symbols respectively. As expected, the latency bounds with the preemption mode are lower than the bounds with non-preemption mode, since the length of the maximum guard band is longer than the length of preemption overhead. On average, non-preemption leads to 18.7% larger WCDs. This

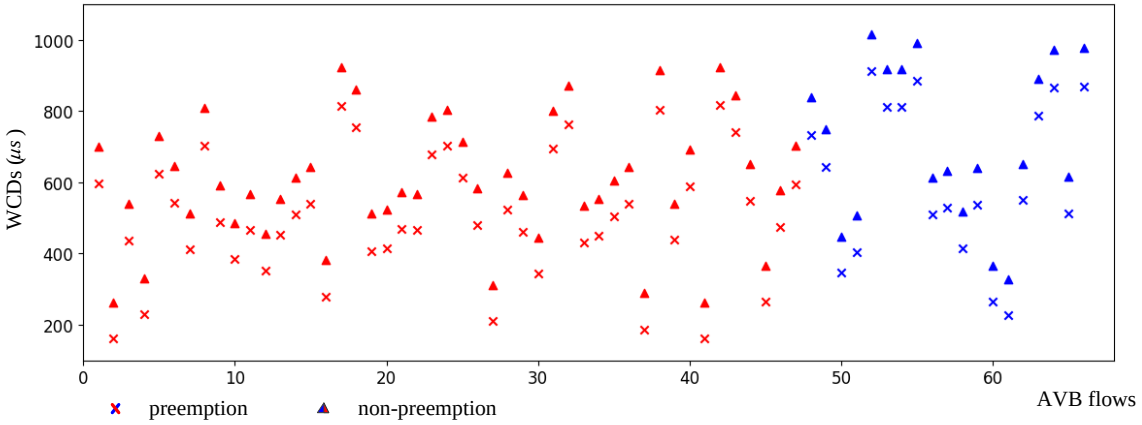

Fig. 13. Comparison of WCDs with preemption and non-preemption modes

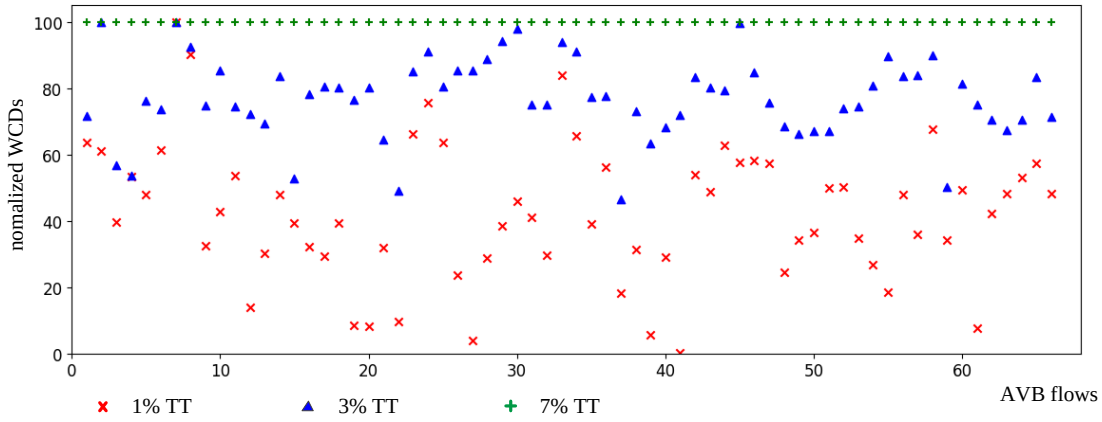

Fig. 14. Comparison of WCDs under different TT bandwidth utilization

number is related to the "porosity" of GCLs, which means the density (intensive or sparse) of GCL windows.

In order to verify the influence of TT windows on AVB traffic, we consider three test cases TC2, TC3 and TC4 with different bandwidth utilization of TT traffic, respectively of 1%, 3% and 7%. Here we assume that there are same AVB traffic flows in such three test cases and the TSN network is with the non-preemption integraion mode. Fig. 14 shows the compared results. In order to represent clearly, we use the WCDs of AVB traffic obtained under 7% bandwidth utilization of TT traffic as the baseline (the "+" symbols forming a horizontal line), i.e., the values on the y-axis show the percentage deviation of WCDs, which is normalized to 100. The upper bounds latency under other TT bandwidth utilization are normalized,

$$Nor^{x} = 100 - \frac{D^{7\%}_{AVB\_M_i} - D^{x}_{AVB\_M_i}}{D^{7\%}_{AVB\_M_i}} \times 100 \cdot \qquad (25)$$

where x represents bandwidth utilization of TT traffic. As shown in Fig. 14, since TT traffic has the highest priority, the higher bandwidth is used by TT traffic, the larger will be the WCDs of AVB traffic as expected.

VII. Conclusion and Future Work

This paper shows how the Network Calculus approach can be applied for the latency bounds of AVB traffic in TSN networks. Compared to the existing approaches, Network Calculus approach provides safe upper bounds on WCDs and reduces the pessimism.

The paper considers both integration modes, i.e., non-preemption and preemption modes in a TSN network. Also we model the influence of TT traffic windows controlled by the GCLs on the AVB traffic.

A first contribution of this paper deals with a proof of non-overflow condition for AVB credit based on general parameters of CBS, which is a pre-condition of timing analysis for AVB traffic.

A second contribution is the modeling of the arrival constraints of TT traffic based on the GCL windows, additional guard bands for the non-preemption mode and extra preemption overheads for the preemption modes.

A third contribution is that we derive the remaining service curve for AVB Class A and Class B respectively with the non-preemption and preemption integration modes.

In the end, this paper evaluates the proposed approach on realistic test cases, the Orion Crew Exploration Vehicle (CEV). Compared with the existing approaches, the approach in this

paper reduces the pessimism of estimated latencies, and can handle both AVB Class A and B.

REFERENCES

[1] IEEE, “802.3 Standard for Ethernet,” 2015.

[2] ARINC, “ARINC 664P7: Aircraft data network, Part 7. Avionics full-duplex switched Ethernet network,” 2009.

[3] SAE, “AS6802: Time-Triggered Ethernet,” SAE International, 2011.

[4] D. Jansen and B. Holger, “Real-time Ethernet: the EtherCAT solution,” Computing and Control Engineering, vol. 15, no. 1, pp. 16-21, 2004.

[5] D. Frank and N. G. Nayak, "No-wait packet scheduling for IEEE time-sensitive networks (TSN)," in Proc. of the 24th International Conference on Real-Time Networks and Systems, ACM, pp. 203-212, 2016.

[6] IEEE, “Time-Sensitive Networking Task Group,” http://www.ieee802.org/1/pages/tsn.html, 2016.

[7] IEEE, “802.1Qbv - Enhancements for Scheduled Traffic,” http://www.ieee802.org/1/pages/802.1bv.html, 2015.

[8] IEEE, “802.1Qbu - Frame Preemption,” http://www.ieee802.org/1/pages/802.1bu.html, 2015.

[9] IEEE, “802.1BA - Audio Video Bridging (AVB) Systems,” http://www.ieee802.org/1/pages/802.1ba.html, 2011.

[10] IEEE, “802.1ASrev - Timing and Synchronization for Time-Sensitive Applications,” http://www.ieee802.org/1/pages/802.1AS-rev.html, 2017.

[11] IEEE, “802.1Qav - Forwarding and Queuing Enhancements for Time-Sensitive Streams,” http://www.ieee802.org/1/pages/802.1av.html, 2009.

[12] IEEE, “802.1Qci - Per-Stream Filtering and Policing,” http://www.ieee802.org/1/pages/802.1ci.html, 2016.

[13] S. S. Craciunas, R. S. Oliver, M. Chmelik and W. Steiner, “Scheduling Real-Time Communication in IEEE 802.1Qbv Time Sensitive Networks,” in Proc. of the 24th International Conference on Real-Time Networks and Systems, 2016.

[14] R. Queck, “Analysis of Ethernet AVB for automotive networks using network calculus,” in Proc. of IEEE International Conference on Vehicular Electronics and Safety, pp. 61-67, 2012.

[15] J. A. R. Azua and M. Boyer, “Complete modelling of AVB in network calculus framework,” in Proc. of the 22nd International Conference on Real-Time Networks and Systems, pp. 55-64, 2014.

[16] A. Philip, D. Thiele, R. Ernst and J. Diemer, "Exploiting shaper context to improve performance bounds of ethernet avb networks," in Proceedings of the 51st Annual Design Automation Conference, pp. 1-6, ACM, 2014.

[17] J. Y. Cao, P. J. L. Cuijpers, R. J. Bril and J. J. Lukkien, "Tight worst-case response-time analysis for ethernet AVB using eligible intervals," in IEEE World Conference on Factory Communication Systems (WFCS), pp. 1-8, 2016.

[18] A. S. Vincentelli, H. Zeng, M. D. Natale and P. Marwedel, “Embedded Systems Development: from Functional Models to Implementations.” Springer Science & Business Media, 1st ed., 2013.

[19] L. X. Zhao, H. G. Xiong, Z. Zheng and Q. Li, "Improving worst-case latency analysis for rate-constrained traffic in the time-triggered ethernet network." IEEE Communications Letters 18(11) , pp. 1927-1930, 2014.

[20] M. Boyer, D. Hugo, N. Nicolas and M. Jörn, "Performance impact of the interactions between time-triggered and rate-constrained transmissions in TTEthernet," 2016. http://orbilu.uni.lu/handle/10993/23845.

[21] L. X. Zhao, P. Pop, Q. Li, J. Y. Chen and H. G. Xiong, “Timing analysis of rate-constrained traffic in TTEthernet using network calculus,” Real-Time Systems, vol. 53, no. 2, pp. 254-287, 2017.

[22] P. Meyer, S. Till, K. Franz and C. S. Thomas, "Extending IEEE 802.1 AVB with time-triggered scheduling: A simulation study of the coexistence of synchronous and asynchronous traffic," in IEEE Vehicular Networking Conference (VNC), pp. 47-54, 2013.

[23] S. M. Laursen, P. Pop and W. Steiner, “Routing Optimization of AVB Streams in TSN Networks,” ACM Sigbed Review, vol. 13, no. 4, pp. 43-48, 2016.

[24] U. D. Bordoloi, A. Aminifar, P. Eles and Z. Peng, “Schedulability analysis of Ethernet AVB switches,” in Proc. of the 20th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, pp. 1-10, 2014`.

[25] P. Pop, M. L. Raagaard, S. S. Craciunas and W. Steiner, “Design optimisation of cyber-physical distributed systems using IEEE time-sensitive networks,” IET Cyber-Physical Systems: Theory & Applications, 1(1), pp.86-94, 2016.

[26] J. Y. Le Boudec and P. Thiran, “Network calculus: A theory of deterministic queuing systems for the internet,” Springer-Verlag Lecture Notes on Computer Science, 5th ed., New York, 2001.

[27] N. Finn, “802.1Qav + P802.1Qbv Time-gated Shapers,” Cisco Systems, http://www.ieee802.org/1/pages/802.1bv.html, 2014.

[28] D. Tamas-Selicean, P. Pop and W. Steiner, “Design optimization of TTEthernet-based distributed real-time systems,” Real-Time Systems, vol. 51, no. 1, pp. 1–35, 2014.

[29] E. Wanderler and L. Thiele, “Real-Time Calculus (RTC) Toolbox,” http://www.mpa.ethz.ch/Rtctoolbox, 2006.

[30] W. Steiner, “Synthesis of static communication schedules for mixed-criticality systems,” in Proc. of the 14th IEEE international symposium on object/component/service-oriented real-time distributed computing workshops, Newport Beach, CA, pp. 11–18, 2011.