Advanced Modeling and Analysis of Individual and Combined TSN Shapers in OMNeT++

Rubi Debnath[1], Philipp Hortig[2], Luxi Zhao[3], and Sebastian Steinhorst[4]

[1,2,4]TUM School of Computation, Information and Technology, Technical University of Munich, Germany
[3]Beihang University, Beijing, China
[1]rubi.debnath@tum.de, [2]philipp.hortig@tum.de, [3]zhaoluxi@buaa.edu.cn, [4]sebastian.steinhorst@tum.de

***Abstract*—The selection of a Time-Sensitive Networking (TSN) shaping mechanism is a crucial design decision that impacts the Quality of Service (QoS) of applications and configuration complexity. However, current research has mainly evaluated TSN shapers individually, despite them being designed to work together in an egress port. Hence, the lack of investigation of the mixed TSN shaping mechanism is a major limitation of the current state of the art. Combined TSN traffic shaping provides greater flexibility to improve QoS than individual shapers, making it particularly beneficial for real-time applications. This paper aims to bridge this research gap by implementing the Asynchronous Traffic Shaper (ATS) in a plug-and-play manner, enabling its use individually or in combination with other TSN shapers. We propose various models of mixed TSN shaper architectures and implement the frozen and non-frozen credit behavior of the Time Aware Shaper (TAS) + Credit Based Shaper (CBS) during the guard band (GB) using OMNeT++. Furthermore, we compare the simulation results of ATS and CBS with the Network Calculus (NC) upper bounds. Our results indicate that the simulated delays (SMDs) were significantly lower than the theoretical worst-case delays (WCDs) obtained from the NC, indicating the need for tighter theoretical upper bounds, particularly for higher network loads. To the best of our knowledge, we are the first to provide simulation-based performance analysis of the combined TAS+ATS+CBS and TAS+ATS+Strict Priority (SP) architecture. Overall, this paper highlights the benefits of combining TSN shapers and encourages further research into the potential advantages of utilizing multiple shapers simultaneously to decrease reliance on TAS and CBS.**

***Index Terms*—time sensitive network, time aware shaper, audio-video bridging, credit based shaper, asynchronous traffic shaper, performance analysis, OMNeT++, real-time networks**

I. Introduction

The need for deterministic and bounded delays in critical traffic, such as those found in industrial automation, vehicular networks, cyber-physical systems, and the aerospace domain, has resulted in the widespread adoption of Time-Sensitive Networking (TSN) [1]. TSN, a layer-2 technology standardized by IEEE 802.1, includes multiple sub-standards and amendments to support different types of traffic, including time-critical and non-time-critical traffic, in a mixed-critical TSN network. To meet the latency and timing requirements, the IEEE 802.1 TSN Working Group (WG) has introduced shapers or schedulers, such as Time-Aware Shaper (TAS) [2] for Time-Triggered (TT) traffic also known as Scheduled Traffic (ST), Credit-Based Shaper (CBS) [3], [4] for audio-video bridging (AVB) traffic, and Asynchronous Traffic Shaper (ATS) [5] for asynchronous traffic.

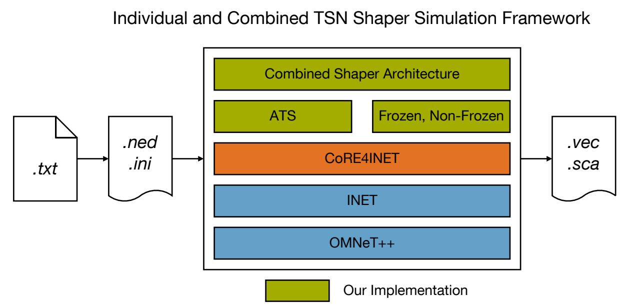


Fig. 1: Block diagram of our simulation framework, showing modules implemented by us in green.

ATS is the latest shaping mechanism standardized by the IEEE 802.1Qcr [5] and overcomes the problem of buffering and burstiness of flows sharing the same network resources. TAS on-the-other hand, widely studied by researchers, provides guaranteed end-to-end delay for higher priority critical traffic. However, the synthesis of the gate control list (GCL), also known as scheduling table, which is an NP-complete problem, poses a significant challenge. While many related works have focused on TAS GCL generation, it remains a challenging problem, particularly for very large networks. Another challenge with TAS is achieving global clock synchronization [6] throughout the entire network. In contrast, ATS is free from the global clock synchronization mechanism and scheduling table, making it easier to implement. Due to its low implementation complexity and predictable worst-case delays, ATS is a promising shaping mechanism. The real-world use case of ATS is currently under investigation due to the lack of detailed experiments and analysis. ATS was first introduced as Urgency-Based Scheduler (UBS) by Specht et al. in [7]. However, the ATS model in the IEEE 802.1Qcr sub-standard differs from the originally proposed UBS model. ATS is based on the principle of segregating per-flow queues and per-flow state, which Specht et al. referred to as interleaved shaping [7], the same as asynchronous traffic shaper. From the novel concept of UBS proposed in [7], the IEEE 802.1Qcr sub-standard was framed and named ATS. ATS provides bounded

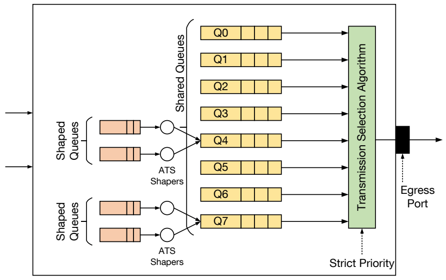


Fig. 2: Block diagram of the ATS mechanism, highlighting the shaped queue and the ATS shaper that assigns eligibility time to packets before enqueuing into the shared queue.

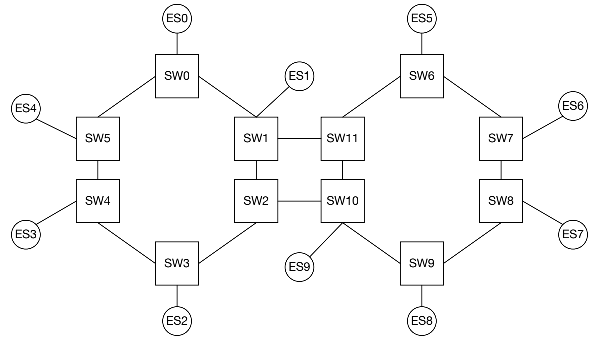


Fig. 3: Industrial ring topology.

end-to-end delay without the additional complexity of global clock synchronization and schedule generation required for TAS. CBS allows rate allocation to the assigned AVB streams, and the credit value in the CBS algorithm determines the transmission state of the frame waiting in the queue. SP on the other hand serves the traffic based on the priority.

A. Problem and Motivation

TSN supports the use of multiple shapers in the same egress port, enabling system engineers to choose from various combinations of shapers. TAS is well suited for safety critical traffic types, however, using TAS for all high priority traffic types is not optimal due to the complexity of GCL synthesis. Additionally, not all high priority traffic types have strict deadline constraints, making CBS or ATS a better choice for such traffic types. Therefore, detailed evaluation and performance analysis of TSN shaping mechanisms are crucial for the successful usage of TSN in different domains such as automotive vehicular networks, industrial automation, and the wireless TSN domain. However, most of the related work has assumed a single traffic shaper in the network, leaving a research gap in the detailed analysis of all the shapers. Furthermore, there have been very limited studies on the combination of different shapers. After thorough evaluation, we realized that the simulation approach on individual and combined traffic shapers is still sparse. Although INET4.4 [8] has recently released a version with TSN capabilities, it does not consider the simulation of TAS+ATS+CBS, and TAS+ATS+SP. Another popular framework for TSN, CoRE4INET [9], does not support ATS and does not consider the TAS+CBS frozen/non-frozen mechanism. Furthermore, NeSTiNg [10], another TSN library, currently does not support CBS and ATS. On the other hand, INET4.4 does not support two shapers on the same queue as proposed in [11]. Although having two shapers on the same queue is not supported by the standard, it would be interesting to support the simulation of such combinations.

The main contributions of this paper are as follows:

Contribution 1: We extend the support for various combinations of mixed TSN shapers (Section IV) through the implementation of a flexible plug-and-play ATS shaper (Sub-section III-B). This includes CBS+ATS on different and same queues, TAS+ATS+CBS, and TAS+ATS+SP. Through our simulation studies (Section V), we provide comprehensive evaluation and performance analysis of these TSN shapers individually and in combination.

Contribution 2: We implement two integration mechanisms of TAS+CBS, which are frozen and non-frozen credit behavior during the guard band (GB). We conduct a performance analysis of AVB traffic in the TAS+CBS architecture under these two mechanisms. Our simulation results show that non-frozen credit behavior provides lower end-to-end delay bounds for AVB traffic (Section V). Additionally, we emphasize the need for both frozen and non-frozen simulation options in open-source tools to support future research.

Contribution 3: We compare ATS and CBS end-to-end delay bounds using two techniques: Network Calculus (NC) and simulation. Our results emphasize the need for tighter theoretical worst-case delay bounds to prevent the wastage of resources in safety-critical applications.

Contribution 4: Finally, we demonstrate the advantages of using ATS in industrial networks, for industrial traffic types (Sub-section V-F). Our findings suggest that ATS can effectively be used for Alarms & Events (A&E) and Configuration & Diagnostics (CoDi) traffic types (Section VI).

In this paper, we have implemented ATS in a flexible manner using CoRE4INET framework in OMNeT++, allowing it to be easily integrated with any other shaper in series (same queue) or parallel (different queue). This flexibility enables us to investigate mixed TSN shaping mechanisms and evaluate the benefits of combining multiple shapers simultaneously, which is not possible with other existing implementations such as the latest version of INET4.4, CoRE4INET and NeSTiNg. We have made the test cases used in this paper openly accessible for further evaluation[1]. The open access to our test cases will enable other researchers to easily replicate and build upon our work. We believe that our study will help system engineers and researchers to make informed decisions when selecting TSN shapers and designing TSN networks for specific use cases. Moreover, our simulation results and insights can guide the performance evaluation of all TSN shapers.

[1] https://github.com/tum-esi/Individual-and-Combined-TSN-Shapers

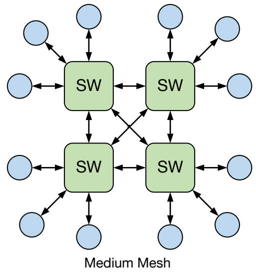


Fig. 4: Medium Mesh (MM) topology.

II. Related Work

[12] presented the comparison between TAS and ATS. However, [12] considered an urgent queue for the ATS bridge operation which is different from the originally proposed model [7] and the IEEE 802.1Qcr standard [5]. This urgent queue change affects the latency bound results of the flows and hence does not provide a clear performance evaluation of the ATS shaper. In [13], researchers analyzed ATS using the Leaky Bucket and Paternoster algorithm, but the published ATS standard is based on the Token Bucket Algorithm, which is different from the approach used in [13]. In our work, we have implemented the ATS shaper as per the published standard to ensure that our results are aligned with the latest specifications. In [14], researchers compared and overviewed different TSN shapers for automotive Ethernet-based applications, including TAS, ATS, SP, and CBS, by creating testing models based on OMNeT++ and the existing CoRE4INET framework. However, the model of ATS in [14] is not fully aligned with the standard, and the paper only covers AVB class A and B traffic types simulation without considering combined traffic shaping mechanisms. In our work, we consider the coexistence of multiple shapers in the same egress port, which differs from the approach used in [14]. In [15], Mohammadpour et al. computed the bounded end-to-end worst-case delay for CBS and ATS and provided tight delay bounds for ATS. Their work considered class A and class B traffic shaped by ATS and CBS, but simulation results were not covered in [15]. By contrast, in our work, we present the simulation results and conduct a comparison between ATS and CBS when used together in the same egress port in different queues. In [16], Fang et. al. compared the performance of CBS, ATS, and Strict Priority (SP) by implementing the models of ATS and CBS using OMNeT++. However, their ATS model did not match the originally proposed model in [5], [7]. They used ATS shaper to shape class A and B traffic types and placed the shared or priority-based queue before the ATS shaper and shaped queue, which is different from the IEEE 802.1Qcr standard. Moreover, [16] implemented ATS mechanisms in both the *ES* and *SW* modes and did not consider TT traffic in the network. [16] showed that ATS performs better for aperiodic traffic types compared to CBS in high network loads. In our work, we consider the effect of TT traffic in the mixed shaper architecture and evaluate the performance of different architectures. It is also important to note that the simulation conducted by [16] did not consider bursty traffic and the traffic type parameters used in their study do not fully adhere to the requirements of industrial automation traffic characteristics. Furthermore, [16] only evaluated the performance of individual shaper CBS, ATS, and the combined shaper CBS+ATS on the same queue. In contrast to [16], our work also considered bursty industrial automaton traffic and an industrial network topology (see Fig. 3) to ensure that the traffic type parameters adhere to the requirements of industrial automation traffic characteristics. Moreover, we investigate the impact of TT traffic on the performance of ATS and CBS mechanisms. In addition, we utilized ATS+CBS in the same queue for a more complex architecture, which is not currently supported by the standard. However, as discussed in [11], this approach is worth further research. Luo et. al. in [17] studied the performance of automotive TSN under different shaping mechanisms, including TAS and SP, using OMNeT++. They used a real-world in-vehicular topology and considered vehicular communication traffic types, but did not provide the results of ATS performance in automotive TSN networks in their work, leaving this for future study. Additionally, the joint mechanism of TSN shapers was not covered in their work. Arestova et. al. presented the performance analysis of TAS and frame preemption (FP) [18] in [19]. [19] created a test setup to verify the TAS, FP [18] and BE traffic for large-scale industrial networks, and used OMNeT++ and NeSTiNg libraries for their analysis. However, their work did not cover ATS, CBS, or any of their combined mechanism.

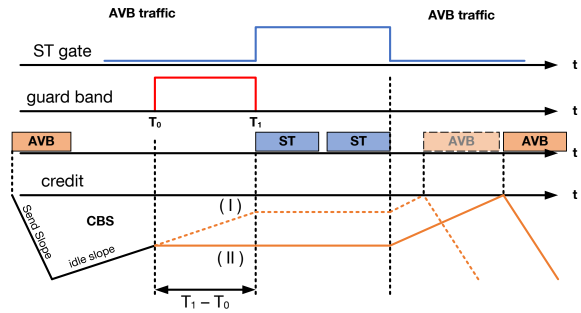


Fig. 5: Non-frozen (I) and frozen (II) credit accumulation during guardband (GB).

[20] presented a study on the impact of the credit freeze before GCL closing and compared three methods including the frozen, non-frozen and a third one as a hybrid approach called return to zero. The simulation tools used to obtain the results presented in [20] have not been explicitly stated. Additionally, [20] has not explored the impact of varying traffic loads on the delay results. Furthermore, the use of only one switch in the simulation results of [20] has limited the size of the network, which may not accurately reflect real-world scenarios. [21] proposed a network calculus-based worst-case delay (WCD) of AVB traffic under the influence of TT traffic in the network. [11] provided an initial study of different combined shapers using a quantitative approach using Network Calculus (NC) beyond the TAS and CBS shaper. However, as shown in our

paper, NC results are overly pessimistic making our simulation results equally significant for the performance study of any TSN network.

In summary, our paper implemented ATS in a flexible plug-and-play fashion, which can be used both individually and in combination with other shapers such as ATS+CBS, TAS+ATS+CBS, and TAS+ATS+SP, providing a joint shaper simulation architecture and results. We also implemented frozen and non-frozen integration modes of TAS+CBS, yielding valuable simulation results. Additionally, we demonstrated the practical applicability of ATS in an industrial network.

III. System Model

In this section, we describe our simulation model as well as the implementation details.

A. Model Overview

A TSN network consists of bridges (TSN switches), links (edges), and end-stations (ESs), where ESs can act as talkers, listeners, or both. Flows are streams of information sent and received by ESs, forwarded through TSN switches (SWs) on links, and can be of any traffic type, such as TT, AVB Stream A, AVB Stream B, or Best Effort (BE), with TT having the highest priority followed by class A and class B traffic, and BE having the lowest priority. SWs are equipped with physical ports that allow ESs to connect to them using Ethernet cables. The egress port of SWs consists of eight egress queues, each with an assigned priority (P_f) from 0 (lowest) to 7 (highest). Our TSN network supports different traffic types, and we can assign eight different priorities to them. We assumed that the traffic type and priority for each application were decided beforehand. For TT traffic, we used the GCL (i.e., the opening and closing time) from [22]. We evaluated the maximum, minimum, and average end-to-end delay as performance parameters and compared the simulation results with the NC based upper bound WCD to highlight the importance of the simulation approach. The end-to-end delay is defined as the total time taken for a data frame to travel from the sending node to the receiving node, including all queuing, processing, and transmission delays. We used synthetic test cases (periodic and non-sporadic flows) for Medium Mesh (MM) topology shown in Fig. 4. Since the related works have not covered the performance of ATS on industrial networks for industrial automation traffic types, we also analyzed the performance of ATS on an industrial network (Fig. 3) adhering to the industrial automation traffic types (Table I) under the non-preemption mode. We randomly selected 5 synthetic test cases from a pool of 130 cases for each scenario. The payload size of the TSN flows was chosen randomly between 64 and 1,522 Bytes. The simulated network link speed was set to the common industry standard of 100 Mbps. The hardware delay of the ES and the SW were set to 0 and 8μs, respectively. Additionally, we set the processing delay of the switch to a value of 4μs. In contrast, for the industrial network, we determined the payload and period of the traffic types based on the industrial traffic requirements. To obtain the simulation results, we ran simulations using OMNeT++ for a duration of 10 seconds in simulation clock. The simulations were conducted on a laptop equipped with an Intel Core i7 processor with 4 cores and 32GB of RAM, running the Windows 10 operating system.

B. Implementation

The proposed ATS implementation is illustrated in Fig. 2. We have designed the ATS as an independent module, allowing it to be utilized in combination with other shapers in the same or different queue. To validate our ATS implementation, we compared the simulated delay (SMD) of ATS with the NC WCD (shown in Fig. 7). All the NC WCD results are taken from [11], and since all the experiments and test cases are the same for the NC and Simulation methods, a direct comparison is possible. The entire simulation framework is shown in Fig. 1. The implemented ATS shaper includes various modules: *InputInterface*, *IEEE8021QcrShapedQueue*, *IEEE8021QcrShaper*, and *AtsOutputSelection*. The *InputInterface* module filters the traffic and sends it to the appropriate shaped queue. We call our shaped queue module *IEEE8021QcrShapedQueue*. The *IEEE8021QcrShaper* module assigns eligibility time to the frames stored in the shaped queue and dequeues them to the output interface (*AtsOutputSelection*) when they become eligible for transmission. The frames are then sent to the shared queue or priority queue. To enable advanced mixed shaper architecture, we implemented two integration modes for TAS+CBS: frozen and non-frozen credit during GB, in addition to the ATS shaper. To facilitate switching between the two modes, we configured them in the *.ini* files. To model various mixed shaper architectures, we have created multiple TSN switches with support for different mixed shaper architecture (refer Fig. 6). Depending on the test cases and the architectures we want to evaluate, we use the corresponding TSN switch and TSN node.

IV. Simulative Evaluation of Combined Traffic Shapers

In this section, we discuss the combined usage of TSN shapers and their combined architecture in detail.

A. TSN/TAS+CBS

Most related works simulate TAS and CBS individually, and they do not consider the effect of TT traffic on AVB traffic in the TSN network. Therefore, we begin by discussing the TSN/TAS+CBS combined shaper mechanism. While this architecture is not new, there is a lack of study and simulation results that support this mechanism. In this paper, we provide the simulation results of TAS+CBS under two distinct conditions: frozen credit behavior and non-frozen credit behavior during GB. Fig. 5 illustrates the mechanism of both frozen and non-frozen behavior. The combined use of TAS and CBS mechanisms, depicted in Fig. 6a, allows for the assignment of time windows for high-priority TT traffic in the network, ensuring hard latency boundaries while enabling lower-priority AVB traffic. However, due to the exclusive switching of the

TABLE I: Traffic type parameters summary (per IEC/IEEE 60802 [23])

Traffic Type	Periodicity	Period	Data Size (in Bytes)	Criticality	PCP
Alarms & Events (A&E)	Sporadic	NA	Variable (50∼1500)	Medium	3
Configuration & Diagnostics (CoDi)	Sporadic	500ms∼2s	500∼1500	Medium	2

gates, there are periods during which AVB traffic transmission is at rest, necessitating an adjustment of CBS behavior to ensure correct credit accumulation. According to the standard, credit is accumulated until the gate closes, and the credit is only frozen while the gate is closed. However, the literature considers the credit accumulation to be frozen during the GB, as opposed to the standard.

1) Frozen Credit: In frozen credit behavior, credit accumulation is paused during both active GB and TT transmission windows. This leads to an upper bound credit value similar to that in a CBS-only TSN network. This behavior was assumed in previous studies, as mentioned in [20].

2) Non-Frozen Credit: In non-frozen credit behavior (standard behavior), credit accumulation is allowed during GB. However, credit accumulation stops during gate closing.

B. TSN/TAS+SP

We now briefly discuss the TAS+SP architecture shown in Fig. 6b. Since the TT traffic has the highest priority, it is not interfered with by SP traffic, and thus, the TT traffic performance remains unchanged in the TAS+SP combined architecture similar to the TSN/TAS+CBS architecture. The SP scheduling dequeues flows based on their priority, so there is no condition of frozen or non-frozen credit during GB in the TAS+SP architecture. The GCL generation of the TAS includes the added GB to prevent the transmission of SP flows when a TT frame is already in transmission. This feature is independent of the simulation method, as it is covered during the GCL calculation. In the combined TSN/TAS+CBS or TSN/TAS+SP architecture, the TT traffic performance remains the same due to the pre-calculated schedule table. The use of CBS and SP on lower priority traffic does not interfere with the TT traffic. However, the traffic shaped by CBS or SP has different performance due to the presence of TT traffic in the network. Therefore, in this architecture, we focus only on the performance of the SP shaper, similar to TSN/TAS+CBS, where we only looked into the performance of the CBS shaper.

C. TSN/CBS+ATS on different queues

The network architecture that utilizes both CBS and ATS is referred to as TSN/CBS+ATS or TSN/ATS+CBS, as shown in Fig. 6d. In this architecture, different queues of the same egress port are shaped by ATS and CBS, respectively. The ATS eligibility time parameter determines the eligible time for frames in ATS-shaped queues, while the standard CBS algorithm is used in CBS-shaped queues. In this paper, we have not provided detailed explanations of the ATS token bucket algorithm and the CBS algorithm. Interested readers can refer to IEEE 802.1Qcr [5] and IEEE 802.1BA [3], respectively.

D. TSN/CBS+ATS on same queue

In this architecture, both ATS and CBS shapers are used on the same queue. Although the current standard does not support the use of two shapers on the same queue [11], previous research has encouraged the study of using both shapers [11]. Therefore, to test such an architecture, we implemented the dual shaping mechanism using OMNeT++. Keeping the ATS shaper as a flexible plug-and-play shaper provides the degree of freedom to use multiple shapers in the same queue. The TSN/CBS+ATS in series architecture is shown in Fig. 6c, where flows are first shaped via ATS and then by CBS.

E. TSN/TAS+ATS+CBS

In this sub-section, we discuss the architecture of the TSN/TAS+ATS+CBS, as shown in Fig. 6e. Since TT traffic shaped by TAS performs the same, the goal of this architecture is to provide insights into the performance of lower priority traffic shaped by ATS+CBS in the same queue. Although two shapers on the same queue are not supported in the standard, we present simulation results of this combination to encourage flexible ATS implementation. The GCL generation algorithm will remain the same for TSN/TAS+CBS and TSN/-TAS+ATS+CBS, as ATS and CBS are used in the same queue. When comparing TSN/TAS+CBS with TSN/TAS+ATS+CBS, an additional shaped queue and ATS shaper are present before the CBS shaper. The frames are stored in the priority queue after ATS shaping, and then the CBS algorithm is used to dequeue the frames from the shared queue.

F. TSN/TAS+ATS+SP

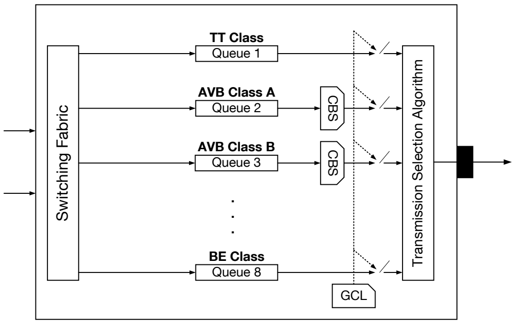
The TSN/TAS+ATS+SP architecture shown in Fig. 6f is similar to the TSN/TAS+ATS+CBS architecture, with the only difference being the use of an SP shaper instead of CBS. In TSN/TAS+ATS+SP, an additional shaped queue and ATS shaper are present before the SP (priority) queue, compared to TSN/TAS+SP.

V. Experimental Results

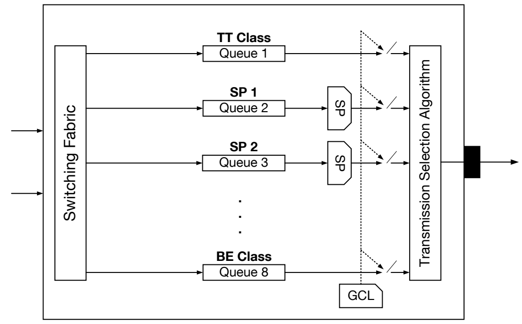
Having explained the different architectures, now we move on to presenting the experimental results obtained from our simulations using OMNeT++. We discuss the findings of our various experiments in detail, including the performance evaluation of individual TSN shapers and mixed TSN shapers, and the impact of varying network load on the overall end-to-end delay performance.

A. ***Individual Traffic Shapers***

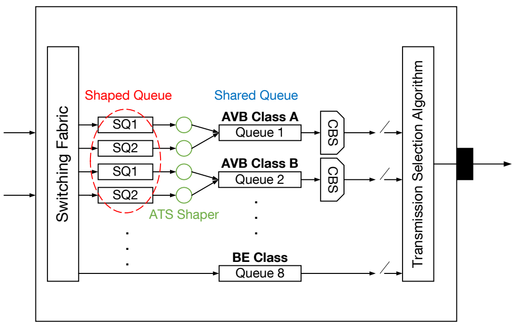
In this experiment, we evaluated the performance of four different traffic shapers, namely TAS, ATS, CBS, and SP, in terms of minimum, average and maximum end-to-end delay. To analyze the results, we utilized a violin plot with



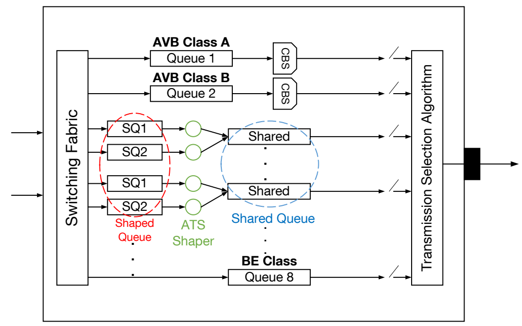
(a) Architecture of **TAS+CBS** mixed TSN shaper with TAS and CBS used in parallel in different queues.



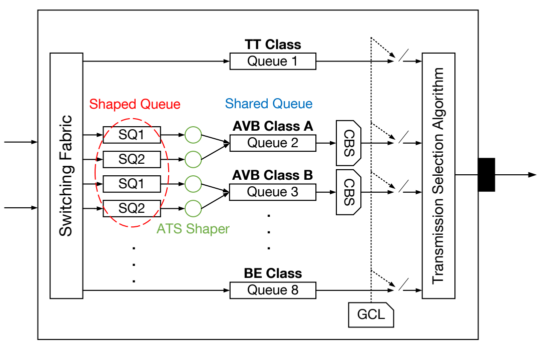
(b) Architecture of **TAS+SP** mixed TSN shaper with TAS and SP used in parallel in different queue.



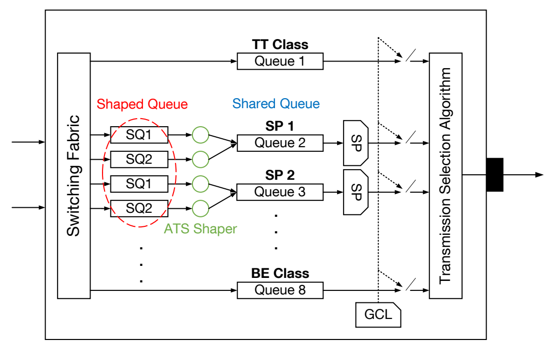
(c) Architecture of **ATS+CBS in Series** mixed TSN shaper, using ATS and CBS in the same queue.



(d) Architecture of **ATS+CBS in Parallel** mixed TSN shaper, using ATS and CBS in different queues.



(e) Architecture of **TAS+ATS+CBS** mixed TSN shaper.



(f) Architecture of **TAS+ATS+SP** mixed TSN shaper.

Fig. 6: Different combinations of TSN shaper architectures, including TAS, ATS, CBS, and SP shapers.

quartiles to visualize the distribution of the end-to-end delay measurements for each traffic shaper, as shown in Fig. 8. The quartiles divide the data into four equal parts, with each quartile representing 25% of the data. The bottom quartile represents the 25% of data with the lowest end-to-end delay values, the median quartile represents the 50% of data with the middle end-to-end delay values, and the top quartile represents the 25% of data with the highest end-to-end delay values. The violin plot reveals that TAS outperforms all other TSN traffic shapers in terms of end-to-end delay. TAS exhibited a concentrated distribution with the top quartile line (75^{th} percentile) being close to the median (50^{th} percentile), indicating that 50% of the end-to-end delay measurements of TAS lie within a relatively narrow range, suggesting low variability. Consequently, TAS efficiently meets stringent deadline requirements and is well-suited for applications with tight jitter constraints. Furthermore, the lower, median, and top quartile of TAS exhibited the lowest values among all traffic shapers including CBS, ATS, and SP, indicating that TAS outperforms all other traffic shapers in terms of meeting stringent deadline requirements and achieving low end-to-end delay. The violin plot for CBS showed a relatively thin and tall distribution above the top quartile line (75^{th} percentile) in minimum and average end-to-end delay, suggesting that the data points in this region are sparse and have lower frequency. Interestingly, in the case of maximum delay for CBS, the lower quartile is relatively far from zero, indicating that there are flows with high SMD values. The violin plot for maximum end-to-end delay in CBS also revealed that the upper quartile is close to the median, indicating that the end-to-end delay values are

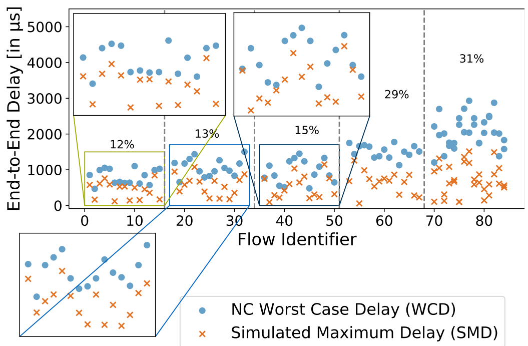


Fig. 7: Comparison of OMNeT++ SMD and NC WCD for ATS shaper under different network loads (12%, 13%, 15%, 29%, 31%) in MM Topology. Both SMD and WCD increase with an increase in the network load, and their difference significantly increases at higher network loads.

more evenly distributed in this region. For maximum end-to-end delay, the lower quartile of ATS was higher than CBS and almost equal to SP, indicating that the minimum end-to-end delay for ATS is relatively high. The upper quartile of ATS was far from the median, indicating that the maximum end-to-end delay values for ATS have low frequency and are sparse. By contrast, the median of SP was higher than ATS, suggesting that the majority of data points for SP have higher end-to-end delay values than ATS. Furthermore, the upper quartile of SP was also higher than ATS, indicating that the maximum end-to-end delay values for SP are more spread out than ATS. Overall, the results of this experiment demonstrate that TAS is the most efficient traffic shaper for achieving low end-to-end delay while meeting stringent deadline requirements. CBS performs better than ATS and SP, with CBS reserving 75% of the bandwidth for AVB flows resulting in faster credit recovery for the CBS algorithm. However, it is challenging to identify a clear winner between ATS and SP for periodic and non-sporadic traffic types.

B. *TSN/TAS+CBS*

In this experiment, we aimed to investigate the performance of the CBS mechanism under two different conditions - frozen credit behavior during GB (as commonly used in most papers, as mentioned in [20]) and non-frozen credit behavior during GB (as specified in the 802.1Q standard [1]). As mentioned earlier, the reason for choosing this combined architecture was to address the research gap in simulation results comparing these two scenarios. Our results showed that the delay of the AVB flows is higher with the frozen credit behavior during the GB period. Furthermore, our simulation results demonstrated that the SMD of AVB flows is on the lower side with the non-frozen credit behavior, as shown in Fig. 10. These findings suggest that the credit recovery behavior during the GB period can significantly affect the performance of CBS in the TSN/TAS+CBS shaper architecture. The frozen credit behavior increases the delay of the AVB flows, thereby indicating that it is not an optimal solution.

C. *TSN/ATS+CBS on different queues*

We considered two test case scenarios with different priority settings for the TSN/ATS+CBS architecture.

1) Scenario 1: For scenario 1, we assigned higher priority traffic to the ATS shaper and lower priority to the CBS shaper. We used the MM topology to test the performance, with four traffic priorities in the network ranging from PCP 7-4. The traffic types with PCP 7, and 6 were shaped by ATS, while those with PCP 4, and 3 were shaped by CBS.
2) Scenario 2: In scenario 2, we assigned higher priority traffic to the CBS shaper and lower priority traffic to the ATS shaper. Once again, we used the MM topology to test the performance, with traffic types with PCP 7, and 6 shaped by CBS and traffic types with PCP 4, and 3 shaped by ATS.

The performance of ATS is better than CBS when used for two different priorities. When using CBS for multiple traffic types, bandwidth division is applied between them. CBS performed better than ATS when we compared the individual shapers in Fig. 8. However, when there are two different traffic types reshaped by CBS, the *idleSlope* is set to 45% and 30% for class A and B, respectively, resulting in slow credit increase. This leads to larger end-to-end delay as shown in Fig. 11. When we further use ATS and CBS for three different AVB classes A, B, and C, the performance of ATS for class A, B and C is far superior (Fig. 12). This is again due to the slow credit recovery of the CBS since the *idleSlope* is now further set to 40%, 20%, and 15% for class A, B, and C, respectively.

D. *TSN/TAS+ATS+CBS*

In this study, we compared the performance of the TAS+ATS+CBS architecture with the TAS+CBS architecture. We used an ATS shaper that was designed in a flexible, plug-and-play manner to be used with any shaper as required. Our objective was to investigate whether using ATS in the same queue as CBS in TAS+CBS can help improve the latency bounds of lower priority traffic. We used frozen credit behavior and the load of AVB traffic was kept below the reserved BW (75%). It is noteworthy that the performance of TT traffic remained the same as TAS in TAS+any other shaper experiment, and hence, we did not focus on the performance of TT traffic in this experiment as it would be the same as using individual TAS in the architecture. Our results, as shown in Fig. 13, indicate that TAS+CBS performed better than TAS+ATS+CBS. CBS was found to be superior to ATS when used for a single traffic individually. Adding ATS before CBS provided no further delay improvements, as CBS is using 75% bandwidth for *idleSlope* calculation, resulting in faster credit recovery already. The positive difference between the SMD of TAS+ATS+CBS and TAS+CBS, as observed in Fig. 13, indicates that the use of ATS and CBS in the same queue may not always be beneficial. Therefore, further investigation is necessary to determine the specific use cases in which the joint use of ATS and CBS can lead to improved performance.

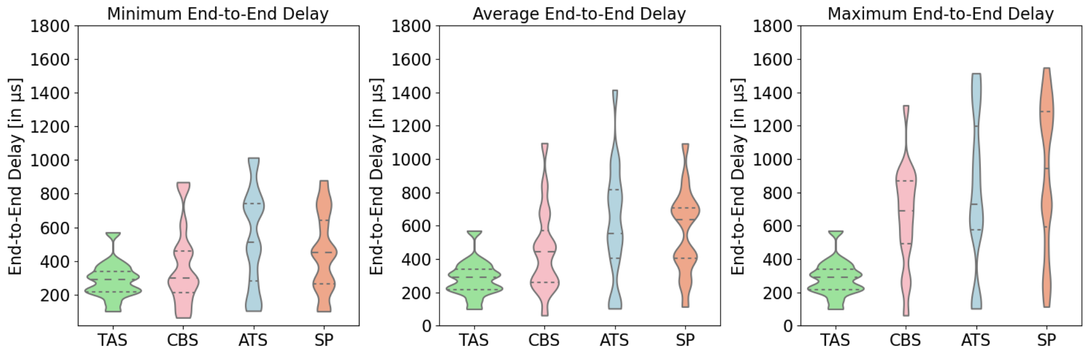


Fig. 8: Comparison of minimum, average, and maximum SMD for individual traffic shapers in synthetic test case under MM topology. TAS outperforms other shaping mechanisms with least SMD, but no clear winner among CBS, ATS, and SP.

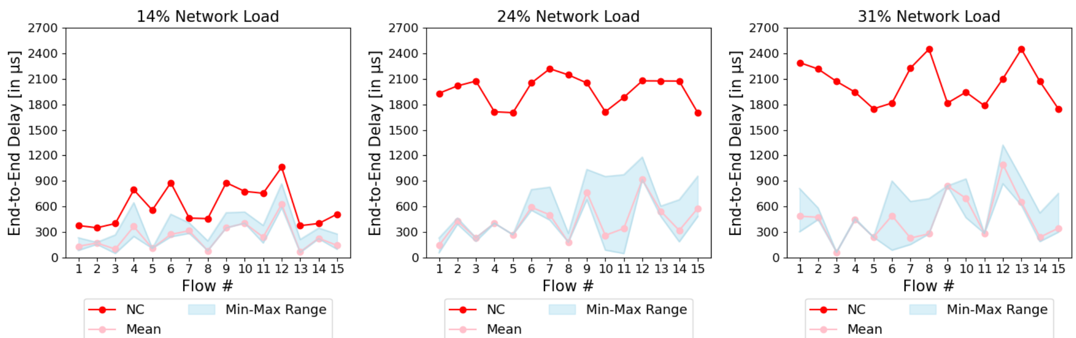


Fig. 9: Comparison of CBS End-to-End Delay under different network load for MM topology. The red line represents the theoretical upper bound provided by NC. The minimum, maximum, and average delay are generated by the simulation. As network load increases, end-to-end delay increases. Furthermore, the SMD of CBS is significantly lower than the NC WCD, particularly under high load, suggesting theoretical upper bound is overly pessimistic, especially for higher network loads.

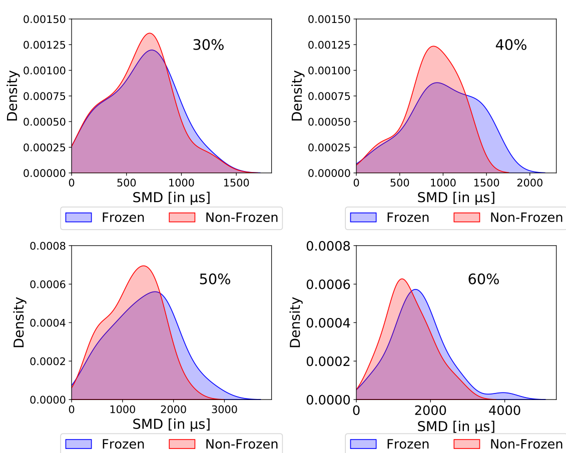


Fig. 10: Comparison of CBS SMD under frozen and non-frozen credit behavior for different network loads (30%, 40%, 50%, 60%) for Class A traffic in MM topology. Results show SMD increases with network load for both behaviors, but non-frozen yields lower SMD than frozen. Using non-frozen can lead to better CBS performance in TAS+CBS architecture.

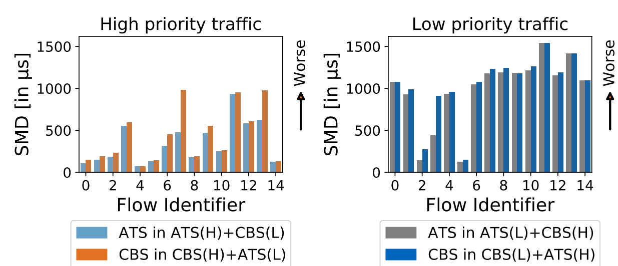


Fig. 11: The experiment uses 4 traffic types, categorized as high and low, with 2 traffic types shaped by CBS (*idleSlope* is set to 45% and 30%) and 2 shaped by ATS under MM topology. Results show that regardless of where ATS is used in the network, it consistently outperforms CBS.

E. TSN/TAS+ATS+SP

Here, we compared the performance of TAS+ATS+SP vs TAS+SP to understand the impact of ATS on the TAS+SP architecture. The performance of TT traffic remains the same as with the individual TAS shaper due to the use of GCL. However, we focused on the effect of ATS on the lower priority traffic when used in combination with SP. TAS+SP performed better than TAS+ATS+SP shown in Fig. 14. The difference between the SMD of TAS+ATS+SP and TAS+SP was positive as seen in Fig. 14. This suggests that the use of ATS and SP in the same queue may not always be beneficial and should be investigated further for specific use cases.

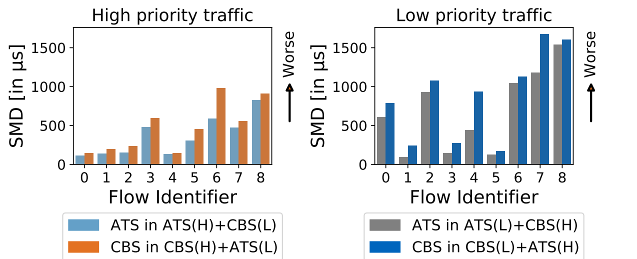


Fig. 12: SMD comparison for ATS+CBS under MM topology with (1) ATS for high-priority and CBS for low-priority traffic, and (2) ATS for low-priority and CBS for high-priority traffic. ATS consistently outperforms CBS in SMD, regardless of its usage for high or low priority traffic. Experiment includes 6 traffic types with AVB class A,B and C (*idleSlope* set to 40%, 20%, and 15%). CBS performs worse when all three classes are shaped by CBS due to slower credit recovery.

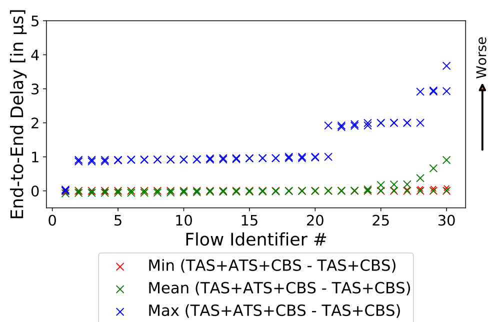


Fig. 13: Scatter plot showing the difference in the min, mean and max SMD values between: TAS+ATS+CBS and TAS+CBS under MM topology. The y-axis shows the difference between TAS+ATS+CBS and TAS+CBS SMD values, measured in μs. Positive y-axis values indicate TAS+ATS+CBS does not exhibit superiority over TAS+CBS.

F. Industrial Use Cases

Lastly, we investigated the performance of ATS in an industrial topology using traffic types commonly found in industrial applications. The experiments were conducted on an industrial ring topology, as shown in Fig. 3, and the traffic types were sourced from the draft version of IEC/IEEE 60802 [23]. The motivation behind this experiment is to analyze the impact of using ATS on specific industrial traffic types. Since the usage of ATS in industrial traffic types has not been covered before, this set of experiments focuses on the performance of bursty and sporadic traffics, particularly the Alarms & Events (A&E) and Configuration & Diagnostics (CoDi) traffic types when shaped by ATS. The A&E traffic type is a bursty traffic with showers of alarm events, up to 2000 alarms per second. On the other hand, CoDi traffic is sporadic in nature, and the period of the traffic was randomly selected between 500ms to 2s, with the data size varying between 500-1500 Bytes, as given in Table I. For the A&E traffic type, the maximum burst size is set to 1500×2000 Bytes.

As TAS is not designed for sporadic and bursty traffic, in

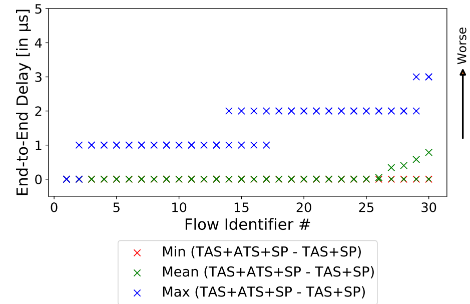


Fig. 14: Scatter plot showing the difference in the min, mean and max SMD values between TAS+ATS+SP and TAS+SP under MM topology. The y-axis displays the difference between the SMD values of TAS+ATS+SP and TAS+SP, measured in μs. All the y-axis values are positive, indicating that TAS+ATS+SP does not show superiority over the TAS+SP.

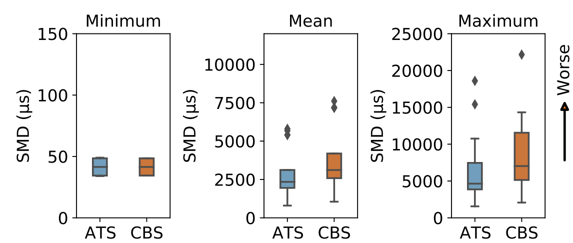


Fig. 15: Comparison of min, mean, and max end-to-end delay of 'Alarms & Events' traffic shaped by ATS and CBS individually in industrial ring topology. CBS *idleSlope* set to 75%. Results show ATS outperforms CBS for bursty and sporadic traffic, despite CBS reserving 75% of bandwidth.

this study, we only focused on evaluating the performance of ATS with A&E and CoDi commonly found in industrial applications. We used ATS individually to assess its effectiveness in shaping A&E and CoDi. Our results, as presented in Fig. 15 and Table II, demonstrate that ATS can meet the latency requirements of both A&E and CoDi, making it suitable for bursty and aperiodic A&E as well as CoDi. Until now, A&E and CoDi traffic types were recommended to be shaped with SP. However, with the related work demonstrating that ATS performs better than SP in high network load scenarios and our experimental results indicating that ATS could be a suitable choice for A&E and CoDi traffic type, we suggest that ATS should be considered as an alternative shaping method for A&E and CoDi traffic in an industrial network.

VI. Conclusion

This paper presented the implementation of ATS shaper in a flexible plug-and-play manner to support combined shaper architecture and provided a comprehensive performance evaluation of the different TSN shapers through simulation using OMNeT++. The paper showed that the quantitative Network Calculus based results of the TSN shapers are pessimistic in nature and highlighted the need for tighter theoretical delay bounds. Our experimental results showed that TAS

TABLE II: End-to-End Delay statistics of 'Configuration & Diagnostics' (CoDi) traffic shaped by ATS under industrial network topology. Table summarizes min, mean, and max delay values in μs and evaluates against QoS requirements. Results demonstrate ATS's ability to meet CoDi's QoS requirements

Traffic Type	Min Delay[in μs]	Mean Delay[in μs]	Max Delay[in μs]	Latency Constraint[in s]	Meets QoS Requirement
CoDi	174.79	270.44	338.23	Latency $<$ 1s	✓
CoDi	239.88	380.72	501.32	Latency $<$ 1s	✓
CoDi	145.03	243.96	369.91	Latency $<$ 1s	✓
CoDi	189.05	256.85	322.39	Latency $<$ 1s	✓
CoDi	242.12	391.15	500.04	Latency $<$ 1s	✓
CoDi	395.50	556.78	744.46	Latency $<$ 1s	✓

outperforms ATS, CBS, and SP in terms of latency for periodic and non-sporadic traffic. Our study revealed that ATS is well suited for sporadic and bursty traffic types such as Alarms & Events and Configuration & Diagnostics in industrial networks, which was not demonstrated in the previous research. We strongly agree that ATS has great potential as a shaper for industrial use cases, particularly for higher traffic loads. Using OMNeT++, we further implemented the frozen and non-frozen TAS+CBS integration method and conducted a comprehensive evaluation of the two mechanisms. Our results showed that non-frozen credit behavior is more suitable for safety-critical applications due to its lower latency values. Subsequently, we presented mixed shaper architectures, CBS+ATS on both different and same queues and TAS+ATS+CBS/TAS+ATS+SP architecture. Our evaluation revealed that when applying ATS and CBS to separate queues for high and low priority traffic types, ATS outperformed CBS for both traffic types. This is attributed to the fact that CBS utilizes bandwidth allocation distribution, which results in slower credit recovery and higher end-to-end delay. In future work, we plan to consider frame preemption with different shapers and provide further analysis for industrial and automotive networks. Overall, the results of our study provide valuable insights for network designers to choose the appropriate shaper for different traffic types and priorities and further encourages the research community to propose tighter analytical delay bounds for TSN shapers.

REFERENCES

[1] "Ieee standard for local and metropolitan area network–bridges and bridged networks," *IEEE Std 802.1Q-2018 (Revision of IEEE Std 802.1Q-2014)*, pp. 1–1993, 2018.

[2] "Ieee standard for local and metropolitan area networks – bridges and bridged networks - amendment 25: Enhancements for scheduled traffic," *IEEE Std 802.1Qbv-2015 (Amendment to IEEE Std 802.1Q-2014 as amended by IEEE Std 802.1Qca-2015, IEEE Std 802.1Qcd-2015, and IEEE Std 802.1Q-2014/Cor 1-2015)*, pp. 1–57, 2016.

[3] "Ieee standard for local and metropolitan area networks–audio video bridging (avb) systems," *IEEE Std 802.1BA-2011*, pp. 1–45, 2011.

[4] "Ieee standard for local and metropolitan area networks - virtual bridged local area networks amendment 12: Forwarding and queuing enhancements for time-sensitive streams," *IEEE Std 802.1Qav-2009 (Amendment to IEEE Std 802.1Q-2005)*, pp. C1–72, 2010.

[5] "Ieee standard for local and metropolitan area networks–bridges and bridged networks - amendment 34:asynchronous traffic shaping," *IEEE Std 802.1Qcr-2020 (Amendment to IEEE Std 802.1Q-2018 as amended by IEEE Std 802.1Qcp-2018, IEEE Std 802.1Qcc-2018, IEEE Std 802.1Qcy-2019, and IEEE Std 802.1Qcx-2020)*, pp. 1–151, 2020.

[6] "Ieee draft standard for local and metropolitan area networks - timing and synchronization for time-sensitive applications," *IEEE P802.1AS-Rev/D6.0 December 2017*, pp. 1–496, 2018.

[7] J. Specht and S. Samii, "Urgency-based scheduler for time-sensitive switched ethernet networks," in *2016 28th Euromicro Conference on Real-Time Systems (ECRTS)*, 2016, pp. 75–85.

[8] INET, "Inet framework," https://inet.omnetpp.org/, 2022, accessed: 2022-11-30.

[9] T. Steinbach, H. D. Kenfack, F. Korf, and T. C. Schmidt, "An extension of the omnet++ inet framework for simulating real-time ethernet with high accuracy," in *Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques*, ser. SIMUTools '11. Brussels, BEL: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2011, p. 375–382.

[10] J. Falk, D. Hellmanns, B. Carabelli, N. Nayak, F. Dürr, S. Kehrer, and K. Rothermel, "Nesting: Simulating ieee time-sensitive networking (tsn) in omnet++," in *2019 International Conference on Networked Systems (NetSys)*, 2019, pp. 1–8.

[11] L. Zhao, P. Pop, and S. Steinhorst, "Quantitative performance comparison of various traffic shapers in time-sensitive networking," *CoRR*, vol. abs/2103.13424, 2021.

[12] A. Nasrallah, A. S. Thyagaturu, Z. Alharbi, C. Wang, X. Shao, M. Reisslein, and H. Elbakoury, "Performance comparison of ieee 802.1 tsn time aware shaper (tas) and asynchronous traffic shaper (ats)," *IEEE Access*, vol. 7, pp. 44 165–44 181, 2019.

[13] Z. Zhou, Y. Yan, M. Berger, and S. Ruepp, "Analysis and modeling of asynchronous traffic shaping in time sensitive networks," in *2018 14th IEEE International Workshop on Factory Communication Systems (WFCS)*, 2018, pp. 1–4.

[14] Z. Zhou, J. Lee, M. S. Berger, S. Park, and Y. Yan, "Simulating tsn traffic scheduling and shaping for future automotive ethernet," *Journal of Communications and Networks*, vol. 23, no. 1, pp. 53–62, 2021.

[15] E. Mohammadpour, E. Stai, M. Mohiuddin, and J.-Y. Le Boudec, "Latency and backlog bounds in time-sensitive networking with credit based shapers and asynchronous traffic shaping," in *2018 30th International Teletraffic Congress (ITC 30)*, vol. 02, 2018, pp. 1–6.

[16] B. Fang, Q. Li, Z. Gong, and H. Xiong, "Simulative assessments of credit-based shaping and asynchronous traffic shaping in time-sensitive networking," *2020 12th International Conference on Advanced Infocomm Technology (ICAIT)*, pp. 111–118, 2020.

[17] F. Luo, B. Wang, Z. Yang, P. Zhang, Y. Ma, Z. Fang, M. Wu, and Z. Sun, "Design methodology of automotive time-sensitive network system based on omnet++ simulation system," *Sensors*, vol. 22, no. 12, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/12/4580

[18] "Ieee standard for local and metropolitan area networks – bridges and bridged networks – amendment 26: Frame preemption," *IEEE Std 802.1Qbu-2016 (Amendment to IEEE Std 802.1Q-2014)*, pp. 1–52, 2016.

[19] A. Arestova, K.-S. Jens Hielscher, and R. German, "Simulative evaluation of the tsn mechanisms time-aware shaper and frame preemption and their suitability for industrial use cases," in *2021 IFIP Networking Conference (IFIP Networking)*, 2021, pp. 1–6.

[20] M. Boyer and H. Daigmorte, "Impact on credit freeze before gate closing in cbs and gcl integration into tsn," in *Proceedings of the 27th International Conference on Real-Time Networks and Systems*, ser. RTNS '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 80–89.

[21] L. Zhao, P. Pop, Z. Zheng, H. Daigmorte, and M. Boyer, "Latency analysis of multiple classes of avb traffic in tsn with standard credit behavior using network calculus," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 10, pp. 10 291–10 302, 2021.

[22] V. Gavriluţ and P. Pop, "Traffic-type assignment for tsn-based mixed-criticality cyber-physical systems," *ACM Trans. Cyber-Phys. Syst.*, vol. 4, no. 2, jan 2020. [Online]. Available: https://doi.org/10.1145/3371708

[23] "P60802 - iec/ieee 60802 time-sensitive networking profile for industrial automation," 03 Mar 2023.