

A Study on the Impact of the Time-Aware Shaper on Best-Effort Traffic

Renan M. Silva, Aellison C. T. Santos, Iguatemi E. Fonseca

Abstract—The goal of Time-Sensitive Networking (TSN) is to provide high-performance deterministic communication and enable different traffic types (e.g., time-critical, Best-Effort, and audio/video streams) to coexist without interfering with the latency requirements of critical flows. To achieve this, techniques such as packet scheduling are applied. In this work, we show how the Best-Effort traffic is impacted when a TSN schedule is applied to the network, using latency and jitter as analysis parameters. To validate the impact of TSN mechanisms on Best-Effort traffic, a simulation-based approach with support for multiple traffic classes, including fault injection, time-critical and Best-Effort flows, is adopted.

Keywords—Time-Sensitive Networks, Time-Aware Shaper, Simulation.

I. INTRODUCTION

Modern industrial systems have several traffic types, such as time-critical, Best-Effort, and audio/video streams. Each of these traffic flows has different priorities and requirements, making network design a challenging task [1]. An increasingly diverse range of industrial domains requires deterministic latency and jitter in their communication, including automotive, aviation, and smart factories. Thus, new methods for network design and communication need to be applied [2].

Time-Sensitive Networking (TSN) has emerged as a set of IEEE standards designed to address these requirements by providing deterministic communication over Ethernet-based networks [3]. TSN introduces mechanisms such as time-aware shaping, traffic scheduling, and resource reservation to ensure bounded latency and low jitter for time-critical traffic. By enforcing strict transmission schedules, TSN enables the coexistence of heterogeneous traffic classes within the same network infrastructure.

However, while TSN mechanisms effectively guarantee performance for critical flows, their impact on Best-Effort traffic remains an important aspect to be analyzed. Best-Effort traffic does not have explicit timing guarantees and is typically served when no higher-priority traffic is scheduled for transmission. As a consequence, the application of TSN scheduling may lead to increased latency, higher jitter, and reduced throughput for Best-Effort flows, especially under high network load or dense scheduling configurations [4], [5].

Understanding the behavior of Best-Effort traffic in TSN-enabled networks is essential for evaluating overall system performance and ensuring that non-critical applications continue to operate within acceptable limits. Many industrial applications rely on Best-Effort communication for tasks such as diagnostics, monitoring, software

updates, and data logging, which, although not time-critical, are still fundamental to system operation [6].

To validate the impact of TSN mechanisms on Best-Effort traffic, a simulation-based approach is adopted. The simulation model represents a TSN-enabled Ethernet network that supports multiple traffic classes, including time-critical and Best-Effort flows. Time-aware scheduling mechanisms are implemented within the model to reproduce TSN behavior accurately, allowing controlled manipulation of scheduling parameters and traffic patterns. The simulation environment enables systematic evaluation under repeatable and configurable conditions. Different network scenarios are defined by varying scheduling window configurations, and the proportion of Best-Effort traffic relative to time-critical streams. This controlled setup makes it possible to isolate the effects of TSN scheduling on Best-Effort traffic and to observe performance variations that would be difficult to measure in real deployments.

Key Contributions

- We present a simulation-based study analyzing the impact of TSN scheduling mechanisms on Best-Effort traffic in mixed-criticality networks.
- We evaluate Best-Effort traffic performance in TSN-enabled scenarios using latency and jitter metrics under scheduling configurations.
- We provide insights derived from simulation results to support TSN network design, highlighting trade-offs between deterministic guarantees for time-critical flows and performance degradation of Best-Effort traffic.

Ultimately, this research provides insights to support TSN network design by identifying conditions under which Best-Effort traffic is significantly affected by different scheduling configurations and traffic loads.

The remainder of this paper is organized as follows. In Section II, we review the relevant literature, highlighting the tools used for TSN scheduling and simulation. In Section III, we perform the experimental evaluation, divided by simulation setup and results. Finally, Section IV goes over the results and conclusions of this work.

II. RELATED WORK

The purpose of TSN is to enable deterministic communication regarding latency and jitter. However, this objective can be separated into three main points: i) the support of scheduled traffic; ii) convergence of different types of traffic in the same network; and iii) zero-loss reliability [7]. In this section, we will highlight the state of the art regarding Time-Aware Shaper (TAS) schedule generation and TSN simulation.

A. TAS Schedule Generation

The Time-Aware Shaper (TAS) is the scheduling technique proposed by the TSN standard IEEE 802.1Qbv, responsible for scheduling traffic into eight different priorities [25]. Figure 1 illustrates how this mechanism works. The generation of a TAS schedule for the network switches is critical to achieve the deterministic communication proposed by TSN. However, this task is challenging since several factors should be considered, such as packet size, periodicity, number of devices, route, link capacity, latency, and jitter. Thus, the TAS scheduling is an NP-complete problem [8].

TSN packets have a VLAN identifier on the IEEE 802.1Q header [9]. This identifier, known as the Priority Code Point (PCP), defines the priority level of a packet at the time of its creation. Each port on a TSN switch has up to eight priority queues. When a packet arrives at a switch, it is processed and goes through a filter, often called the *Switching Fabric*. This Priority Filter will check the PCP of the incoming packet and direct it to the corresponding priority queue.

Each queue has a Forwarding Gate. This gate has two states: open (1) and closed (0). The packets of that queue can only be transmitted if the respective forwarding gate is open. The status of each forwarding gate is determined by the scheduling table known as the Gate Control List (GCL).

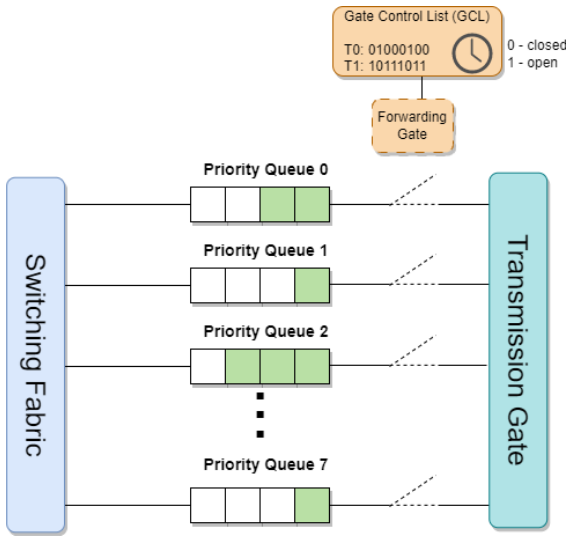


Fig. 1. Time-Aware Shaper [23].

The TAS schedule contains all information necessary to control the forwarding gates and priority queues of the network switches. However, this schedule is computed offline, at design time, and remains the same for the rest of the network execution, making it difficult to add or remove devices and flows from the network dynamically.

Dynamic applications, such as the Industrial Internet of Things (Industry 4.0), can benefit from the scheduled traffic proposed by TSN [10]. However, these scenarios require online flow scheduling, since new devices and flows can be added or removed constantly. This is called Plug-And-Produce configuration systems [11]. When this happens, the network schedule must be recalculated, and this may take time.

TSNsched [15] is an application that uses the Z3 SMT (Satisfiability Modulo Theories) Solver [14] to generate a TAS schedule

given an input file with the network topology. This input file must contain information about the devices and data flows of the network, such as: packet size, periodicity, latency and jitter requirements, etc.

A key advantage of using SMT-based approaches is the existence of mature SMT solvers, such as Z3. This means that workflows using such tools can take advantage of improvements made directly in the solvers. However, TSNsched is open-source, whereas the tools produced by TTTech [2], [8], [12], [13] are not.

B. Simulating TSN

Generating the schedule for a TSN network is only part of the challenge. There are multiple ways to model this problem, as discussed previously in this section. Tools used to create network schedules usually cannot guarantee the effectiveness of the schedule by themselves. A way to prove the effectiveness of the schedules generated by these tools is by simulation.

The use of a simulated environment gives the user more freedom to create and validate more complex scenarios with less effort, preventing potential damage to the equipment that would be used in experimental scenarios.

We can find a robust tool in NeSTiNg [21]. This tool is also an extension to the INET Framework [17], designed to support multiple TSN Standards, such as 802.1Qav, 802.1Qbv, and 802.1Qbu. The main difference in NeSTiNg is the method used to configure a scenario. NeSTiNg uses XML files to configure essential parts of the simulation, e.g., routing and port scheduling of the TSN Switches.

A survey [7] mentions the use of simulation frameworks as an evaluation method. From the 49 simulation studies investigated, 23 used OMNET++ [16]; 21 of these papers also used INET. Among the studies that used INET, 7 of them used NeSTiNg and only 3 used CoRE4INET [18]. These numbers show how NeSTiNg has been accepted and used as a reliable open source simulation framework for TSN.

Although TSN was designed to guarantee deterministic latency and jitter for critical applications, Best-Effort traffic also plays an important role in industrial networks, since many non-critical applications share the same communication infrastructure.

To analyze the impact on Best-Effort traffic, it is necessary to employ a simulation model capable of supporting multiple traffic types while adhering to TSN standards. For this purpose, we use the Nest-Sched simulation model [23]. This tool is the result of our ongoing research, and it extends the NeSTiNg model by incorporating additional features, including the capability to simulate faulty devices.

Studies such as [19] and [20] highlight how TSN mechanisms, such as the Time-Aware Shaper (TAS), can affect network operability and resource availability. The extensive use of reserved transmission windows and guard bands increases the determinism of critical flows but reduces the bandwidth available to non-critical traffic, potentially degrading its performance and even causing starvation. In contrast, our work extends this analysis by considering not only the degradation of network flows but also the presence of faulty devices. The simulation tool employed in our study and the results obtained take these aspects into account, enabling the evaluation of how device faults and flow degradation affect the performance and behavior of TSN networks.

III. EXPERIMENTAL EVALUATION

In this section, we describe the simulation setup, including the tools used and the network specifications. We also analyze the impact on Best-Effort traffic when TSN techniques are applied and when crude modifications to the schedule are introduced.

A. Simulation Setup

TSNsched was the first open-source tool able to solve the TAS schedule generation problem, having extensive academic documentation [22] and offering more granular configurability when compared to related works (as discussed in Section 2.1), which motivated the adoption in this study.

We use the TSNsched tool to generate the schedule for the network depicted in Figure 2. The network has a constant data transfer rate of 1 Gbit/s, 16 end-devices, 4 switches, and 16 traffic flows with different priorities, packet length, and periodicity. Out of the 16 end-devices, 10 are used for the critical (time-sensitive) flows and the other 6 are designated for non-critical (best-effort) flows. The details about the network can be found in Tables I and II.

In Table I, the Start and End columns represent the transmitter and receiver devices. The Priority column specifies the priority of that flow. Finally, the columns Size and Interval indicate the packet size and transmission periodicity, respectively. These values represent the static parameters used for TSN data flows and are necessary to obtain deterministic results.

The non-critical traffic described in Table II does not use static variables to define packet interval and size. To make the scenario more realistic, the simulation model we developed uses the value defined by the user as a seed for probability distributions, such as Poisson's Number Generator, and randomly assigns values to the non-scheduled traffic throughout the simulation execution, since Non-Critical (Best-Effort) traffic flows are unpredictable.

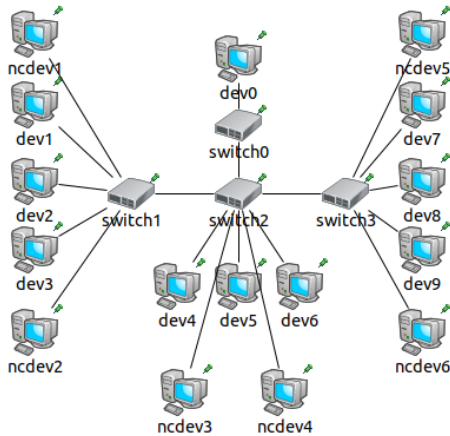


Fig. 2. Network topology used for simulations.

B. Results

The graph depicted in Figure 3 represents the latency of critical flows operating under ideal conditions. As can be observed, all flows remain stable and consistently meet the specified latency constraints defined by the schedule. This behavior demonstrates that the TSN

TABLE I

TIME-SENSITIVE TRAFFIC FLOW DETAILS.

	Start	End	Priority	Size	Interval
flow1	dev0	dev5	0	500 B	400 μ s
flow2	dev1	dev3	2	300 B	800 μ s
flow3	dev3	dev1	1	300 B	800 μ s
flow4	dev2	dev0	2	300 B	500 μ s
flow5	dev4	dev6	2	400 B	800 μ s
flow6	dev6	dev4	6	600 B	800 μ s
flow7	dev5	dev0	6	700 B	500 μ s
flow8	dev7	dev9	0	500 B	800 μ s
flow9	dev9	dev7	0	300 B	800 μ s
flow10	dev8	dev0	7	800 B	500 μ s

TABLE II

BEST-EFFORT TRAFFIC FLOW DETAILS.

	Start	End	Priority	Size*	Interval*
NC-flow1	NCdev1	dev0	5	700 B	19 μ s
NC-flow2	NCdev2	dev2	5	700 B	19 μ s
NC-flow3	NCdev3	dev0	5	700 B	19 μ s
NC-flow4	NCdev4	dev5	5	700 B	19 μ s
NC-flow5	NCdev5	dev0	5	700 B	19 μ s
NC-flow6	NCdev6	dev8	5	700 B	19 μ s

configuration is capable of guaranteeing deterministic communication for time-critical traffic when the network operates as expected.

Figure 4, on the other hand, illustrates the latency observed for non-critical flows. In contrast to the critical traffic, these flows experience noticeably higher latency values. The difference between the two classes of traffic is considerable, highlighting the prioritization mechanisms implemented in TSN networks. Since the scheduling mechanisms are primarily designed to guarantee strict timing requirements for critical flows, non-critical traffic is served only when transmission opportunities become available.

As shown in the network topology (Figure 2), multiple non-critical flows share common transmission paths and compete for forwarding resources, particularly at *switch2*, which acts as the central aggregation point to the network, connecting the sub-networks with the main device (*dev0*).

The oscillatory patterns and latency peaks observed in Figure 4 also indicate temporary congestion periods caused by the interaction between scheduled traffic and shared network resources. These effects become more pronounced for flows traversing heavily utilized paths, especially those converging toward *dev0*.

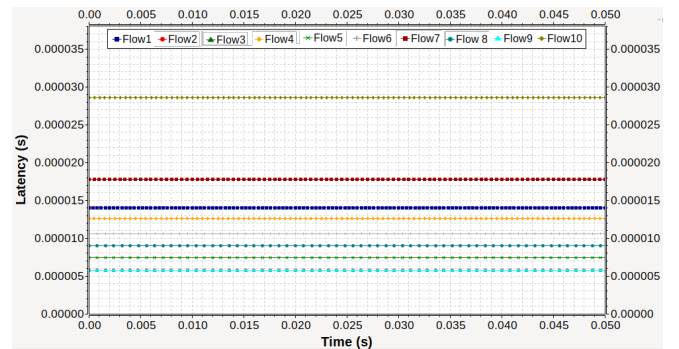


Fig. 3. Critical flows expected latency.

If we take a look at Table I, we can see that multiple flows also communicate with *dev0*. However, unlike the non-critical flows,

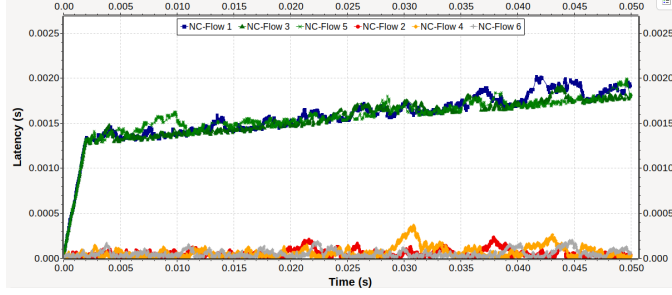


Fig. 4. Best-Effort traffic latency.

these communications are not affected, since the TAS schedule allocates dedicated transmission windows and priority levels to critical traffic. As a result, these flows have deterministic access to the network resources, allowing them to maintain low latency and jitter, while the non-critical flows are degraded.

This behavior becomes even more relevant when additional mechanisms are introduced into the network. TSN is continuously evolving, with new techniques being proposed to enhance determinism and reliability. One such mechanism is the Protection Band technique [24], which was introduced to enable the real-time forwarding of emergency event traffic in TSN networks by reserving a dedicated time window within the schedule.

While this reserved window improves the responsiveness of emergency traffic, it also affects the overall distribution of transmission opportunities in the network. In particular, the Protection Band directly impacts Best-Effort traffic, as it consumes time slots that were previously available for non-critical flows. As a result, the available bandwidth for Best-Effort traffic is further reduced, leading to increased latency and lower throughput.

The quantitative impact of this mechanism on non-critical traffic can be observed in the simulation results. Table III summarizes the measured performance metrics for the evaluated scenarios, illustrating how the introduction of additional TSN mechanisms influences the behavior of Best-Effort traffic within the network. The most significant degradation can be observed in NC-Flow4, whose latency increased from $71.984\mu s$ to $1275.572\mu s$, representing an increase of approximately 1672%. Its jitter also increased approximately 60.3%. This result indicates that the flow became substantially more affected by transmission blocking and queue accumulation after the introduction of Protection Bands.

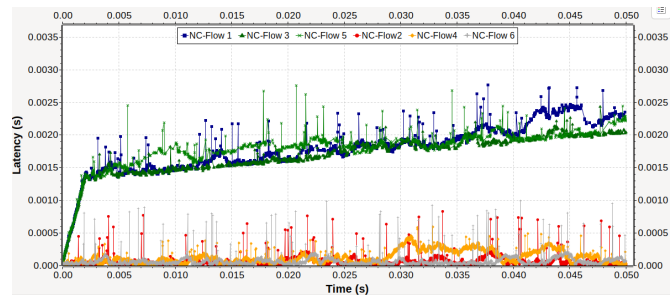


Fig. 5. Best-Effort traffic latency with Protection Band.

In TSN networks, all traffic transmissions must be precisely coordinated and accounted for to maintain deterministic communication. The scheduling mechanisms used in these

 TABLE III
BEST-EFFORT TRAFFIC RESULTS.

	Scenario			
	Without Protection Band		With Protection Band	
	Latency (μs)	Jitter (μs)	Latency (μs)	Jitter (μs)
NC-Flow1	1577.441	255	1777.86	368.057
NC-Flow2	56.190	41.5	61.654	57.094
NC-Flow3	1559.193	242.209	1706.384	297.685
NC-Flow4	71.984	60.349	1275.572	96.723
NC-Flow5	1535.527	265.304	1791.672	328.891
NC-Flow6	56.849	39.412	63.342	67.061

networks rely on carefully designed transmission windows that guarantee timing requirements for critical flows. Any deviation from the expected configuration may compromise these guarantees and affect the overall network performance.

Therefore, the introduction of a faulty device, such as a malfunctioning switch, or significant modifications to a carefully designed TAS schedule can disrupt the network [23]. When such disruptions occur, the deterministic behavior expected from the schedule may no longer be preserved. This can lead to unexpected delays, increased contention for transmission opportunities, and variations in packet delivery times.

Although TSN mechanisms are primarily designed to protect critical traffic, disturbances in the schedule can also propagate to other traffic classes. In particular, non-critical and Best-Effort flows may experience noticeable performance degradation as a consequence of these irregularities. Since these flows rely on the remaining available transmission opportunities, any disruption in the schedule may reduce the resources allocated to them, increasing latency and reducing throughput. The effects of this scenario can be observed in Figures 6 and 7, which illustrate the latency behavior of the evaluated traffic under the modified conditions. A more detailed quantitative analysis of the results is presented in Table IV, where the performance metrics for the different scenarios are summarized. The results demonstrate that faulty devices can significantly degrade the performance of non-critical communication.

The results also reveal that some flows are affected differently depending on their communication paths and contention levels. While NC-Flow2 and NC-Flow6 remained relatively stable, flows such as NC-Flow1, NC-Flow3, and NC-Flow5 experienced significant degradation due to the higher concentration of traffic and resource competition at shared switches. These observations reinforce the strong relationship between TSN scheduling configuration, network topology, and the performance of Best-Effort traffic.

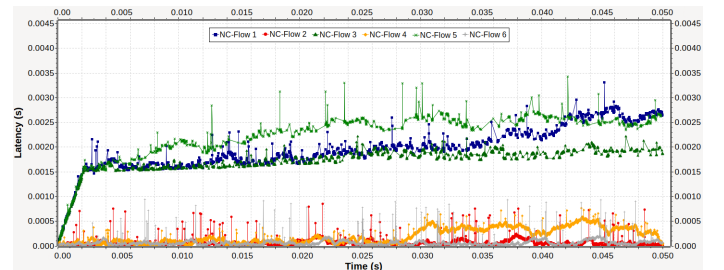


Fig. 6. 5% decrease of Best-Effort window with switch overload.

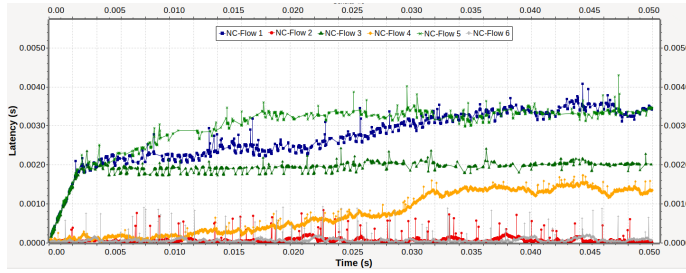


Fig. 7. 10% decrease of Best-Effort window with switch overload.

TABLE IV

COMPARISON OF BEST-EFFORT TRAFFIC WITH SWITCH OVERLOAD RESULTS.

	Scenario					
	Original Schedule With Switch Overload and Protection Band		5% Window Decrease With Switch Overload		10% Window Decrease With Switch Overload	
	Latency (μ s)	Jitter (μ s)	Latency (μ s)	Jitter (μ s)	Latency (μ s)	Jitter (μ s)
NC-Flow1	1777.86	368.057	1958.93	418.808	2697.294	621.43
NC-Flow2	61.654	57.094	62.168	59.288	62.302	57.816
NC-Flow3	1706.384	297.685	1739.87	259.291	1911.813	260.938
NC-Flow4	1275.572	96.723	200.682	151.271	756.666	52.243
NC-Flow5	1791.672	328.891	2261.856	457.536	3027.85	628.92
NC-Flow6	63.342	67.061	63.488	64.67	63.133	63.724

IV. CONCLUSION AND FINAL COMMENTS

In this work, we focused on analyzing the impact that the adoption of a TAS schedule in TSN networks has on traffic flows other than time-critical ones. By using a simulation-based approach to emulate the network and evaluate different scenarios, we observed not only substantial latency differences among traffic types, but also the performance degradation experienced by Best-Effort traffic when additional TSN mechanisms are introduced or when scheduling errors occur.

The results demonstrate that mechanisms such as Protection Bands and reserved transmission windows reduce transmission opportunities for Best-Effort traffic, increasing queue accumulation, latency, and jitter, particularly under switch overload or reduced scheduling windows. Flows sharing highly contended paths are more affected, highlighting the relationship between network topology, traffic distribution, and TSN scheduling policies.

Although deterministic communication is essential for time-critical applications, TSN networks must also support the coexistence of multiple traffic classes without excessive performance degradation. Therefore, ensuring reliable critical traffic while maintaining adequate performance for non-critical flows remains an important challenge in TSN networks.

As future work, we intend to further investigate the non-periodic characteristics associated with Best-Effort traffic, and how it can be used to improve TSN networks. Since non-critical traffic does not follow a strict periodic transmission pattern, adaptive scheduling and traffic-aware approaches may use resources more efficiently while reducing latency and jitter degradation. Recent studies, such as [5], have shown that considering the variability and non-periodic behavior of Best-Effort traffic allows TSN networks to achieve a better balance between critical flows deterministic guarantee and overall network efficiency.

REFERENCES

[1] C. Simon, M. Maliosz, and M. Mate, "Design Aspects of Low-Latency Services with Time-Sensitive Networking," *IEEE Communications Standards Magazine*, vol. 2, no. 2, pp. 48–54, 2018.

[2] W. Steiner, S. S. Craciunas, and R. Serna Oliver, "Traffic Planning for Time-Sensitive Communication," *IEEE Communications Standards Magazine*, vol. 2, no. 2, pp. 42–47, 2018.

[3] IEEE 802.1, "Time-Sensitive Networking Task Group." Available: <http://www.ieee802.org/1/pages/tsn.html> (Accessed: Feb. 27, 2026).

[4] D. Wei et al., "DC-GCL: Dynamically Configurable Gate Control List in Automotive TSN Switch for ADAS," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 45, no. 3, pp. 1225–1238, March 2026.

[5] N. Diaz and N. Gautam, "Non-Periodic Traffic Allocation in Time-Aware TSN Networks," *IEEE Access*, vol. 13, pp. 187758–187773, 2025.

[6] L. Zhao, D. Pop, Z. Zheng, H. Daigmore, 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. 10291–10302, Oct. 2021.

[7] Y. Seol, D. Hyeon, J. Min, M. Kim, and J. Paek, "Timely Survey of Time-Sensitive Networking: Past and Future Directions," *IEEE Access*, vol. 9, pp. 142506–142527, 2021.

[8] S. Craciunas, R. Serna Oliver, M. Chmelík, and W. Steiner, "Scheduling Real-Time Communication in IEEE 802.1Qbv Time Sensitive Networks," in *Proc. ACM International Conference*, 2016.

[9] IEEE, *IEEE Standard for Local and Metropolitan Area Networks—Bridges and Bridged Networks*, IEEE Std 802.1Q-2018, 2018.

[10] E. Baccarelli, P. G. V. Naranjo, M. Scarpiniti, M. Shojafar, and J. H. Abawajy, "Fog of Everything: Energy-Efficient Networked Computing Architectures, Research Challenges, and a Case Study," *IEEE Access*, vol. 5, pp. 9882–9910, 2017.

[11] L. Monostori, B. Kádár, T. Bauernhansl, S. Kondoh, S. Kumara, G. Reinhart, O. Sauer, G. Schuh, W. Sihn, and K. Ueda, "Cyber-physical systems in manufacturing," *CIRP Annals*, vol. 65, no. 2, pp. 621–641, 2016.

[12] W. Steiner, "An Evaluation of SMT-Based Schedule Synthesis for Time-Triggered Multi-Hop Networks," in *2010 31st IEEE Real-Time Systems Symposium (RTSS)*, San Diego, CA, USA, 2010, pp. 375–384.

[13] S. Brooks and E. Uludag, "Time-Sensitive Networking: From Theory to Implementation in Industrial Automation," TTTech and Intel, White Paper, 2015.

[14] Z3Prover Project, "Z3 Solver." Available: <https://github.com/Z3Prover/z3>

[15] A. C. T. dos Santos, B. Schneider, and V. Nigam, "TSNSCHED: Automated Schedule Generation for Time-Sensitive Networking," in *2019 Formal Methods in Computer Aided Design (FMCAD)*, San Jose, CA, USA, 2019, pp. 69–77.

[16] OMNeT++ Community, "OMNeT++ Discrete Event Simulator." Available: <https://omnetpp.org/> (Accessed: Feb. 27, 2026).

[17] INET Framework Team, "INET Framework for OMNeT++." Available: <https://inet.omnetpp.org/> (Accessed: Feb. 27, 2026).

[18] Communication over Real-Time Ethernet Group, "CoRE4INET Framework." Available: <https://core-researchgroup.de/projects/simulation.html> (Accessed: Feb. 27, 2026).

[19] B. Houtan, M. Ashjaei, M. Daneshmand, M. Sjödin and S. Mubeen, "Work in Progress: Investigating the Effects of High Priority Traffic on the Best Effort Traffic in TSN Networks," 2019 IEEE Real-Time Systems Symposium (RTSS), Hong Kong, China, 2019, pp. 556–559.

[20] B. Houtan, M. Ashjaei, M. Daneshmand, M. Sjödin, S. Afshar and S. Mubeen, "Schedulability Analysis of Best-Effort Traffic in TSN Networks," 2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vasteras, Sweden, 2021, pp. 1–8.

[21] J. Falk, D. Hellmanns, B. Carabelli, N. Nayak, F. Dürr, S. Kehrler, and K. Rothenmel, "NeSTiNg: Simulating IEEE Time-sensitive Networking (TSN) in OMNeT++," in *2019 International Conference on Networked Systems (NetSys)*, 2019, pp. 1–8.

[22] A. C. T. Santos, R. M. Silva, B. Schneider, M. Wilhelm, I. E. Fonseca, and V. Nigam, "A novel open-source framework for performing TSN schedules," *Simulation Modelling Practice and Theory*, vol. 144, p. 103147, 2025.

[23] R. M. Silva, A. C. T. Santos, I. E. Fonseca, and V. Nigam, "Fault Injection and Reliability Analysis on Time-Sensitive Networking," *IEEE Internet of Things Journal*, vol. 12, no. 2, pp. 1153–1164, 2025.

[24] M. Kim, J. Min, D. Hyeon, and J. Paek, "TAS Scheduling for Real-Time Forwarding of Emergency Event Traffic in TSN," in *2020 International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju, South Korea, 2020, pp. 1111–1113.

[25] "IEEE Standard for Local and Metropolitan Area Network—Bridges and Bridged Networks," in IEEE Std 802.1Q-2018 (Revision of IEEE Std 802.1Q-2014), vol., no., pp.1-1993, 6 July 2018,