Towards TSN-5G integration: simulating time synchronization through 5G via OMNeT++

Sergio Rossi B. da Silva[1], **Francisco Germano Vogt**[1], **Christian Esteve Rothenberg**[1]

[1]Faculdade de Engenharia Elétrica e Computação
Universidade Estadual de Campinas (UNICAMP) - SP - Brasil

s217042@dac.unicamp.br, f234632@dac.unicamp.br, chesteve@unicamp.br

Abstract. *Integrating Time-Sensitive Networking (TSN) and wireless networks, like the Fifth Generation of mobile networks (5G), affords deterministic communication with great flexibility and mobility, enabling new services to be used in smart industries. This work aims to provide advances in simulating this integration through OMNeT++ simulator and developing extended modules to support it, such as the translator devices. We focus on time synchronization between the TSN and 5G based on 3GPP Release 17, which no simulator has fully covered yet, demonstrate the synchronization in every supported direction using the 5G as a time-aware system and presenting the accuracy achieved by our implementation.*

1. Introduction

Factories and society are transforming digitally, resulting in several time-critical services such as Extended Reality (XR), collaborating robots, and autonomous driving [Satka et al. 2023a]. These services' requirements are distinct from traditional networks, requiring deterministic communication that delivers low End-to-End (E2E) latency and high reliability [Rao et al. 2022].

Time-Sensitive Networking (TSN) is an Ethernet-based set of standards developed to achieve deterministic communication requirements. With time synchronization, flow and resource management, reliability, and security, it offers a comprehensive solution for robust, hard real-time communications [Deng et al. 2022]. However, while TSN provides reliable communication, the technology lacks flexibility and mobility in its applications due to cable dependency, being more suitable for local applications.

Several studies have been conducted on integrating TSN with wireless networks to address this issue, such as the Fifth Generation of Mobile Networks (5G) developed by the 3rd Generation Partnership Project (3GPP). The 5G, different from the previous generations, brings the concept of dividing the network into different service classes, providing better resource allocation for each service [Bhattacharjee et al. 2021]. Ultra-Reliable Low Latency Communication (URLLC) is one of those services, also planned to be present in the next generation (6G) and, like TSN, brings different features to achieve real-time communication [Haque et al. 2023].

Integrating TSN and 5G is also part of 3GPP specifications, bringing new elements to the 5G system (5GS) to apply translator functionalities between these networks. However, due to the difference between both technologies, integration is not an easy task, and the research in this area seeks to address the core challenges, such as time synchronization, security, and several architectural properties [Satka et al. 2023b].

Besides that, [Satka et al. 2023a] explains that another challenge is in simulating the integration, which is essential in understanding and promoting more advances. In [Muslim et al. 2024], they used the OMNeT++ simulator with INET and Simu5G frameworks to integrate both technologies and evaluate time synchronization. However, as pointed out in [DETERMINISTIC6G 2023], the correction field of gPTP from the INET was not implemented yet, leading to a decrease in the synchronization accuracy when it passes through a bridge node, the 5GS in the case of the integration. Besides, other issues in the gPTP, regarding the frequency ratio between the devices' clocks, for example, may not be considered in [Muslim et al. 2024].

This work uses the same tools to integrate TSN and 5G, concentrating predominantly on User Plane (UP) functionalities. We evaluate time synchronization with gPTP according to Release 17, extending the synchronization for Uplink (UL) direction and between User Equipments (UE), which, to our knowledge, has not yet been addressed in a simulation environment. We implemented our modifications to the gPTP module related to the correction field and also to pass through the 5G system (5GS).

PAPER OUTLINE: The rest of this paper is divided as follows: Section 2 presents the motivation and goal of this work, Section 3 shows some works related to time synchronization between TSN and 5G, Section 4 introduces some aspects of TSN-5G, and Section 5 finalizes by demonstrating the potential of our synchronization through results from some experiments and about our demonstration in SBRC.

2. Motivation and Goal

To improve efficiency and even enable some applications, deterministic communications are required. TSN is a suitable option, being compatible with other solutions already implemented in the industry. Several TSN standards, however, are based in a network with the device's clocks well synchronized, so when integrating with other technologies, the time synchronization between them is crucial. Our goal in this work is to create a simulated environment capable of time synchronization between TSN and 5G, which can further contribute to the integration development of these technologies.

3. Related work

This section presents relevant works related to the integration between TSN and 5G.

In [Nikhileswar et al. 2022], they made a demonstration of TSN-5G integration capabilities where they implemented a control system for a Ball-Balancing Table using devices that support TSN features, and, for the 5GS, a Release 15 modem, emulated RAN, and Core Network. Their experiments showed how TSN standards can minimize latency and provide high reliability, delivering the control commands more appropriately.

The systematic review in [Satka et al. 2023a] evaluated other works related to this integration. They observed the tendency to take time synchronization into account since its critical impact in deterministic systems, being the generic Precise Timing Protocol (gPTP), or the PTP itself, the majority approach used to synchronize the device's clock. They also observed that the optimization parameters more often analyzed in those works were latency, jitter, and resource efficiency, which were justified by the requirements of the target applications.

In [Magnusson and Pantzar 2021] and [Satka et al. 2022] works, they simulated the integration of these networks for scenarios inspired by the vehicle industry, using OMNeT++ with INET and Nesting frameworks. They represented the 5GS as a connection with different parameters according to the Ethernet flow and worked with the traffic shaping between the domains. Currently, the INET has added the TSN functionalities, such as time synchronization, filtering, and policing, while Nesting, which also contains these functionalities, has stopped being updated.

In recent reports from the Deterministic6G project [DETERMINISTIC6G 2023], they work on the integration using the OMNeT++ with INET only, using 5G data collected by them for the simulation and validating some scenarios, one of them being a time synchronization scenario. However, as they explain in the report, they didn't present their results for time synchronization since it would not represent a real scenario, where one of the reasons is a problem in the INET in setting the correction field correctly, which is currently under investigation by the developers.

4. TSN-5G Integration: Reference Architecture

The 3GPP proposed architecture to support TSN and 5G integration has been developed since Release 16 and is illustrated in Figure 1. In the proposed framework, the 5G is seen as a logical bridge inside the TSN, containing elements to make translator functionalities between the systems in both UP and Control Plane (CP), called TSN translators (TT).

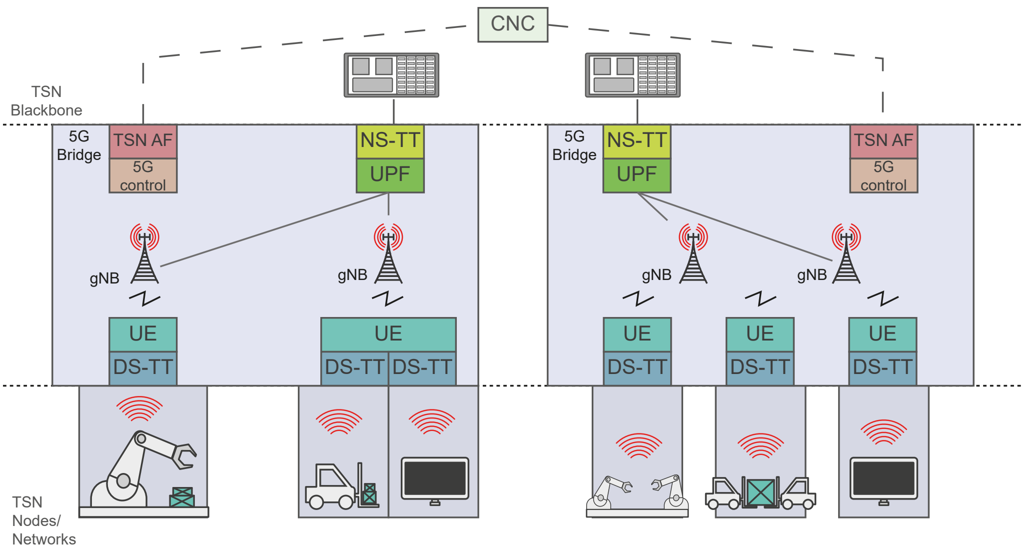


Figura 1. TSN-5G Architecture example

The CP translator, TSN Application Function (TSN AF), manages interactions with the TSN Centralized User Configuration (CNC), monitors and calculates new network configurations, and deploys them to the TSN Bridges through the UP translators. On the UP side, the translators are present on both edges of the 5G bridge, being divided into the Device-Side TSN translator (DS-TT), attached to UEs, and the Network-Side TSN Translator (NS-TT), attached to the UPF. They serve as the ingress and egress ports of the 5GS, maintaining the transparency of 5GS-specific procedures hidden from the TSN system. Optionally, they are responsible for hold-and-forward functionality for de-jittering, per-stream filtering and policing, link layer connectivity discovery, and reporting (optionally only for the DS-TT). Using those translators, the 5G bridge can separate one or more applications connected to the UE from the Programmable Logical controllers (PLC) attached to the UPF.

Regarding time synchronization, the 3GPP has proposed a framework using 5G as a PTP instance, being the node with the Grand Master (GM) clock node or not, and applying the PTP or gPTP. In Release 17, when the 5GS doesn't have the GM clock, it can be located anywhere, and the synchronization can be done by UL, DL, and between UEs.

The gPTP is the standard used for time synchronization in TSN, where the devices to be synchronized are inside a gPTP domain composed of three types of nodes: a master node, bridge nodes, and slave nodes. The master node contains the GM clock, acting as the reference clock for synchronizing the other nodes, while the slave nodes are just synchronized, and the bridge nodes are made both.

According to [Puttnies et al. 2018], in a gPTP domain, the nodes can present master, slave, passive, and disabled ports, where only the first two participate in the synchronization. Except for slave nodes, the others can have one or more master ports responsible for sending synchronization messages to a slave port and are present in all nodes except for the master.

Figure 2 symbolizes the gPTP synchronization process, from knowing the essential parameters to the sync messages. An important parameter in the gPTP process is the peer delay between the nodes, calculated after the first three messages exchanged between two nodes and updated periodically. Each slave port sends a request message to the master connected, saving the time the packet leaves (t1); when the master port receives the request, it sends a response with the time it arrived (t2); the time the response leaves (t3) is saved and sent in a follow-up message; the slave port receives the response for the request made and saves the time (t4), when the follow-up message arrives, the slave can calculate the peer delay of this connection by the following equation, where r is the ratio between the slave and master clock frequency, also updating with the sync process.

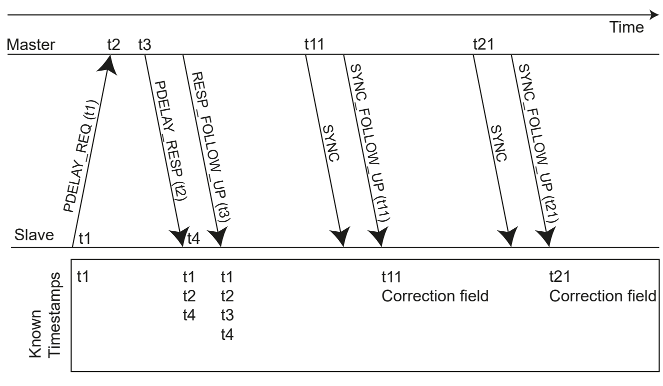


Figura 2. gPTP synchronization proccess

$$peerDelay = \frac{r * (t4 - t1) - (t3 - t2)}{2} \quad (1)$$

With the peer delay parameter known, the master frequently sends sync and follow-up messages to synchronize the slave containing the master's time, the correction field, and the frequency ratio between the device clock and the GM. The correction field contains information updated by each time-aware system between the master and the slave port. The slave synchronizes its clock by summing the master's time, the last calculated peer delay, the correction field received, and the time between the current time and the time the sync message has arrived.

To extend the time synchronization in the TSN networks, the 5GS calculates the residence time that the gPTP messages stay by considering the Timestamp ingress (TSi) and Timestamp egress (TSe) through the TTs and adding to the correction field in the gPTP.

Time synchronization in the TSN-5G integration is still an active and evolving research area since the integration is not feasible until it fulfills the accuracy requirements of the IEEE 802.1AS standard. Being a Layer 2 protocol, as in the 5GS it works oriented by IP, it adds more delay sources in real networks. Moreover, it is also possible that something happens and the master node fails; in this case, the BMCA chooses another GM clock. However, this approach can be costly by the time the algorithm works to cover that more than one gPTP domain can be used simultaneously for redundancy, as explained in [Ali et al. 2021].

5. Time synchronization cases for TSN-5G integration

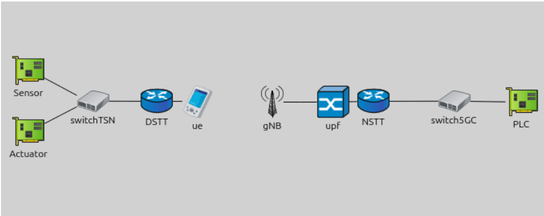
We now focus on the time synchronization scenarios and feature our OMNeT++ based simulation setup. We provide the topologies and preliminary results used in our experiments and describe what will be presented at the SBRC.

Time synchronization between TSN and 5G networks is implemented using the IEEE 802.1AS standard and modified to pass through a 5GS using UDP/IP encapsulation. The setup is based on a generic factory scenario where sensors capture and send data to an industrial controller, which, in turn, sends command data to actuators. One integration case can be seen in a Controller-to-Device communication with centralized control. The control is made with virtualized controllers located in an edge cloud as explained in [ACIA 2021].

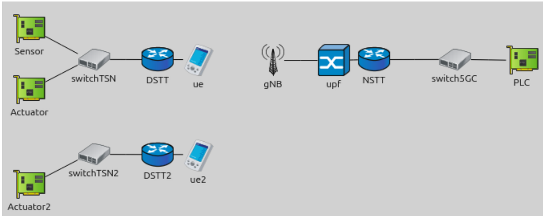
5.1. Environment

Our simulation uses OMNeT++, a modular C++ simulator environment for network and complex system modeling, with two of its frameworks. The INET framework provides essential features for wired and wireless modules and TSN basic features. Meanwhile, Simu5G is tailored explicitly for simulating 5G network scenarios.

Figure 3 represents the topologies used in our experiments. The sensor, actuators, and PLC devices are described using the *TsnDevice* module, while the switches, the *switchTSNs* and *switch5GC* nodes in the figure, are represented using the *TsnSwitch* model, both from the INET framework. For the 5GS, the Simu5G implements the base functionalities of UE, gNB, and UPF nodes, using no-connected modules to complement its functionalities. We modified those nodes to be capable of dealing with Ethernet protocols, increasing their compatibility with TSN standards. Furthermore, we also implement modifications in Simu5G to UPF and gNB forward packets destined to devices connected to the UEs.



(a) Environment with one UE



(b) Environment with two UE

Figura 3. Simulated environments used

The translators were implemented by extending the Router INET module to accept TSN standards and translator functions. Moreover, we also implemented a version of the gPTP module to travel through the 5GS using UDP/IP encapsulation, which the translators use to calculate the residence time inside the 5GS. We applied some modifications in Simu5G to subscribe to the existence of gPTP domains, which the NS-TT will analyze.

5.2. Experiments

To evaluate our time synchronization, we used parameters based on those used in [Muslim et al. 2024], with constant oscillator drift and considering perfect clock synchronization between the translators. In each experiment, the clock drift of the GM clock is set in $10ppm$ while the drift in other nodes is given by the OMNeT++ function *uniform* varying from $10ppm$ to $100ppm$. Regarding 5G, we used a subcarrier spacing (SCS) $30kHz$ and a Time Division Duplex (TDD) communication. The parameters for the synchronization process are the default established in the gPTP of the INET model. Sync messages are sent by the GM clock every $125ms$ according to its clock, and the peer delay is calculated every $1s$.

The first topology was used in the DL and UL synchronization experiments, while the second was used for synchronization between UEs. For the DL synchronization experiment, the GM was set to be the *switch5GC*; in contrast, the GM in the other experiments was the *switchTSN* node. In the synchronization experiment between UEs, the *switch5GC* and PLC do not participate.

We ran all our experiments during 200s in the simulated environment; however, using the gPTP, from the second synchronization onwards, the device's clocks compensate their drifts to be closer to the master's clock, which is mandatory for the protocol. Therefore, Figure 4 presents the difference between the device's clock and the GM clock immediately before and after the synchronization in the first $5s$ when the clock compensation was already done. The recalculation of the peer delay can be seen in the first synchronization after 1s. Moreover, as the curves were initially similar in all experiments, we present the results for the DL synchronization, but the results also represent the others.

As the synchronization results of the devices attached to the GM are the same in testing the gPTP module of the INET alone, in Figure 5, we present a boxplot of the time difference of the device whose synchronization passes through the 5G. To better comprehend, we show the results from the rest of the synchronization, taking off the initial data where the clock drifts were the same as initially configured. The results show that our synchronization reaches an accuracy of hundreds of nanoseconds, which is even better for UE-to-UE synchronization since the 5G bridge link is more symmetric.

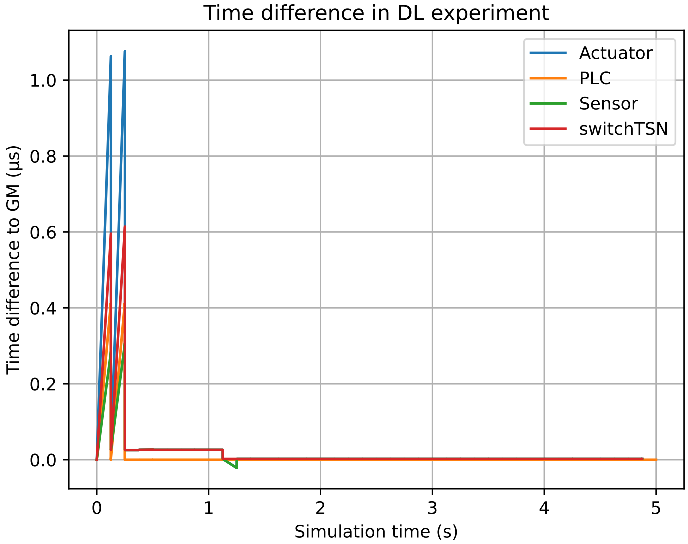
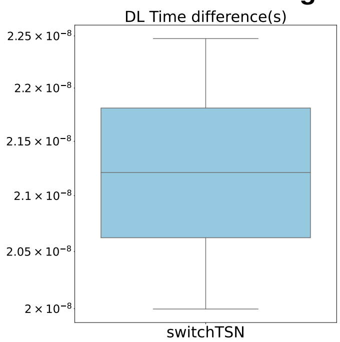
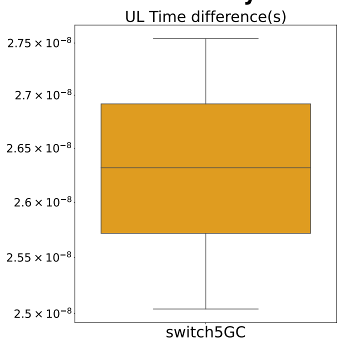


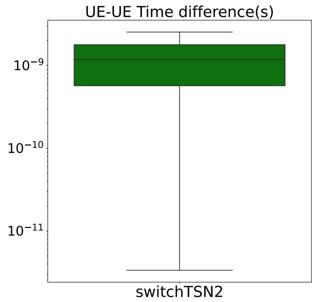
Figura 4. First 5s of time synchronization



(a) DL synchronization



(b) UL synchronization



(c) UE to UE synchronization

Figura 5. Boxplot of the time difference

5.3. Documentation, Code, and Demonstration

The implementation code of this work is available at `https://github.com/SergioRBDS/TSN5G-TimeSynchronization`, which also contains the documentation that provides a complete usage description of the main parameters (i.e., sync interval, clock drift, 5G numerology), where the others are the standard for the models. Additionally, a video tutorial demonstrating the configuration and execution of the tool is available in the drive folder `https://drive.google.com/drive/folders/19i3TFdgFgAJK6RsIkUxCERzfJSq4TeBk?usp=drive\_link`. The project welcomes contributions, bug reports, discussions of current functionalities, and proposals for new features. For the live demonstration at SBRC, we will show the synchronization process in every direction, as described in the experiments section, presenting how the simulation parameters impact the results.

Acknowledgment

This work was supported by the Innovation Center, Ericsson S.A., and by the São Paulo Research Foundation (FAPESP), grant 2021/00199-8, CPE SMARTNESS. This study was partially funded by CAPES, Brazil - Finance Code 001.

Referências

ACIA, G. (2021). Integration of 5g with time-sensitive networking for industrial communications.

Ali, R., Zikria, Y. B., Bashir, A. K., Garg, S., and Kim, H. S. (2021). Urllc for 5g and beyond: Requirements, enabling incumbent technologies and network intelligence. *IEEE Access*, 9:67064–67095.

Bhattacharjee, S., Katsalis, K., Arouk, O., Schmidt, R., Wang, T., An, X., Bauschert, T., and Nikaein, N. (2021). Network slicing for tsn-based transport networks. *IEEE Access*, 9:62788–62809.

Deng, L., Xie, G., Liu, H., Han, Y., Li, R., and Li, K. (2022). A survey of real-time ethernet modeling and design methodologies: From avb to tsn. *ACM Computing Surveys (CSUR)*, 55(2):1–36.

DETERMINISTIC6G (2023). Deterministic6g deliverable 4.1, "deterministic6g detcom simulator framework release 1". Online. Available: `https://deterministic6g.eu/index.php/library-m/deliverables` [Acces: April 05, 2024].

Haque, M. E., Tariq, F., Khandaker, M. R., Wong, K.-K., and Zhang, Y. (2023). A survey of scheduling in 5g urllc and outlook for emerging 6g systems. *IEEE access*.

Magnusson, A. and Pantzar, D. (2021). Integrating 5g components into a tsn discrete event simulation framework.

Muslim, A. B., Tönjes, R., and Bauschert, T. (2024). Synchronizing tsn devices via 802.1 as over 5g networks. *Electronics*, 13(4):768.

Nikhileswar, K., Prabhu, K., Cavalcanti, D., and Regev, A. (2022). Time-sensitive networking over 5g for industrial control systems. In *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–8. IEEE.

Puttnies, H., Danielis, P., Janchivnyambuu, E., and Timmermann, D. (2018). A simulation model of ieee 802.1 as gptp for clock synchronization in omnet++. In *OMNeT++*, pages 63–72.

Rao, A. S., Radanovic, M., Liu, Y., Hu, S., Fang, Y., Khoshelham, K., Palaniswami, M., and Ngo, T. (2022). Real-time monitoring of construction sites: Sensors, methods, and applications. *Automation in Construction*, 136:104099.

Satka, Z., Ashjaei, M., Fotouhi, H., Daneshtalab, M., Sjödin, M., and Mubeen, S. (2023a). A comprehensive systematic review of integration of time sensitive networking and 5g communication. *Journal of Systems Architecture*, 138:102852.

Satka, Z., Barhia, D., Saud, S., Mubeen, S., and Ashjaei, M. (2023b). Experimental analysis of wireless tsn networks for real-time applications. In *2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–4. IEEE.

Satka, Z., Pantzar, D., Magnusson, A., Ashjaei, M., Fotouhi, H., Sjödin, M., Daneshtalab, M., and Mubeen, S. (2022). Developing a translation technique for converged tsn-5g communication. In *2022 IEEE 18th International Conference on Factory Communication Systems (WFCS)*, pages 1–8. IEEE.