Wireless Time-Sensitive Networking for Real-Time Unmanned Ground Vehicle Control and Mapping

Elena Ferrari
Department of Information Engineering
University of Padova
Padova, Italy
elena.ferrari.7@phd.unipd.it

Dave Cavalcanti
Intel Corporation
Hillsboro, Oregon, USA
dave.cavalcanti@intel.com

Valerio Frascolla
Intel Deutschland GmbH
Neubiberg, Germany
valerio.frascolla@intel.com

Alberto Morato
National Research Council of Italy
CNR-IEIIT
Padova, Italy
alberto.morato@cnr.it

Stefano Vitturi
National Research Council of Italy
CNR-IEIIT
Padova, Italy
stefano.vitturi@cnr.it

Angelo Cenedese
Department of Information Engineering &
Department of Industrial Engineering
University of Padova
Padova, Italy
angelo.cenedese@unipd.it

***Abstract*—In light of the evolving landscape of future converged networks, heightened demands for scalability and performance have become paramount, particularly in the industrial context where enhanced flexibility, adaptability, time synchronization and deterministic low latency are key features. Time-Sensitive Networking (TSN) has been introduced to provide converged networks with determinism. To mitigate the complexities associated with cabling requirements, Wireless Time-Sensitive Networking (WTSN) emerges as a more suited solution in industrial applications that need mobility. This study presents an application in smart manufacturing that incorporates WTSN technology. Specifically, it focuses on a scenario where an Unmanned Ground Vehicle (UGV), under the control of an edge controller connected through a wireless link, operates within a critical environment subject to constraints such as to avoid obstacles and to build a real-time map of the surrounding area. The mapping outcome performed by the edge controller demonstrates the effectiveness of the WTSN capabilities.**

***Index Terms*—Wireless Time-Sensitive Networking, deterministic communication, Unmanned Ground Vehicle, IEEE 802.1AS, IEEE 802.1Qbv**

I. Introduction

The introduction of the Internet of Things (IoT) poses great opportunities for innovation across diverse industrial sectors and enterprises [1]. Historically, Information Technology and Operations Technology have been treated as distinct domains [2], whereas the contemporary industrial landscape requires their integration [3] to address the evolving requirements of industrial applications, facilitate remote access, and enable the interconnection of machines in the cloud, so that the resulting unified and converged network can concurrently transmit critical and non-critical data.

This study was partially carried out within the PNRR research activities of the consortium iNEST (Interconnected North-Est Innovation Ecosystem) funded by the EU Next-GenerationEU (Piano Nazionale di Ripresa e Resilienza (PNRR) – Missione 4 Componente 2, Investimento 1.5 – D.D. 1058 23/06/2022, ECS_00000043) and partially funded by the EC Horizon Europe SNS JU PREDICT-6G (GA 101095890) and MULTIX (GA 101192521) projects.

Industrial IoT (IIoT) can be seen as a synergy between the technology enablers above, aiming to decrease the gap between field and control levels [4], fostering flexible and intelligent production, and unlocking the full potential of industrial systems [5] in fields such as control systems, robotics, automation, manufacturing, production and logistics [3], [6].

These future converged industrial networks need scalability and greater flexibility, adaptability, time synchronization, and deterministic low-latency communications [7], [8] to ensure predictability, cost reduction, reliability, and operational efficiency improvement.

To address these challenges and create more integrated and standardized networks with precise time synchronization and timeliness in a network shared by time-critical and other types of traffic, Time-Sensitive Networking (TSN) and its wireless counterpart, i.e., Wireless TSN (WTSN), have been proposed [7], [9]. In detail, the latest advancements of Wi-Fi [10] and 5G networks enhanced with TSN capabilities currently stand out as the best candidates to meet the needs of time-critical applications, particularly in the field of robotics [5], [11] where the cable complexity is to be reduced and mobility introduced [12].

This paper investigates how to improve the predictability of time-sensitive (TS) processes and ensure deterministic latency over Wi-Fi by optimizing the TSN standards, such as IEEE 802.1AS for time synchronization and IEEE 802.1Qbv for traffic scheduling. It also evaluates an application of WTSN in the context of IIoT and smart manufacturing to demonstrate the benefits of reconfigurability, scalability and portability while meeting strict performance requirements in terms of latency over wireless links.

The considered scenario includes a mobile robot application, referred as a *LoCoBot*, controlled by an edge controller operating over WTSN in a time-critical fashion but guaranteeing deterministic communication between the controller and the LoCoBot itself. Opting for a remote edge controller is driven by its ability to create a scalable and adaptable system suited for factory and warehouse settings, going beyond the limits of onboard embedded hardware and allowing to achieve higher efficiency, as evidenced by [11] and demonstrated in [13].

The LoCoBot scans the environment thanks to a 2D LiDAR and sends the data to the edge controller via User Datagram Protocol (UDP) over a Wi-Fi network. The edge controller processes the data, sending suitable linear and angular velocities back to the LoCoBot to avoid obstacles and optimize mapping procedure. WTSN enables precise timing and low latency over Wi-Fi, critical for the high-speed maneuvers of the LoCoBot.

This paper evaluates how WTSN traffic scheduling affects UGV performance in building environment maps despite the Best Effort (BE) traffic. The need for deterministic communication arises from the LoCoBot requirements of fast operations, precise control, and carefully management of packet loss or delays. By scheduling TS and BE traffic, this study demonstrates the efficacy of WTSN in wireless communication for robotized intelligent production environments.

The paper is organized as follows. Section 2 describes the application scenario used, including hardware, network architecture, controller specifications, and the implementation of IEEE 802.1Qbv. Section 3 presents the experimental results focusing on the generated environment map and network performance metrics, i.e., latency and jitter, and finally conclusions are presented in Section 4.

II. Experimental Setup and Methodologies

This section delves into the practical setup employed to assess a LoCoBot within the WTSN framework.

A. Hardware devices and network configuration

The robotic component comprises a mobile robot, specifically a LoCoBot, equipped with a computational unit consisting of a NUC (NUC2) mounted on it and outfitted with a RPLIDAR, a 2-dimensional 360° Laser Scanner operating at a frequency of 10 Hz, as shown in Fig. 1. The RPLIDAR captures values within a range of $-40°$ to $40°$, with $0°$ aligning with the heading direction of the LoCoBot.

Fig. 1: LoCoBot structure with highlighted sensing module (LiDAR) and computational unit (NUC).

The network configuration, illustrated in Fig. 2, comprises two Wi-Fi stations (STAs). NUC1 serves as both the edge controller and a Software Access Point (SoftAP), establishing a point-to-point wireless connection with the second STA, NUC2, in the LoCoBot. This wireless connection between the NUCs forms the foundation of the WTSN domain. NUC1, as the Grand Master (GM), provides clock reference for synchronization to NUC2, acting as a Slave.

Both NUCs are equipped with the 11th generation Intel processors (i5-1135G7) and run on the Ubuntu 20.04 platform. Furthermore, NUC2 also runs the Robot Operating System (ROS), which facilitates the acquisition of LiDAR range data and controls the locomotion of the LoCoBot platform.

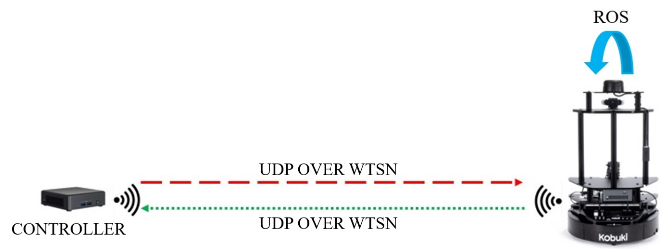


Fig. 2: Structure of the network communication and traffic flows between the involved devices.

B. Mobile Robot control application

NUC2 subscribes via a ROS service to LiDAR range values within an angular range from $-0.698rad$ to $0.698rad$, acquiring measurements every $0.01644rad$ thus limiting the number of acquired scan data to 67 scan measures per second. This allows to construct UDP frames, defined as frames on which the application level packet is added as a payload, with size of 1280 Bytes, reaching a good compromise between efficiency and reliability in the transmission to not affect the transmission conditions. Then the UDP frame is transmitted, via UDP, to NUC1 over WTSN.

Each UDP frame from NUC2 to NUC1, as shown in Fig. 3, comprises:

- the header, which includes sender and receiver device addresses, UDP frame length, and checksum
- the padding, which consists on 4 Bytes set to zero to prevent protocol encoding errors
- the timestamp
- the packet number
- the actual time
- the (x,y) position of the LoCoBot in the World Frame $\mathcal{F}_W$
- the values of the scan measures (ranges) acquired by the LiDAR, expressed both in the body frame of the LoCoBot $\mathcal{F}_L$ and with respect to $\mathcal{F}_W$. In details, to express the scan measures in $\mathcal{F}_W$, they are first transformed from polar coordinates to cartesian coordinates in $\mathcal{F}_L$ and, then, are transformed from $\mathcal{F}_L$ to $\mathcal{F}_W$.

Upon receiving a UDP frame from NUC2, NUC1 computes the suitable linear and angular velocities through the open-loop logic controller shown in Fig. 4. These calculations are based on the distance ρ in $\mathcal{F}_L$ and the angle α, which indicate the direction and orientation at which an obstacle is detected. The chosen linear and angular velocities are listed in Table I.

	$-50° \leq \alpha < -20°$	$-20° \leq \alpha < 20°$	$20° \leq \alpha < 50°$
$0.15 \leq \rho \leq 0.30$ [m]	$v = -0.1 \quad \omega = -0.35$	$v = -0.1 \quad \omega = -0.5$	$v = -0.1 \quad \omega = -0.35$
$0.3 < \rho \leq 0.35$ [m]	$v = 0.05 \quad \omega = -0.25$	$v = 0.05 \quad \omega = -0.3$	$v = 0.05 \quad \omega = -0.25$
$0.35 < \rho \leq 0.45$ [m]	$v = 0.25 \quad \omega = -0.15$	$v = 0.25 \quad \omega = -0.2$	$v = 0.25 \quad \omega = -0.15$
$0.45 < \rho \leq 0.55$ [m]	$v = 0.3 \quad \omega = -0.1$	$v = 0.3 \quad \omega = -0.15$	$v = 0.3 \quad \omega = -0.1$
$0.55 < \rho$ [m]	$v = 0.4 \quad \omega = 0$	$v = 0.4 \quad \omega = 0$	$v = 0.4 \quad \omega = 0$

TABLE I: Commanded linear velocity (v) [m/s] and angular velocity (ω) [rad/s] determined by spatial ranges.

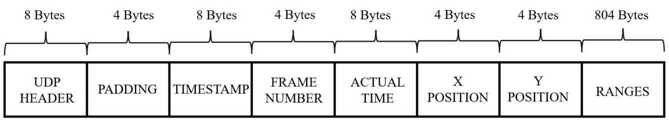


Fig. 3: Structure of the UDP frame transmitted by the LoCoBot to the edge controller (NUC2 → NUC1).

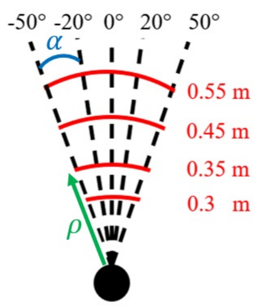


Fig. 4: Open loop controller idea.

Then the velocity values are transmitted back to the LoCoBot via UDP to steer its movements to guarantee safe exploration. The UDP frame structure from NUC1 to NUC2 is depicted in Fig. 5 and its size is of 256 Bytes.

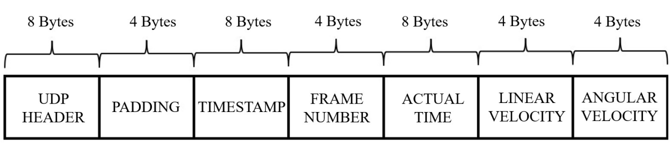


Fig. 5: Structure of the UDP frame that has linear and angular velocities values as payload (NUC1 → NUC2).

C. IEEE 802.1Qbv implementation

In addition to the TS traffic, the BE traffic has been generated using `iperf` and introduced on the same communication channel, directed from NUC1 to NUC2 (controller to LoCoBot), with the explicit goal of congesting the channel by consistently utilizing the entire available bandwidth. To effectively manage both TS and BE traffic, IEEE 802.1Qbv has been deployed to segregate these distinct traffic types in different protected windows to ensure reliable frame reception.

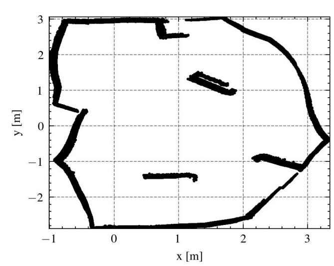
Hence, to define the scheduled protected windows, the $cycle_{time}$ is chosen so that it is given by the sum of three scheduled windows. The first one is equal to the transmission time of both TS frames, i.e., t_{TS}, the second one is for the BE traffic, while the last one works as a Guard Band (GB). The GB has been added to prevent congestion and to enhance network reliability and performance.

Despite t_{TS} can be computed as the ratio between size of the frame and bandwidth, this scheduling does not guarantee the correct execution of IEEE 802.1Qbv due to the presence of beacon frames [14]. To avoid this, the time reserved to each scheduled window, i.e., TS, BE and GB, must be increased by a predefined amount to ensure a reliable scheduling.

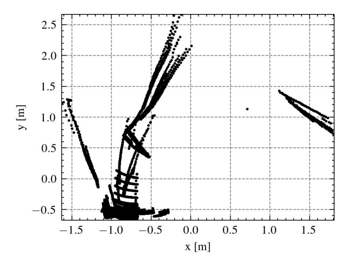
III. Experimental Results

In this section, we analyze the mapping results achieved in three different scenarios, namely when: *Experiment #1*: no BE traffic is added; *Experiment #2*: BE traffic is added in the Wi-Fi channel direction from controller to LoCoBot; *Experiment #3*: IEEE 802.1Qbv is enabled to manage both the TS and BE traffics. In all these scenarios the mapping process has been facilitated by the incorporation of a Kalman filter to effectively correct any drifting discrepancies arising from the utilization of odometry and Inertial Measurement Unit (IMU) sensors.

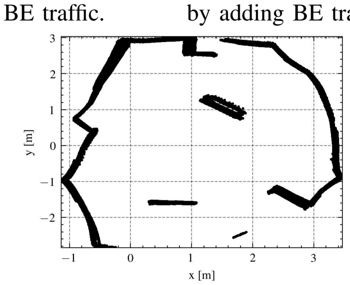
The obtained outcomes are shown in Fig. 6.



(a) Experiment #1: Map obtained without adding BE traffic.



(b) Experiment #2: Map obtained by adding BE traffic.



(c) Experiment #3: Map obtained adding BE traffic and enabling IEEE 802.1Qbv.

Fig. 6: Mapping outcomes in the three Experiments.

The results show how, without any added BE traffic, the controller can generate a precise map of the environment.

Instead, adding the BE traffic on the channel direction from controller to LoCoBot, the controller cannot build an accurate map, as shown in Fig. 6b, and consequently the LoCoBot moves randomly in the environment. This occurs because the BE traffic interferes with the TS one containing the velocities commands.

Then, by adding IEEE 802.1Qbv, as illustrated in Fig. 6c, the controller can build again an accurate map and avoid the static obstacles, proving the capability of IEEE 802.1Qbv to handle correctly the two kinds of traffic, i.e., TS and BE.

Regarding latency, the probability density functions of the latency from LoCoBot to controller and from controller to LoCoBot are shown for all the three experiments in Fig. 7.

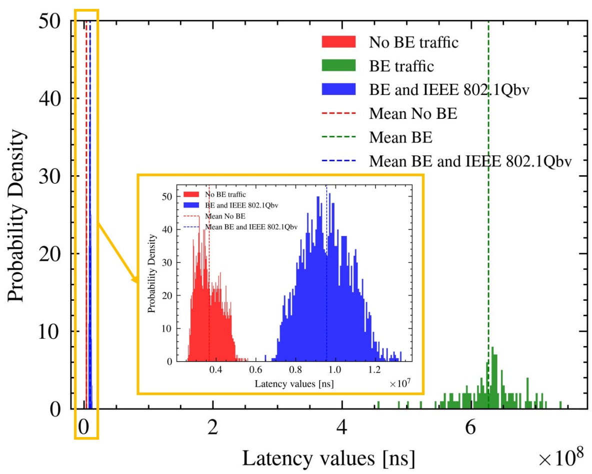


Fig. 7: Distributions and mean values of latency under three scenarios: without BE traffic, with BE traffic, with BE traffic and IEEE 802.1Qbv enabled.

Note that such latency values are computed as the time difference between the time instant at which a UDP frame is received and the one at which the same UDP frame was sent, thanks to the implementation of the IEEE 802.1AS time synchronization mechanism.

The mean and standard deviation values of the latency from controller to LoCoBot are given in Table II.

	mean [ms]	std [ms]	max [ms]
no BE	3.660	0.632	5.572
with BE	626.590	46.368	739.888
with BE and IEEE 802.1Qbv	9.557	1.2257	13.304

TABLE II: Latency metrics from controller to LoCoBot.

As expected, considering the mean latency values acquired when BE traffic is added without TSN capabilities, both the mean values increase with respect to the case without BE traffic due to the fact that BE and TS traffic compete for resources, thus leading to delay in the UDP frame transmission.

Then, adopting the IEEE 802.1Qbv standard, the mean latency values decrease and almost reach the values obtained without the addition of any BE traffic, demonstrating the effectiveness of the 802.1Qbv capability to schedule TS and BE traffic and avoid congestion delay.

IV. Conclusions and Future works

Our study presents a smart manufacturing application using a LoCoBot for real-time mapping, leveraging WTSN to manage time-critical data and navigate static obstacles, reducing wired connection complexity. We highlight the efficacy of IEEE 802.1Qbv in traffic scheduling to ensure low latency and successful task completion.

Regarding future works, we plan to integrate Model Predictive Control (MPC) to enhance mapping and obstacle avoidance, thus offering an adaptive navigation in dynamic environments. Additionally, we plan to use a data-driven approach to create a more accurate modeling, crucial for better predicting latency and characterizing Wi-Fi channels.

References

[1] AIOTI, "Aioti strategic research and innovation agenda (sria) 2023 – advancing next-generation iot and edge computing research and innovation," https://aioti.eu, 2023, accessed: 2025-07-14.

[2] S. Santos, P. Costa, and A. Rocha, "It/ot convergence in industry 4.0 : Risks and analisy of the problems," in *2023 18th Iberian Conference on Information Systems and Technologies (CISTI)*, 2023, pp. 1–6.

[3] S. Sudhakaran and et al., "Measurement method for end-to-end time synchronization of wired and wireless tsn," in *2023 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2023, pp. 1–6.

[4] M. K. Atiq, R. Muzaffar, Seijo, I. Val, and H.-P. Bernhard, "When ieee 802.11 and 5g meet time-sensitive networking," *IEEE Open Journal of the Industrial Electronics Society*, vol. 3, pp. 14–36, 2022.

[5] J. Hicking, M.-F. Stroh, and S. Kremer, "Collaboration through Digital Integration – An Overview of IT-OT-Integration Use-Cases and Requirements," in *IFIP Advances in Information and Communication Technology*, ser. Smart and Sustainable Collaborative Networks 4.0, L. M. Camarinha-Matos, X. Boucher, and H. Afsarmanesh, Eds., vol. AICT-629. Saint-Etienne, France: Springer International Publishing, Nov. 2021, pp. 403–410, part 11: Digitalization Strategy in Collaborative Enterprises Networks. [Online]. Available: https://hal-emse.ccsd.cnrs.fr/emse-03345986

[6] V. Frascolla and et al., *Lesson Learnt and Future of AI Applied to Manufacturing*, 01 2022, pp. 207–240.

[7] S. Sudhakaran and et al., "Wireless time sensitive networking for industrial collaborative robotic workcells," in *2021 17th IEEE International Conference on Factory Communication Systems (WFCS)*, 2021, pp. 91–94.

[8] Y. Liu, M. Kashef, K. B. Lee, L. Benmohamed, and R. Candell, "Wireless network design for emerging iiot applications: Reference framework and use cases," *Proceedings of the IEEE*, vol. 107, no. 6, pp. 1166–1192, 2019.

[9] T. Fedullo, A. Morato, F. Tramarin, L. Rovati, and S. Vitturi, "A comprehensive review on time sensitive networks with a special focus on its applicability to industrial smart and distributed measurement systems," *Sensors*, vol. 22, no. 4, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/4/1638

[10] V. Frascolla, D. Cavalcanti, and R. Shah, "Wi-fi evolution: The path towards wi-fi 7 and its impact on iiot," *Journal of Mobile Multimedia*, vol. 19, no. 1, pp. 263–276, 2023.

[11] S. Sudhakaran and et al., "Zero-delay roaming for mobile robots enabled by wireless tsn redundancy," in *2023 IEEE 19th International Conference on Factory Communication Systems (WFCS)*, 2023, pp. 1–8.

[12] A. Merwaday and et al., "Communication-control co-design for robotic manipulation in 5g industrial iot," in *2023 IEEE 21st International Conference on Industrial Informatics (INDIN)*, 2023, pp. 1–6.

[13] A. Baxi and et al., "Towards factory-scale edge robotic systems: Challenges and research directions," *IEEE Internet of Things Magazine*, vol. 5, no. 3, pp. 26–31, 2022.

[14] A. Morato, S. Vitturi, F. Tramarin, C. Zunino, and M. Cheminod, "Time-sensitive networking to improve the performance of distributed functional safety systems implemented over wi-fi," *Sensors*, vol. 23, no. 18, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/18/7825