# Next-Generation Industrial Networks: Integrating Time-Sensitive Networking for Smart Factory Reliability

Yazen S. Sheet, Mohammed Younis Thanoun, and Firas S. Alsharbaty

***Abstract*—Traditional industrial communication networks may not meet the requirements of the main smart factory applications, such as remote control and safety applications (which can be considered strictly real-time) and augmented reality (which consumes a large bandwidth). This work suggests an enhanced communication network that can serve an optimum case for the smart factory model to include heavy data and real-time applications. The concepts of Time-Sensitive Networks (TSN) are applied to address the limitation of real-time applications, whereas edge computing is used to assimilate the heavy data applications. The current work considers an experimental scenario where the H.265 compression method based on the concept of edge computing was used. This was done to mitigate the impact of the consumed capacity on the performance of the overall network for an augmented reality application. The results of implementing the enhanced communication network indicated that the latency of real-time applications is less than 1 ms, while the packet data delivery rate for rest applications is 99.999%.**

***Index terms*—Industry 4.0, Smart Factory, TSN, Edge Computing, H265.**

## I. Introduction

Smart factories, are considered one of the revolutionary concepts of Industry 4.0, represent a groundbreaking shift in the manufacturing sector. They are characterized by the integration of digital technologies and automation to create highly efficient, flexible, and intelligent production environments. Indutrial Internet of Things (IIoT) devices are used to connect machinery, sensors, and systems, enabling real, semi-real and non-real-time data exchange. This connectivity allows for seamless communication across the production line, facilitating quick responses to changes and challenges [1]. There are several applications in smart factories with different requirements in terms of delay, bandwidth, and reliability of data transmission. Some of such applications involve bounded latency requirements (for instance, a robotic arm may be required to pick up an object that moves on a conveyor belt at a specific time), while others may have high bandwidth and reliability requirements, such as Augmented Reality (AR) applications.

Table I shows the most essential applications used in smart factories and their requirements in terms of delay, reliability, and data rates [2]-[4]. To achieve the requirements of these applications, it is essential to establish a reliable communication network capable of transmitting data within a predetermined timeframe, which is a significant challenge for researchers in this field. Fieldbus communications networks such as Controller Area Network (CAN) and industrial Ethernet protocols such as EtherCat developed to accomplish this objective [5, 6]. The main problem with such protocols is the need for interoperability among them. Therefore, in 2012 a group of researchers within the IEEE Working Group developed a new approach known as Time-Sensitive Network (TSN), established under the IEEE 802.1 standard.

In this approach, data are classified into several categories based on the type of application in use and are then placed in separate queues according to the relative significance of each category. Types of data are served using various mechanisms and methods. These include transmission mechanisms such as Strict Priority (SP), specified in the 802.1Q standard, and the Credit-Based Shaping (CBS) defined in the 802.1av standard. Other mechanisms that deal with the queue itself, such as Time-Aware Shaping (TAS) described in the 802.1bv standard and Frame Preemption (FP), introduced in the 802.1bu standard. This trend supports the need to transmit different kinds of sensitive and non-sensitive data applications on the same network while preserving the speed of traditional Ethernet networks [7]-[9].

TABLE I
MAPPING REQUIREMENTS OF MAIN SMART FACTORY APPLICATIONS

| Application | Requirements | | |
|---|---|---|---|
| | Latency | Bandwidth | Availability |
| Automated Guided Vehicles | Medium (10-20ms) | Variable (<100Mbps) | 99.9999% |
| Augmented Reality | Medium (10-20ms) | High (<1Gbps) | 99.99% |
| Remote Control | High (0.1-1ms) | Low | 99.9999% |
| Monitoring and Predictive Maintenance | Low (20-100ms) | Variable | 99.99% |
| Safety and Protection | Medium (<10ms) | Low (<1Mbps) | 99.9999% |

Manuscript received April 1, 2025; revised October 24, 2025. Date of publication July 1, 2026. Date of current version July 1, 2026.
Authors are with the College of Engineering, University of Mosul, Iraq ({yazenalnuaimi, myounisth, Alsharbaty}@uomosul.edu.iq).
Digital Object Identifier (DOI): 10.24138/jcomss-2025-0047

The contributions of the present paper in comparison with the recent related works can be summarized as follows:

- To present a communication network for a typical case of the smart factories to deal with the essential candidate applications for these factories in the era of Industry 4.0.
- To model most important TSN algorithms in this work to support the performance of industrial communication network and meet the requirements of the adopted smart factory applications.
- To enable heavier AR applications to be assimilated into industrial networks, this work employs emerging technologies in the form of edge computing and data compression to control the performance of an industrial communication network to meet the requirements of such applications.
- H265 video compression technology based on the concept of edge computing, has been practically applied to AR video data captured in an industrial environment to determine the appropriate compression ratio. This process carried out without affecting the quality of the AR video data, which has been also evaluated experimentally to achieve an optimum balance between the compression ratio and the QoS for transmission over a communications network.

The remainder of this paper is divided into seven sections. The related works have been presented in Section II. Descriptions of the TSN and emerging technology are given in Sections III and IV, respectively, while Section V provides an explanation of the research methodology. Section VI presents the adopted scenarios, and Section VII discusses the results. Lastly, Section VIII explains the most significant conclusions of the work.

## II. Related Works

TSN mechanisms had been examined by many authors in their research, implemented them, and evaluated their performance in different industrial scenarios. For example, the authors of [10, 11] presented a comprehensive framework for the integration of TSN features into standard Ethernet and emphasized its potentiality to satisfy the Quality of Service (QoS) demands of industrial applications. They carried out simulation experiments to demonstrate the efficacy of TSN features, including SP, FP, and TAS; the results indicated that TAS generally outperformed FP and ST regarding latency and jitter, although it imposed considerable configuration overhead. The authors of [12]-[14] designed an extensive simulation model using the OMNET++ and MATLAB simulation packages, which was specifically intended to assess TSN protocols and their efficacy in industrial applications.

They presented an implementation of the Gate Control Lists (GCLs) of the IEEE 802.1 Qbv standard in a simulation model that included FP to guarantee deterministic transmission of time-sensitive data.

The authors of [15]-[17] offered a complete analysis of several TSN latency management methods, such as FP, CBS, and GCL. They examined the influence of these systems on end-to-end (E2E) delay and presented optimization methods in which a surrogate model was employed to ascertain optimal configurations for TSN networks, hence enhancing E2E delay to meet the varied QoS demands of the many data streams produced during the quality inspection process in a smart factory. They have performed comprehensive simulations across multiple scenarios to evaluate the performance of many network configurations.

The work in [18] evaluated the effectiveness of TSN networks within IIoT environments by employing standard topologies and messaging protocols and proposing scenarios for assessing traffic management. It offered a comparative review of the various technologies employed for modeling services, with an focus on the latency, channel efficiency, jitter, and queuing behavior across different congestion levels.

The article examined the application of edge computing to improve network performance and presented scenarios for both centralized and edge deployment strategies to improve service delivery.

The work [19] examined the optimization of multi-shaper TSN configurations for industrial applications, emphasizing the integration of diverse traffic shaping methodologies, such as TAS and CBS, it established a combinatorial optimization problem for configuring TSN networks, utilizing a simulated annealing metaheuristic to minimize end-to-end delays while maintaining the stream deadlines. The authors in [20] offered a schedulability analysis of TSN that incorporates TAS and FP to guarantee deterministic communication for time and safety-critical applications, especially in industrial control and automotive systems. They presented a formal timing analysis methodology employing Compositional Performance Analysis (CPA) to determine worst-case delay constraints for traffic in TSN networks, utilizing simulation results from OMNeT++ across diverse network circumstances. The results indicated improvements in minimizing delay estimates, therefore enhancing the reliability of TSN in real-time applications.

The paper in [21] explored the adoption of TSN inside centralized architectures for Software Defined Vehicles, (SDV) focusing on the need for reliable and low-latency communication in automotive networks. The authors examined diverse TSN protocols and techniques, including traffic shaping and synchronization to fulfill stringent QoS requirements. Using formal analysis and simulations, the authors illustrated how an adapted TSN configuration may effectively manage varied data streams, guaranteeing optimal performance and minimal latency.

The aforementioned works outlined certain shortcomings, as they did not utilize a complex communication network capable of supporting the main applications of a smart factory in terms of the reliability and latency requirements of these applications; instead, they concentrated exclusively on supporting time-sensitive applications within a simple network model, neglecting the other applications and their impact on real-time applications, as well as the challenges of establishing TSN mechanisms and analyzing their performance in detail.

Table II summarizes the main contributions of related works and compares them with this work (part 1 in terms of adopted algorithms and part 2 in terms of emerging technologies and applications).

TABLE II
COMPARISON AMONG THIS WORK AND PREVIOUS RELATED WORKS

| Ref. | Adopted Algorithms | | | | |
|---|---|---|---|---|---|
| | | TSN Mechanisms | | | |
| | (FIFO) | SP | CBS | TAS | FP |
| [10] | √ | √ | | √ | √ |
| [11] | | √ | | √ | |
| [12] | | | | √ | |
| [13] | | √ | | √ | √ |
| [14] | | | | √ | √ |
| [15] | | √ | √ | √ | √ |
| [16] | | | | √ | √ |
| [17] | √ | √ | √ | √ | |
| [18] | √ | | √ | | |
| [19] | | | √ | √ | |
| [20] | | | | √ | √ |
| [21] | | | | √ | |
| This Work | √ | √ | √ | √ | √ |

Emerging Technologies and Applications Kind

| Ref. | Emerging-Technology | Applications Kind | | | |
|---|---|---|---|---|---|
| | | RT | SRT | NRT | BE |
| [10] | | √ | | | √ |
| [11] | | √ | | | √ |
| [12] | | √ | | | √ |
| [13] | | √ | | | √ |
| [14] | | √ | | | √ |
| [15] | √ | √ | √ | | √ |
| [16] | | √ | √ | | √ |
| [17] | | √ | √ | | √ |
| [18] | √ | √ | √ | √ | √ |
| [19] | | √ | | | |
| [20] | | √ | | | |
| [21] | | √ | | | |
| This Work | √ | √ | √ | √ | √ |

FIFO: First In First Out, RT: Real Time, SRT: Semi Real Time, NRT: Non-Real Time, BE: Best Effort

## III. TIME-SENSITIVE NETWORKS (TSN)

TSN are communication networks that carry out a time-bound transfer of information using standard Ethernet technology. It was developed by a research group as part of the IEEE working group under the IEEE 802.1 standard. This technology is used in industrial applications, aircraft, vehicles, and the fields of health and transport. TSN operates within the second layer (MAC layer) of (OSI layers Ref.) in which a 4 byte tag is added to the standard Ethernet header. Different standards are applied to TSN to meet the required functionalities [7].

The IEEE 802.1AS generalized precision time protocol (gPTP) is a standard that was derived from IEEE 1588 and is more efficiently designed for applications that require precise timing. Similarly to 1588, the 802.1AS protocol manages the synchronization among a sequence of nodes that are linked by Ethernet cables, switches, and bridges [9].

According to IEEE 802.1Q, standard bridging employs SP algorithm as a default transmission selection approach for the queues in the output ports of a TSN switch. The function of these queues is based on the queuing mechanism that can be defined as the process that is employed to manage the network traffic by storing packets in these queues prior to forwarding them to the output port when their arrival rate exceeds the transmission bandwidth. There are eight queues in the output port that correspond to eight distinct traffic classes, each is allocated a unique priority level. Frames belonging to the same traffic class are often selected using a First in, First out (FIFO) approach.

The priority levels are established based on the value of the Priority Code Point (PCP) field in the 802.1Q tag of a new Ethernet frame, where eight types of services can be defined using this field. According to the SP mechanism, the traffic with PCP equal to 7 has been sent to queue 7, and so on. The highest priority queue (7) frames have been transmitted over the rest of the queues with lower priority frames that make them suffer from long waiting and lost in the queue [23].

Table III represents the mapping between priority and traffic classes. The original Ethernet frame header is appended with this tag to create the TSN frame header [22], as illustrated in Fig. 1.

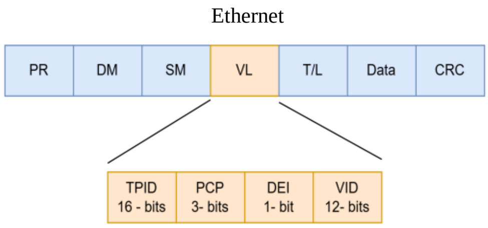


Fig. 1. IEEE 802.1 Q Tag.

Another algorithm for frame transmission is called Credit-Based Shaper (CBS) under the IEEE 802.1Qav standard, the transmission process due to this algorithm does not depend only on the value of the queue's priority; instead, it relies on the concept known as a credit. This algorithm added a kind of justice to other queues with the lowest priority by dedicating a specific capacity for each queue that uses this algorithm.

According to this algorithm, there are two significant terms [24]:

*IdleSlope:* This represents the rate at which a flow accumulates credits. It occurs during the period while frames are still awaiting transmission in a queue. IdleSlope is crucial when deciding the amount of bandwidth allocated to a flow, which can be calculated as shown in Equation 1:

$$\text{bandwidthFraction} = \frac{\text{IdleSlope}}{\text{portTransmitRate}} \quad (1)$$

*SendSlope:* This represents the rate of credit decrease when frames are being transmitted from the queue and can be calculated by Equation 2:

$$\mathrm{SendSlope} = \mathrm{IdleSlope} - \mathrm{portTransmitRat} \quad (2)$$

TABLE III
PRIORITY AND TRAFFIC CLASS MAPPING

| Traffic Type | PCP | Priority |
|---|---|---|
| Background | 1(lowest) | 0 |
| Best effort | 0(default) | 1 |
| Excellent effort | 2 | 2 |
| Critical Application | 3 | 3 |
| Video | 4 | 4 |
| Voice | 5 | 5 |
| Internetwork Control | 6 | 6 |
| Network Control | 7 (highest) | 7 |

The TAS defind in the IEEE 802.1Qbv standard, is regarded as the most important mechanisms for TSNs and has received a significant attention from researchers. It it used to support time-sensitive scheduled traffic, such as remote-control applications in many domains, including factories, autos, transportation, and others. In this standard, there is a gate for each of the eight queues, and each gate has two possible states: open or closed. The operation of these gates is controlled by a specialized module known as the GCL. When a queue gate is closed, transmission from that queue is not possible, whereas when the gate is open, transmission occurs from the queue associated with this gate. This mechanism ensures that time-critical messages reach their destination within a specific amount of time. There are three periods during one cycle of sensitive applications transmission, known as the protected window, unprotected window, and guard band [25]. Fig. 2 illustrates the concept underlying this standard.

FP, identified in the IEEE 802.1Qbu standard, is another TSN algorithm employed to decrease the delay for the highest-priority time-sensitive application frames by interrupting and halting the transmission of lower-priority application frames upon the arrival of a higher-priority frame.

The transmission of lower-priority frames resumes once the higher-priority frames have been sent. This operation may be iterated multiple times whenever a frame with a higher priority arrives while transmitting frames with a lower priority [26].

## II. EMERGING TECHNOLOGIES IN INDUSRTY 4.0

Emerging technologies in Industry 4.0 are transforming manufacturing and industrial processes by integrating advanced digital technologies, including IIoT, AI, ML, big data analytics, and edge computing. This work employs edge computing technology, which emphasizes the execution of computations near the data source. Edge computing resources may consist of networks or computers operating between end users and cloud data centers. IIoT devices generate enormous amounts of raw data that may exceed the capacity of the traditional cloud computing paradigm, meaning that the majority of IIoT data will be consumed at the network edge rather than being transferred to the cloud. The cloud computing model may also be insufficient for many other reasons. For instance, the volumes of data at the edge may be excessive, using significant bandwidth and computational resources unnecessarily. Moreover, the necessity for privacy protection prevents the use of cloud computing in IIoT.

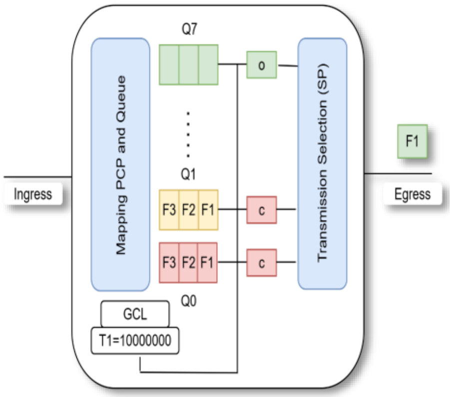


Fig. 2. TAS Concept.

Hence, the concept of edge computing has emerged to handle such cases [27]. Fig. 3 illustrates the concept of edge computing.

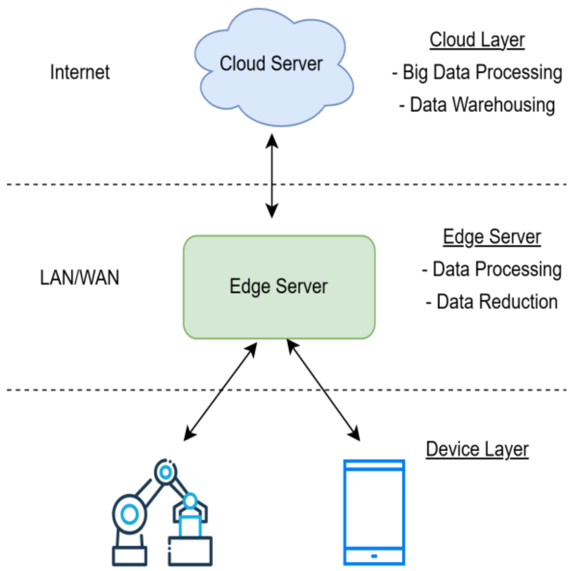


Fig. 3. Edge Computing Concept

In this context, data compression is one approach that can be combined with edge computing concepts to enhance the system performance. The most widely used form of data compression is video compression, which is based on several regularly applied standards: H.264/AVC, VP9, AV1, and H.265/HEVC [28]. By using the H.265/HEVC technique, the streaming services can enhance the user experience and reduce data transmission costs by providing higher-quality videos with manageable bandwidths. H.265 is a codec that replaces H.264, offering a compression ratio that is up to 50% greater than that of H.264, hence enhancing its applicability across several domains..

Compression solutions that rely on edge computing do have certain drawbacks, one of which is the processing time required for encoding and compression, which is around 20 ms per megapixel when using a 50:1 compression ratio [29].

## III. Methods and Materials

### A. Model Description

The smart factory communication network that has been implemented in this work addresses an industrial unit that is made up of a collection of components that are representative of the most important applications of a smart factory. The designed model includes the following components:

Condition Monitoring Sensors (CMS): This is a collection of monitoring sensors linked to machineries and equipment and are used to assess their operational conditions before transmitting regular data to a local controller.

Actuators Control (AC): This is a local controller that receives and analyzes the data from the condition monitoring sensors and then sends a control signal to actuators (ACTs) to carry out a specific task in the case where the conditions are defective.

Safety Sensor (SS): Safety sensors are included to monitor the conditions in the surrounding environment, including any changes that may take place. Data are periodically sent to the monitoring unit to alert it about any emergency situation that may occur. For example, if a fire or gas leak occurs, a decision must be taken regarding appropriate action.

Safety Monitor (SM): Data are collected on a regular basis from safety sensors to monitor the various factors and situations that affect the workers and factory floor.

Augmented Reality (AR): One of the most important applications in modern systems is AR, an interactive application that permits users the ability to interact virtually with industrial units. This is accomplished by representing physical objects and components in digital reality, displaying them, and allowing the user to interact with them through technological devices such as tablets and laptops. One of the application's main uses is to allow a remote expert to fix a problem by aiming a high-resolution camera at industrial machinery and transmitting video streams with information, thus enabling the expert to interact with the system and provide the necessary notes to resolve the issue, as assumed in this work. The AR_Sender (ARS) and AR_ Receiver (ARR) elements represent this application in the proposed model.

RoboticArm and Control: Industrial units involve various continuous production processes, which require effort to complete. Examples of these production processes include packaging, sorting, and sealing a product. These require the presence of controlled automated arms that can carry them out with high precision and without delay or error. This application, which can be considered as a remote-control application, is represented in the proposed model using two elements: a RoboticController (RC) for sending control signals and an RoboticArm (RA) to carry out these objectives.

AGVs and Control: The model also incorporates mobile vehicles, representing the use of AGV for transporting items within the industrial unit. This application is represented in the model by AGVs and Vehicle Control (VC) components. As the AGVs move into a particular area, they transmit data to their control unit, which then sends control signals to the AGVs if there is a condition that requires changes to the path taken by these vehicles or modifications to some of their functions.

Update Unit (UU): This unit is responsible for generating best-effort data, which is then distributed to all applications within the network for the purpose of updating the software of these applications.

Data_Center (DC): A data storage unit receives reports from the Actuator Control unit about the condition of the industrial unit after analyzing the data received from the condition sensors.

### B. The Suggested Communication Network for the Adopted Model

It is obvious from Fig. 4 that the suggested communication network for the described model depends on a wired network based on Ethernet technology. This provides suitable shielding for industrial purposes in addition to handling the available commercial types of communication network technologies. It comprises a set of applications that represent a typical industrial case, interconnected by Ethernet switches that support TSN algorithms. These nodes are linked via a 100 Mbps Ethernet link, which is widely used commercially for industrial cases, as well as the core link between switches A and B that has been set at a 100 Mbps Ethernet link to show the effect of TSN algorithms. Switches (C and D) in the network are used to distribute and balance the load, thereby enhancing the reliability of the network applications. Switch A connects the edge networks to the network backbone, while switch B finalizes the connections of all components to the designated network. The model specifications, number of application nodes, and size of the data are shown in Table IV. It is worth mentioning that the time delay, bandwidth, and availability requirements of these applications have been explained in Table I.

TABLE IV
Specifications of Model

| Application | Representing Node | No. | Packet Size (B) | Packet Interarrival time |
|---|---|---|---|---|
| AR | AR_Sender | 1 | 1500 | 12μs |
| AR | AR_Receiver | 1 | 1500 | 40ms |
| Condition Monitoring | Condition_Sensor | 10 | 500 | 100ms |
| | Actuators | 10 | ----- | |
| Safety | Safety Sensor | 1 | 500 | 100ms |
| | Safety Monitor | 1 | ----- | |
| Automated Guided Vehicle | AGVs | 2 | 1500 | 500μs |
| Remote Control | Robottcontroller | 1 | 354 | 400μs |
| | Actuators Control | 1 | 354 | Depend on condition |
| | Vehicles Control | 1 | 354 | Depend on condition |

The network model is simulated via the OMNeT++ modeler, mostly based on the INET framework, which includes most components of both wired and wireless networks, allowing for their reuse in the constructed models.

Alongside this framework, NeSTing [30] is an additional framework that specializes in simulating TSN and can be incorporated into the work environment during the design of the suggested models.

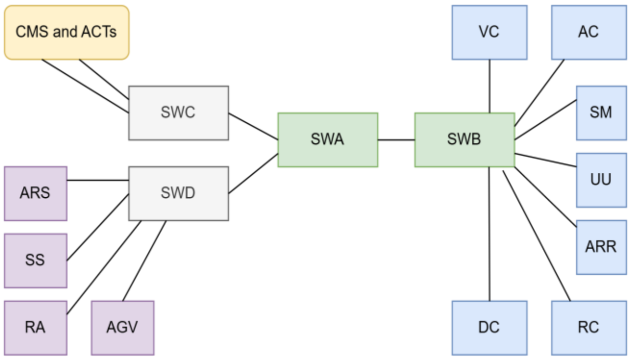


Fig. 4. Network Model of Industrial Case.

## IV. The Adopted Scenarios and Assumptions

### *A. Basic Scenario*

A conventional Ethernet network scenario is presented as the basic scenario without prioritization, utilizing the First-In-First-Out FIFO scheduling algorithm to manage the network traffic for a singular queue at the network switches.

This algorithm does not ensure Quality of Service (QoS) for applications, as data is processed according to the FIFO principle.

### *B. Time Sensitive Network Scenarios*

The TSN algorithms (SP, CBS, TAS, FP) described before, have been modeled in these scenarios of the adopted model.

The applications of the network were categorized into eight classes, each class mapped to a separate queue in the network switches depending on the value in the 3-bit PCP field of the frame header, which is determined by the application's requirements. The PCP values assigned to the applications utilized in the model are given in Table V.

TABLE V
PCP Value of Applications

| No. | Application | PCP Value |
|---|---|---|
| 1 | Remote Control | 7 (highest) |
| 2 | Safety and Protection | 6 |
| 3 | Augmented Reality (AR) | 5 |
| 4 | Automated Guided Vehicle | 4 |
| 5 | Condition Monitoring | 3 |
| 6 | Update | 2 (lowest) |

### *C. Enhanced Scenario*

In this scenario, the edge computing and TSN algorithms were exploited to process the AR application data close to its source of generation. The bit rate of AR is high due to its video nature, which may affect other applications in the context of traffic received and delay. To manage the large data volume and conserve network resources, the AR data are compressed by appropriate compression techniques based on edge computing technology. H.265 is the compression technique used in this scenario to compress the AR data.

Several factors can be used to measure the efficiency of the compression method, such as the Compression Ratio (CR), which can be defined as a critical metric for assessing the efficacy of a compression technique, it is the ratio between the size or data rate of the source video to the size or data rate of the compressed video. An example of CR that shows how much the file size has reduced is 50:1. CR can be calculated by Equation 3, and the Peak Signal-to-Noise Ratio (PSNR), which measures the quality of the compressed data by comparing it with the original data. It can be computed by Equation 4, and is measured in dB.

$$CR = \frac{\text{Uncompressed Size}}{\text{compressed Size}} \quad (3)$$

$$PSNR = 10 \cdot \log 10 \left(\frac{\text{Max}^2}{\text{MSE}}\right) \quad (4)$$

where Max is a maximum possible pixel value, and MSE is the mean squared error between the pixel values of the uncompressed and compressed video versions.

In this scenario, the compression ratio was practically determined using the well-known HandBrake software [31], implemented on an edge server and applied to a Full HD video that was recorded in an industrial environment using an industrial HD camera. The compression process included applying the H.265 compression method and iteratively changing the Constant Quality (RF) value to attain an appropriate file size while keeping good video quality.

After compression, the PSNR was empirically measured using the MSU Video Quality Measurement Tool on the same edge server to evaluate the resulting video quality [32].

Subsequently, the Mean Opinion Score (MOS) was computed based on the relationship depicted in Fig. 5, illustrating the score of delivered service quality when this score is on a 1 to 5 scale, the scores are 1: unacceptable, 2: Poor, 3: Fair, 4: Good, 5: Excellent. Table VI represents the specifications for the edge computing server.

TABLE VI
Edge Server Applications

| NO. | Specification |
|---|---|
| 1 | CPU Intel Core i9 |
| 2 | 64 – bit OS (Operating System) |
| 3 | 32 GB RAM (Random Access Memory) |
| 4 | 500 GB dynamically allocated Storage |

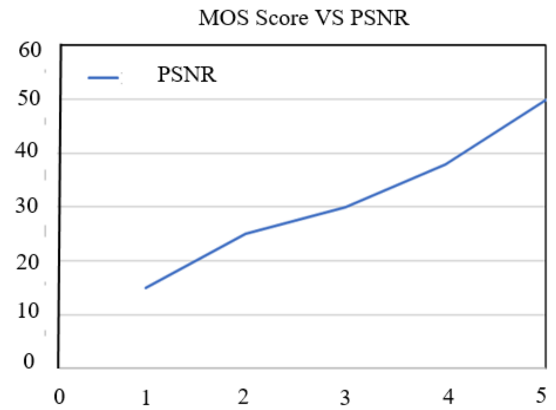


Fig. 5. MOS and PSNR Relationship [33].

### *D. Upgrade Scenario in terms of Physical Cables*

In the earlier scenarios, the bandwidth of the links connected the nodes of the network was 100 Mbps. In this scenario, in addition to the TSN algorithms and emerging technologies, the bandwidth is increased by replacing the 100 Mbps cables with 1 Gbps Ethernet cables for all links of the network. In this case, the effect of enhancing the physical resources (from the point of view of cables) has been investigated on the performance of the network. Fig. 6 presents a flowchart for the model methodology.

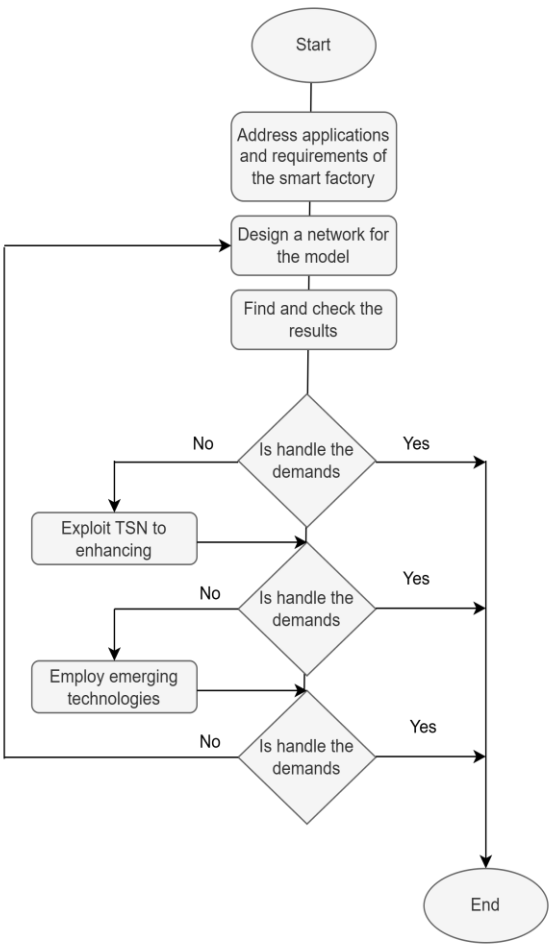


Fig. 6. Flowchart of Model Methodology.

## V. Results and Discussion

This section examines and discusses the most notable results of executing the proposed scenarios of model. The metrics of assumed applications of network model were selected in context of traffic and delay time as described below:

- Percentage of Received Data Reliability (RDR%) can be calculated through the following equation:

$$\frac{\text{Traffic Received of Application}}{\text{Traffic Sent of Application}} \cdot 100$$

- End to End delay: The amount of time for frame transmission from source to destination.

The RDR% value for each application in the basic scenario using the FIFO technique is shown in Fig. 7. Since the output port of a switch contains only a single queue for all traffic and the service is provided based on the order of frames arrival to this queue according to port data rate, thus, this algorithm does not ensure the QoS for certain applications.

The figure shows that most applications experience a reduction in the percentage of RDR traffic, indicating the algorithm’s unreliability for these applications. This is particularly true for the highest priority robot controller application, which has a strong real-time requirement, since not all of its data reached the destination. Conversely, the RDR% value for the AR application is approximately 8%, which is the lowest, due to the high bit rate of 995 Mbps generated by a full HD camera with a resolution of 1920×1080.

Fig. 8 illustrates the E2E delay, indicating discrepancies between the recorded delays and their respective requirements for the adopted applications. For example, the delay for the remote-control application reaches 4 ms, whereas its required range is 0.1-1 ms. Similarly, the safety application experiences delay value exceeding 12 ms, while its maximum acceptable limit is 10 ms. Conversely the AR application exhibits a substantially higher delay, approximately 900 ms, due to long waiting times of its frames in the queue, which leads to the loss of the majority of these frames.

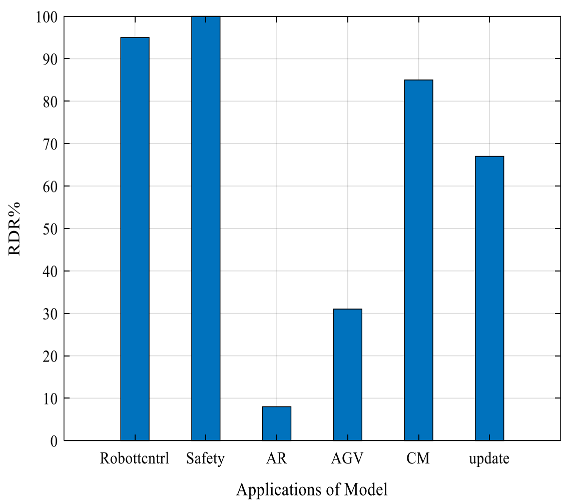


Fig. 7. RDR% for Model Applications in FIFO Scenario.

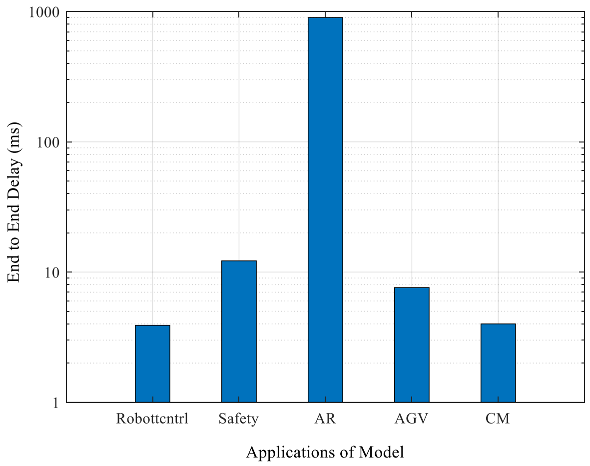


Fig. 8. End to End Delay for Model Applications in FIFO Scenario.

The results for the TSN scenario are presented in Table VII, wherein the applications are classified into various categories, where the highest priority applications being processed first, followed by the remaining ones. The SP algorithm is utilized as a default method for data retrieval from the queues, prioritizing the highest-priority applications for servicing at all times. As shown in the table, both the remote control and safety applications are fully serviced. It can also be observed that the AR application consumes the entire bandwidth of the link to transmit just 10% of its data, approximately 97 Mbps, thereby, completely preventing lower-priority applications from being serviced., as indicated by the results which show that no data received for the AGV and CM applications. The CBS is the second method used for serving applications from the queues. It is less strict than SP, as it distributes the service among queues based on the IdleSlope value. In general, it is used for video applications, and is allocated here as a default to send AR data from its queue (5).

The results have been collected at IdleSlope values ranging from 0.1 to 0.9. At a value range of 0.1 to 0.5, the RDR% for AR application diminished to 5%, which is near to the previous 10%, indicating failure in both cases. Conversely, the RDR% for other applications (AGV, CM) increased to 100%.

Following a value of 0.5, the condition getting worse, nearing the SP case, especially at 0.9 value of IdleSlope; thus, the optimal value for IdleSlope, to be established in remaining scenarios is 0.5.

In the TAS scenario, the protected window is set to 140 µs, which corresponds to the approximate arrival time of the remote-control application frames from source to destination. The unprotected window is 220 µs, and the guard band is 40 µs. The combined three periods are equal to 400 µs, which is the interval time between the transmission of the remote control application frames. By controlling the queue gates, this mechanism prioritizes time-sensitive applications over other traffic. As a result, the AR and AGV applications experience a lower RDR ratio compared to the CBS (0.5) scenario. Conversely, in the FP scenario, the results are better when the queue gates remain open; only the queues with the lowest-priority are interrupted to transmit frames from the highest-priority queue.

Table VIII shows the E2E delay for the TSN scenarios. The results show that the delay for the remote-control application is minimum in all scenarios; this is most evident in the TAS scenario, where the delay reaches 0.138 ms, while it is slightly higher in the FP scenario due to its high priority. The safety application requires a minimal bandwidth of approximately 1 Mbps, resulting in no loss or delay in all scenarios, its delay is marginally more than that of the remote control application, maintaining a consistent value across all scenarios.The AR application experiences the highest delay among all TSN scenarios, due to its large data size and the network's inability to transmit them. As a result, the majority of the data are lost, and the rest stay in the queue for a long period before being served. The delay for the remaining applications varies depending on the scenario.

In summary, it is generally preferable to use the SP algorithm for all queues in the TSN scenarios except for queue 5, designated for the AR application which should be based on the CBS algorithm with an IdleSlope value of 0.5.

The FP mechanism should be used in place of TAS, which has negative effects on all applications except the critical ones. However, the AR application continues to suffer failures and requires alternative solutions, as seen in the following scenarios.

TABLE VII
RDR% FOR MODEL APPLICATIONS IN TSN SCENARIOS

| App. | TSN Mechanisms | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | SP | CBS (0.1) | CBS (0.3) | CBS (0.5) | CBS (0.7) | CBS (0.9) | TAS | FP |
| Robott cntrl | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| Safety | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| AR | 10 | 1 | 3 | 5 | 6 | 8 | 2 | 5 |
| AGV | 0 | 100 | 100 | 100 | 72 | 25 | 13 | 100 |
| CM | 0 | 100 | 100 | 100 | 1 | 0 | 100 | 100 |
| update | 70 | 70 | 70 | 70 | 70 | 70 | 35 | 70 |

The RDR% results for the applications when the peripheral links of the network are improved to 1 Gbps are displayed in Fig. 9. It can be seen that the RDR % for the AR application increases to 100%. However, it should be highlighted that the lower-priority applications have no RDR% value; this is because the AR application consumes the entire bandwidth of switch D to transfer its data at a rate of around 1 Gbps, thereby, preventing any service from being allocated to those applications. On the other hand, it is noted that the RDR % for the update application reaches 100%, as the port bandwidth of switch B and links in this scenario have been upgraded to 1 Gbps, allowing all of its data to be successfully transmitted unlike in the previous scenarios.

TABLE VIII
END TO END (MS) DELAY FOR MODEL APPLICATIONS IN TSN SCENARIOS

| App. | TSN Mechanisms | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | SP | CBS (0.1) | CBS (0.3) | CBS (0.5) | CBS (0.7) | CBS (0.9) | TAS | FP |
| Robott cntrl | 0.3 | 0.3 | 0.3 | 0.3 | 0.3 | 0.3 | 0.138 | 0.15 |
| Safety | 0.37 | 0.37 | 0.37 | 0.37 | 0.37 | 0.37 | 0.37 | 0.37 |
| AR | 902 | 905 | 903 | 902 | 902 | 902 | 903 | 902 |
| AGV | --- | 0.76 | 0.8 | 0.9 | 11 | 30 | 58 | 0.9 |
| CM | --- | 1.2 | 2.2 | 7.4 | 13.5 | --- | 2.3 | 8 |

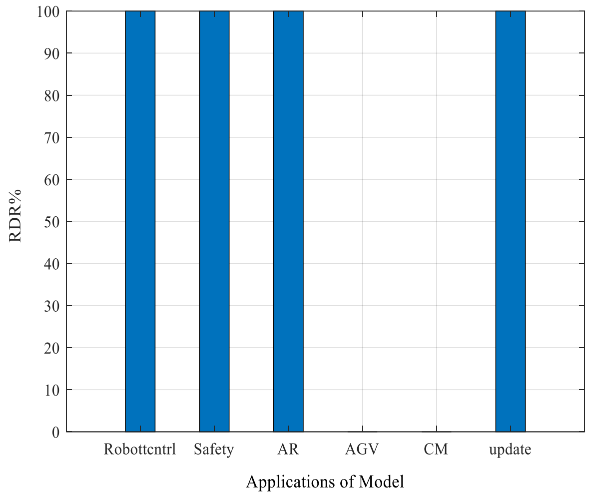


Fig. 9. RDR% for Model Applications in Upgrade Scenario.

Conversely, the network upgrades lead to a substantial reduction in the delay values (µs) for applications data received, as illustrated in Fig. 10. The delay for the remote-control application decreases from approximately 0.15 ms at 100 Mbps to 40 µs, and for the safety application from 0.37 ms to 60 µs, whereas the delay for the AR application remains the highest at approximately 430 µs. However, since the AGV and CM applications do not receive any service, no values are recorded for the delay time.

As can be seen from previous scenarios, the AR application can be considered the main reason why the network is not sufficiently reliable to serve all applications and exhausts the network resources due to the huge size of its data. Therefore, the AR data are compressed using several compression ratios.

The optimal compression ratio for achieving reliability is 22:1 at a bandwidth of 100 Mbps for peripheral links; however, at 1 Gbps, a compression ratio of 2:1 is required to achieve full reliability for the network. Fig. 11 illustrates this effect.

It is worth mentioning that the main limitation of the proposed method lies in the accuracy of the adopted AR application. Although, the compression technique helps to manage the huge size of the transmitted data regarding the mentioned application, but the accuracy of the application will be effected negatively in particular at the higher compression ratio, therefore, a balance must be established between the compression ratio and the quality of the compressed video.

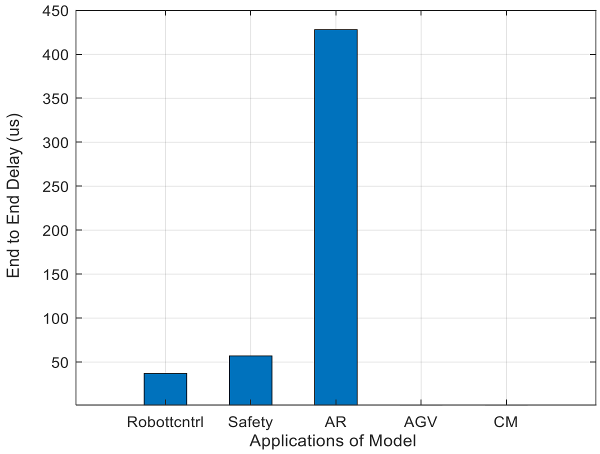


Fig. 10. End to End Delay for Model Applications in Upgrade Scenario.

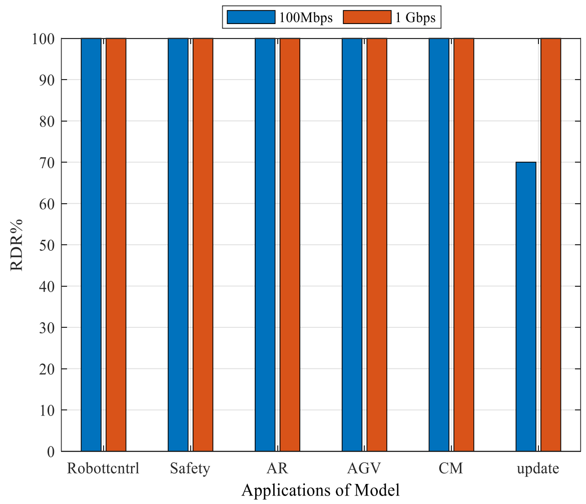


Fig. 11. RDR% for Model Applications in Enhanced Scenario.

To ensure that the level of AR data compression has no effect in terms of application quality, the PSNR value is empirically measured. It is equal to 36 dB for a compression ratio of 22:1, which is deemed acceptable according to the relationship illustrated in Fig. 5.

Fig. 12 (a) and (b) show samples of recorded AR video before and after the compression process. In this scenario, edge computing is utilized alongside the CBS algorithm with an IdleSlope of 0.5 to serve the AR data from queue 5, while the SP algorithm is employed to serve data from the remaining queues. Additionally, time-sensitive applications are supported by the FP mechanism.

The delay values for the model applications at 100 Mbps and 1 Gbps in the enhanced scenario are shown in Fig. 13; it can be seen that all applications have delay values within the requirements for the applications. The AR application has the largest delay, due to the additional processing time required to process its data at the edge server. This is one of the drawbacks of data compression based on the edge computing concept.

(a)

(b)

Fig. 12. AR Video Sample (a) Before Compression, (b) After Compression.

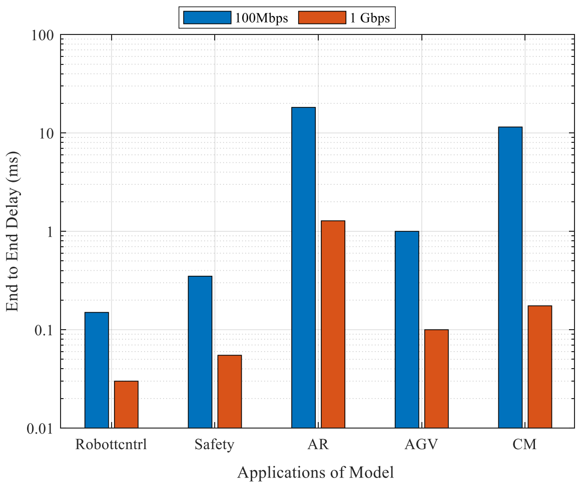


Fig. 13. End to End Delay for Model Applications in Enhanced Scenario.

Fig. 14 illustrates the delay values for the control signals produced by the local controllers (Actuators_control and Vehicles_control) in response to specific events requiring an immediate action. It can be seen that, except for the FIFO and TAS scenarios, the delay values for these signals upon transmission remain within the specified limits of the requirements of these applications. Conversely, it has been observed that in the remaining scenarios, no delay values recorded for these signals, as their data failed to reach the controllers due to network transmission failure.

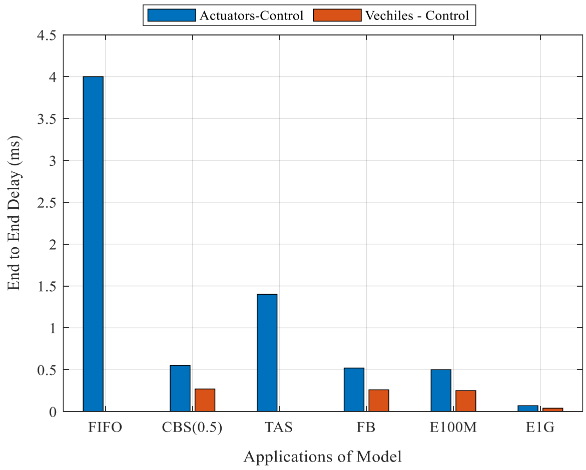


Fig. 14. End to End Delay for Control Applications of Different Scenarios.

Table IX shows a comparison of all scenarios according to how well they meet the requirements of the main smart factory applications selected for this work.

TABLE IX
HANDLING THE APPLICATIONS REQUIREMENTS ACCORDING TO NETWORK SCENARIOS

| Scenario | The Main of Smart Factory Applications | | | | |
|---|---|---|---|---|---|
| | Remote_Cntrl | Safety | AR | AGV | CS |
| Basic | × | × | × | × | × |
| TSN(SP) | √ | √ | × | × | × |
| TSN(CBS) | √ | √ | × | √ | √ |
| TSN(TAS) | √ | √ | × | × | √ |
| TSN(FP) | √ | √ | × | √ | √ |
| Upgrade | √ | √ | √ | × | × |
| Enhance | √ | √ | √ | √ | √ |

## VI. CONCLUSION

This work has presented a reliable communication approach for next-generation industrial networks, specifically targeting the requirements of smart factories in the context of Industry 4.0. By integrating TSN with edge computing, the proposed model effectively addresses the challenges posed by diverse applications requiring varying levels of end to end delay, bandwidth, and reliability. The experimental results demonstrate that the enhanced communication network achieves impressive performance metrics, including a delay of less than 1 ms for real-time applications and RDR% of 99.999%. These results highlight the feasibility of deploying TSN algorithms such as SP and FP in conjunction with advanced data compression techniques such as H.265 to optimize network performance. Moreover, this work emphasizes the critical role of edge computing in managing heavy data loads in order to ensure that applications such as AR can function seamlessly without degrading the performance of real-time systems. The balance achieved between various application requirements underscores the potential for this integrated approach to support complex industrial environments. Future works may concentrate on further refining the model to integrate adaptive mechanisms that respond dynamically to changing network conditions and application requirements, thereby enhancing the resilience and efficiency of smart factory operations. This work's achievements substantially advance the development of industrial communication networks, facilitating the emergence of more intelligent and responsive manufacturing systems.

## Acknowledgment

The authors would like to express their gratitude to the University of Mosul, College of Engineering, Department of Electrical, for providing the necessary facilities that contributed to enhancing the quality of this paper.

## References

[1] Y. S. Sheet, M. Younis Thanoun, and F. S. Alsharbaty, "Smart Factory based on IIoT: Applications,Communication Networks and Cybersecurity," International Journal of Wireless and Microwave Technologies, vol. 14, no. 4, pp. 29-47, 2024, doi: 10.5815/ijwmt.2024.04.03.

[2] J. H. David Lou , Dhruvin Patel , Ulrich Graf and Matthew Gillmore (Itron). "The Industrial Internet of Things Networking Framework," Industry IoT Consortium. https://www.iiconsortium.org/iicf, 2022.

[3] I. B. a. H. Austad, "Real-Time Performance of Industrial IoT Communication Technologies: A Review," IEEE Internet of Things Journal,, vol. 11, no. 5, pp. 7399 - 7410, 2023, doi: 10.1109/jiot.2023.3332507.

[4] A. R. M. Noor et al., "Wireless Communications for Smart Manufacturing and Industrial IoT: Existing Technologies, 5G and Beyond," Sensors (Basel), vol. 23, no. 1, Dec 21 2022, doi: 10.3390/s23010073.

[5] C. Xu, H. Yu, X. Jin, C. Xia, D. Li, and P. Zeng, “Industrial Internet for intelligent manufacturing: past, present, and future,” Frontiers of Information Technology and Electronic Engineering, vol. 25, no. 9. Zhejiang University, pp. 1173–1192, Sep. 01, 2024. doi: 10.1631/FITEE.2300806.

[6] P. Galeas et al., "EtherCAT as an alternative for the next generation real-time control system for telescopes," Journal of Astronomical Telescopes, Instruments, and Systems, vol. 9, no. 01, 2023, doi: 10.1117/1.Jatis.9.1.017001.

[7] T. Zhang, G. Wang, C. Xue, J. Wang, M. Nixon, and S. Han, “Time-Sensitive Networking (TSN) for Industrial Automation: Current Advances and Future Directions,” ACM Computing Surveys, vol. 57, no. 2, Oct. 2024, doi: 10.1145/3695248.

[8] H. Trifonov and D. Heffernan, "OPC UA TSN: a next-generation network for Industry 4.0 and IIoT," International Journal of Pervasive Computing and Communications, vol. 19, no. 3, pp. 386-411, 2021, doi: 10.1108/ijpcc-07-2021-0160.

[9] A. B. D. K. J. B. M. A. A. Lysko, "An Overview of Time-Sensitive Communications for the Factory Floor," presented at the 2021 IST-Africa Conference (IST-Africa), South Africa, South Africa, 2021.

[10] C. S. M. M. Bella, "Ethernet with Time Sensitive Networking Tools for Industrial Networks," INFOCOMMUNICATIONS JOURNAL,IX. 6-14, 2017.

[11] 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," presented at the 2021 IFIP Networking Conference (IFIP Networking), 2021.

[12] A. Nasrallah et al., "Performance Comparison of IEEE 802.1 TSN Time Aware Shaper (TAS) and Asynchronous Traffic Shaper (ATS)," IEEE Access, vol. 7, pp. 44165-44181, 2019, doi: 10.1109/access.2019.2908613.

[13] Y. L. Junhui Jiang, Seung Ho Hong,Aidong Xu, Kai Wang, "A Time-sensitive Networking (TSN) Simulation Model Based on OMNET++," presented at the Proceedings of 2018 IEEE International Conference on Mechatronics and Automation August 5 - 8, , Changchun, China, 2018.

[14] G. Tong, J. Li, and H. Gao, "Simulation of Time Delay Characteristics of Time Sensitive Networking Based on MATLAB/Simulink," Journal of Physics: Conference Series, vol. 2005, no. 1, 2021, doi: 10.1088/1742-6596/2005/1/012178.

[15] D. Bezerra et al., "A machine learning-based optimization for end-to-end latency in TSN networks," Computer Communications, vol. 195, pp. 424-440, 2022, doi: 10.1016/j.comcom.2022.09.011.

[16] A. G. Hagargund, M. kulkarni, and H. S. Satheesh, "Performance analysis of cost effective multi-hop Time Sensitive Network for IEEE 802.1Qbv and IEEE 802.1Qbu standards," Journal of Physics: Conference Series, vol. 2161, no. 1, 2022, doi: 10.1088/1742-6596/2161/1/012002.

[17] M. Seliem, A. Zahran, and D. Pesch, "Quality Checks After Production: TSN-based Industrial Network Performance Evaluation," presented at the 2022 4th International Conference on Electrical, Control and Instrumentation Engineering (ICECIE), 2022.

[18] D. L. Gomez, G. A. Montoya, C. Lozano-Garzon, and Y. Donoso, "Strategies for Assuring Low Latency, Scalability and Interoperability in Edge Computing and TSN Networks for Critical IIoT Services," IEEE Access, vol. 11, pp. 42546-42577, 2023, doi: 10.1109/access.2023.3268223.

[19] P. Pop, K. Alexandris, and T. Wang, “Configuration of multi-shaper Time-Sensitive Networking for industrial applications,” IET Networks, vol. 13, no. 5–6, pp. 434–454, Sep. 2024, doi: 10.1049/ntw2.12129.

[20] F. Luo et al., “Schedulability analysis of time aware shaper with preemption supported in time-sensitive networks,” Computer Networks, vol. 269, p. 111424, Sep. 2025, doi: 10.1016/j.comnet.2025.111424.

[21] P. Cuenot, Q. Bailleul, P. Cuenot, and Q. Bailleul, “Evaluation of Time-Sensitive Networking for future centralized architecture.” [Online]. Available: https://hal.science/hal-04958002v1.

[22] F. Luo et al., "Design Methodology of Automotive Time-Sensitive Network System Based on OMNeT++ Simulation System," Sensors (Basel), vol. 22, no. 12, Jun 17 2022, doi: 10.3390/s22124580.

[23] Y. S. Sheet, M. Younis Thanoun, and F. S. Alsharbaty, “The Future of Smart Factory: Wireless Time-Sensitive Networks for Industry 4 Applications,” International Journal on Electrical Engineering and Informatics, vol. 17, no. 3, 2025, doi: 10.15676/ijeei.2025.17.3.8.

[24] M. M. Hasan, "Performance Analysis for Deterministic System using Time Sensitive Network," Master, School of Electronic and Information Engineering Beihang university, China, 2022.

[25] J. Lee and S. Park, "Time-Sensitive Network (TSN) Experiment in Sensor-Based Integrated Environment for Autonomous Driving," Sensors (Basel), vol. 19, no. 5, Mar 5 2019, doi: 10.3390/s19051111.

[26] R. E. Lopes, D. Raposo, R. Martins, and S. Sargento, “Time-Sensitive Smart Cities: Addressing Smart City Challenges With TSN,” IEEE Access, vol. 13, pp. 10459–10480, 2025, doi: 10.1109/ACCESS.2025.3528257.

[27] O. M. Ali and A. F. Mahmood, “Edge Computing Towards Smart Applications: A Survey,” Recent Advances in Computer Science and Communications, vol. 16, no. 1, Jan. 2023, doi: 10.2174/2666255815666220225102615.

[28] A. A. Kadhim, A. M. Abid, and Z. H. Ali, "Subject Review: Video Compression Algorithms," International Journal of Engineering

Research and Advanced Technology, vol. 06, no. 11, pp. 21-25, 2020, doi: 10.31695/ijerat.2020.3668.
[29] T. Dong, K. J. Oh, J. Park, and E. S. Jang, "Compression Performance Analysis of Experimental Holographic Data Coding Systems," Sensors (Basel), vol. 23, no. 18, Sep 6 2023, doi: 10.3390/s23187684.
[30] J. F. D. H. B. C. N. N. F. D. S. Kehrer, "NeSTiNg: Simulating IEEE Time-sensitive Networking (TSN) in OMNeT++," presented at the 2019 International Conference on Networked Systems (NetSys), Munich, Germany, 2019.
[31] [Online]. Available: https://handbrake.fr/.
[32] www.compression.ru/video/quality_measure/vqmt_download.ht
[33] Video Clarity, Understanding MOS, JND, and PSNR – Tools for Video Analysis, Video Clarity Inc., White Paper, 2020. [Online].Available:https://videoclarity.com/videoqualityanalysiscasestudies.

**Yazen S. Sheet** completed the B.S. in electrical engineering / electronics and communication from the University of Mosul, Iraq, in 2005 and received an M.Sc. degree in computer networks in 2011 from Mosul University. He interested in the field of computer networks and communication and he had published many papers in this field. He has been working as a communications and computer network lecturer at the University of Mosul Since 2011.

**Mohammed Younis Thanoun** completed the B.S. in electrical engineering / electronics and communication from the University of Mosul, Iraq, in 1991 and received the M.Sc. degree in electronics and communication in 2000 and the Ph.D. in 2011 from Mosul University. He is interested in the field of computer networks and communication and has been working as an assistant professor at the University of Mosul since 2021.

**Firas S. Alsharbaty** completed the B.S. in electrical engineering / electronics and communication from the University of Mosul, Iraq, in 2007 and received the M.Sc. degree in computer networks and communication in 2010 and the Ph.D. in 2023 from Mosul University. He is interested in the field of computer networks and communication and has many published many papers in the industrial communication networks field.